1996

IMAGE PROCESSING LSIS AND MODULES

SUMITOMO METALS

IMAGE PROCESSING LSIS AND MODULES

REAL-TIME IMAGE PROCESSING

SUMITOMO METALS

IMAGE PROCESSING LSIs AND MODULES

◆ REAL-TIME IMAGE PROCESSING ◆

JPA Electronics Supply, Inc.
Park 80 West Plaza 1
Saddle Brook, NJ 07663

Sumitomo Metal Industries, Ltd.
Development Section
Electronic Components Division
108 Fuso-cho, Amagaski
Hyogo, 660 Japan

SUMITOMO METALS

Sumitomo Metal Industries

IMPORTANT NOTICE

While Sumitomo Metal Industries, Ltd. (hereinafter "SMI") believes that this manual is accurate and complete as of its date of issuance, SMI assumes no liability for damages or injuries resulting from any errors or omissions in this manual, or from the use of information contained in this manual. SMI reserves the right to revise this manual and to make changes from time to time in the content hereof without having any obligation to notify any person, including without limitation purchasers of the products described herein, of such revision or changes. SMI also reserves the right to make changes in the products or to discontinue any such product without notice and advises purchasers to obtain the latest version of relevant information before placing any order, to verify that the information being relied on is current.

The circuit diagrams described herein are shown merely as typical examples of product applications and are not intended to guarantee operation of any circuit designed by the purchaser. SMI makes no warranty of any kind as to user support in the application of this product, nor assumes any responsibility for customer designs, or infringement of any of their party's patents, copyrights or other intellectual property rights resulting from the use of the products described in this manual.

Use of the products described herein in certain applications ("Critical Applications") may involve potential risks of death, personal injury or severe property or environmental damage.

THE PRODUCTS DESCRIBED HEREIN ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Use of the products described in this manual in any Critical Application requires the prior written approval of an authorized representative of SMI. In order to minimize risks associated with the purchaser's applications, adequate design and operating safeguards should be established by the purchaser to minimize inherent or procedural hazards.

The products described herein generate, use and can radiate radio frequency energy. Operation of such product in a residential area is likely to cause radio interference, in which case the purchaser, at his own expense, will be required to take whatever measures may be required to correct the interference.

SMI makes no representation or warranty that any license, either express or implied, is granted under any patent right, copyright, mask work right or other intellectual property right of SMI covering or relating to any combination, machine or process in which the products described in the manual might be or are used.

WARNING

Use of the products in certain applications ("Critical Applications") may involve potential risks of death, personal injury or severe property or environmental damage.

THE PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED OR WARRANTED TO BE SUITABLE FOR USE IN LIFE SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Use of the products in any Critical Application requires the prior written approval of an authorized representative of Sumitomo Metal Industries, Ltd. In order to minimize risks associated with the purchaser's applications, adequate design and operating safeguards should be established by the purchaser to minimize inherent or procedural hazards.

Sumitomo Metal Industries

REAL-TIME IMAGE PROCESSING LSIs/MODULES DATA BOOK

Table of Contents

LSI Chips

SIDIP Modules

IP90MD08 **Template Matching Module...PM VER. E 1.1**
Template matching module with core template matching chip (IP90C08) and peripheral circuitry. Single size. (48.8mm x 127.0mm)

IP90MD10 **Labeling Module...PM VER. E 1.3**
Entire labeling module with LABop (IP90C10) chip and peripheral circuitry. Double size. (98.6mm x 127.0mm)

IP90MD15 **Sketch Module...PM VER. E 1.2**
Averaging image data reduction and simple image data expansion module with Sketch (IP90C15) chip and peripheral circuitry. Double size. (98.6mm x 127.0mm)

IP90MD20 **Rank Value Filter Module...PM VER. E 1.2**
Rank value filter module with RKFil (IP90C20) chip and peripheral circuitry. Single size. (48.8mm x 127.0mm)

IP90MD25 **Spatial & Logical Filter Module...PM VER. E 1.1**
Spatial and logical filter module with SLFC (IP90C25) chip and peripheral circuitry. Single size. (48.8mm x 127.0mm)

IP90MD81/ 01/05 **Histogram / Projection Module...PM VER. E 1.2**
Five types offered. They have one or two functions in 8-bit value histogramming (IP90C01) and / or X-Y axes gray level projection (IP90C05A-HS). Single size. (48.8mm x 127.0mm)

IP90MD100 **Frame Memory Module...PM VER. E 1.1**
Expansion frame memory for image processing systems which has two frames capability. Single size. (48.8mm x 127.0mm)

Baseboards

IP90BD301 **Baseboard for ISA bus (PC/AT)...PM VER. E 2.1**
Gray scale frame grabber board with two image stream exchange chips (IP90C55). 3 slots for SIDIP modules.

IP90BD351 **Expansion Board for ISA bus (PC/AT)...PM VER. E 2.1**
3 slots for SIDIP modules.

IP90BD550 **Baseboard for VME bus (6U)...PM VER. E 1.3**
Expansion board with two image stream exchange chips (IP90C55). 4 slots for SIDIP modules.

IndustryPack

IP90MS800 **Area Sensor Image Frame Grabber...PM VER. E 1.2**
An 8-bit monochrome frame grabber for area sensors with IndustryPack standard. IndustryPack double wide.

IP90MS803 **Line Sensor Image Frame Grabber...PM VER. E 1.2**
An 8-bit monochrome frame grabber for line sensors with IndustryPack standard. IndustryPack double wide.

SMI ASSP

Image Processing LSI Series

IP90C01

Histogramming Processor (HIST)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 3.1

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 6: Sample Applications

Section 1. Overview

1.1 Product Overview

The IP90C01 is a single-chip LSI processor for grayscale histogram processing. The IP90C01 is manufactured using the latest CMOS processes and high-speed circuit technology. The IP90C01 has a maximum operating speed (input image data transfer rate) of 50 MHz, and can process both non-interlaced and interlaced image data. The IP90C01 can process 8-bit grayscale pixel images in real-time, up to a maximum size of 1023 x 1023 pixels.

1.2 Basic Specifications

Histogram Processing Functions

- Histogram processing of 8-bit grayscale images
- Maximum image processing area: 1023 x 1023 pixels
- Maximum operating frequency: 50 MHz (IP90C01 without rank code)
  20 MHz (IP90C01-LC)
- Horizontal and vertical dimensions of the processing "window" can be set using a programmable 10-bit counter.
- Synchronized with external horizontal and vertical synchronization signals
- One-shot clearing of histogram data

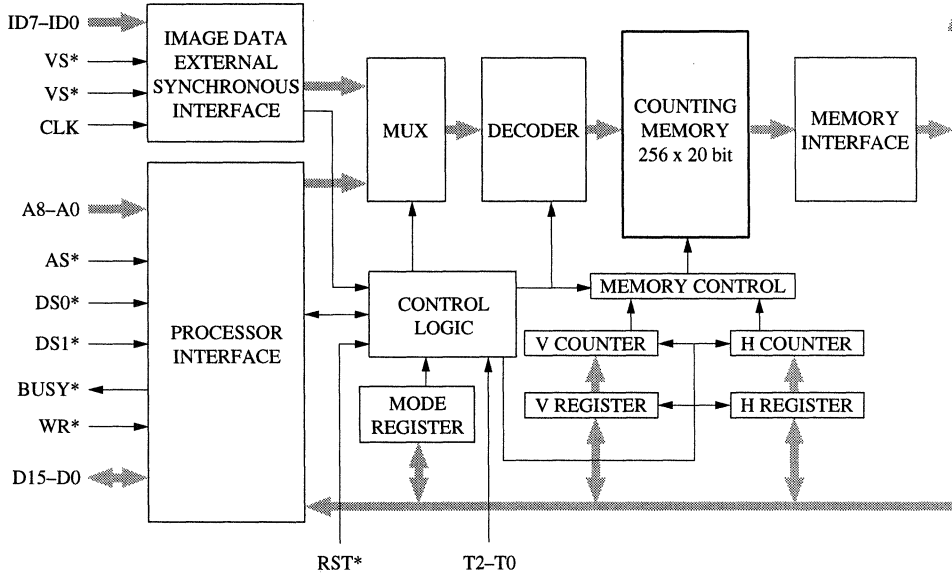
External Interfaces

- General interface through 16-bit bus
- Mode register for operating mode settings:
  - Histogram processing mode
  - Histogram processing results read mode
  - Memory clear mode
- Data readout by word:
  20-bit data upper-word/lower-word selection (selected by DS1*, DS0* pins)

Other

- Process: CMOS
- Power supply: 5V single power supply
- Input/output interface level: TTL compatible
- Package: QFP 64-pin (plastic)

1.3 Block Diagram

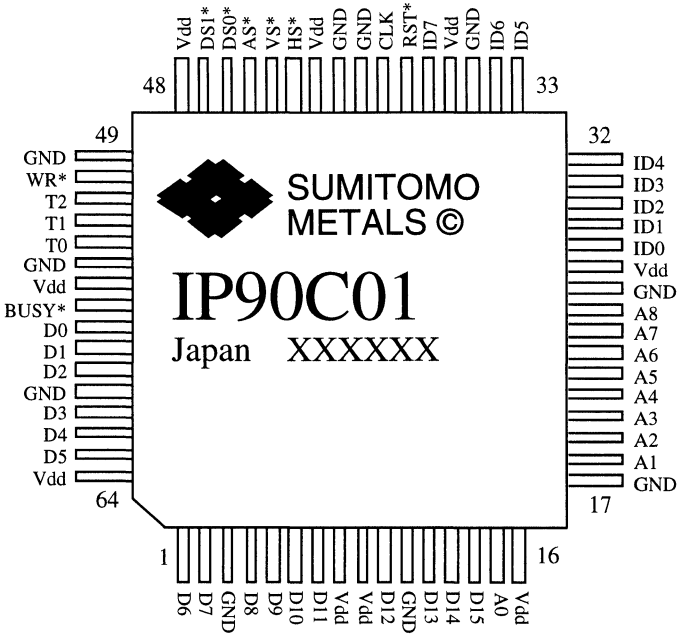


IP90C01 Block Diagram

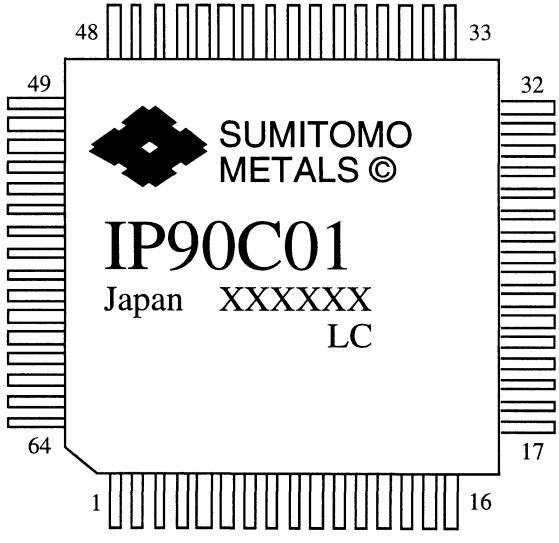
Section 2. Pin Functions

2.1 Pin Placement and Pin Assignments

IP90C01



IP90C01-LC



Note: The IP90C01 and IP90C01-LC have different markings, but have identical pin assignments.

Pin Assignment Table

Pin No.	Name
1	D6
2	D7
3	GND
4	D8
5	D9
6	D10
7	D11
8	Vdd
9	Vdd
10	D12
11	GND
12	D13
13	D14
14	D15
15	A0
16	Vdd
17	GND
18	A1
19	A2
20	A3
21	A4
22	A5

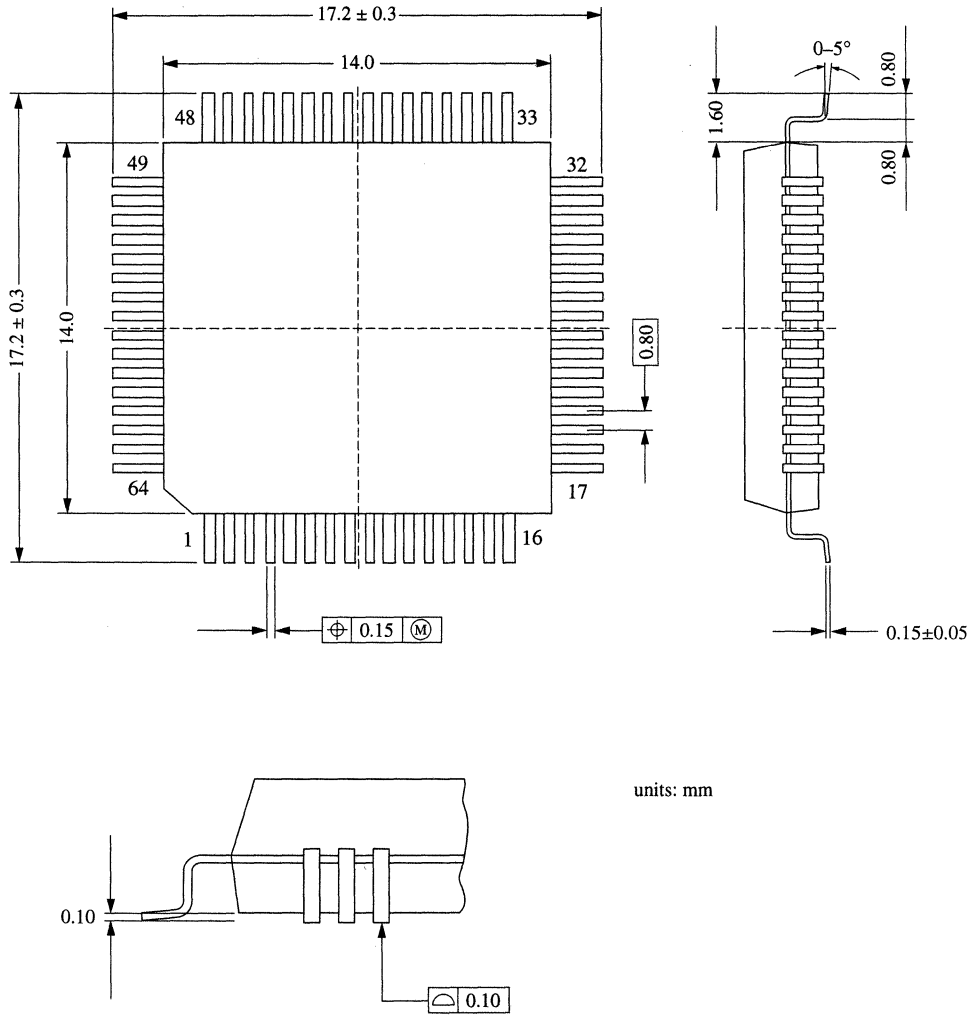
Pin No.	Name
23	A6
24	A7
25	A8
26	GND
27	Vdd
28	ID0
29	ID1
30	ID2
31	ID3
32	ID4
33	ID5
34	ID6
35	GND
36	Vdd
37	ID7
38	RST*
39	CLK
40	GND
41	GND
42	Vdd
43	HS*
44	VS*

Pin No.	Name
45	AS*
46	DS0*
47	DS1*
48	Vdd
49	GND
50	WR*
51	T2
52	T1
53	T0
54	GND
55	Vdd
56	BUSY*
57	D0
58	D1
59	D2
60	GND
61	D3
62	D4
63	D5
64	Vdd

2.2 Pin Descriptions

Pin Group	Pin Names	No. of Pins/(I/O)		Description
Image input bus	CLK	1	I	Pixel sync clock signal
	ID7–ID0	8	I	8-bit pixel data bus
	HS*	1	I	Horizontal sync signal
	VS*	1	I	Vertical sync signal
CPU bus	WR*	1	I	Register write signal
	AS*	1	I	Address signal
	DS1*	1	I	Memory data upper-word select signal
	DS0*	1	I	Processor data verify signal and memory data lower-word select signal
	A8–A0	9	I	9-bit address bus
	D15–D0	16	I/O	16-bit processor data bus
	BUSY*	1	O	BUSY* signal; BUSY* is low during histogram processing
	RST*	1	I	System reset: clears memory and registers
Power supply,	Vdd	9	PW	5V power supply
ground	GND	10	PW	Ground
Other	T2–T0	3	I	Test signal (set high when in use)
Total pins		64		

2.3 External Dimensions



units: mm

Section 3: Registers

3.1 Register List (Write Only)

Table 1 shows the address of each register. Registers are selected by the address signal pins A8–A0, and register settings are made through the processor data bus. Write access to registers requires enabling the register write signal WR* and the processor data bus verify signal DS0*.

		Bit		
Register	Address	15 14 13 12 11 10 9 8 7 6 5 4 3 2	1	0
Mode register	001h	Always 0	Start histogram processing	Clear memory
V register	002h	Always 0	Vertical size of processing area	
H register	004h	Always 0	Horizontal size of processing area	

Table 1: Register List

3.2 Mode Register (2-bit)

The mode register has a 2-bit configuration and controls the IP90C01's operating mode (histogram processing mode or memory clear mode). The mode register is set through bits D1–D0 of the data bus. (The upper 14 bits D15–D2 are not used and should be set to 0, but are available for upwardly-compatible products.)

(1) Memory Clear Mode

bit 0: Write 1 to this bit to clear the IP90C01's memory. The time required to clear the memory is the same as the system reset pulse width. To release the clear setting, write 0 to this bit.

(2) Histogram Processing Mode

bit 1: Write 1 to this bit to put the IP90C01 in histogram processing mode. The chip then begins histogram processing at the next VS* pulse. When histogram processing is finished, this bit is automatically reset to 0 (so there is no need to write 0 to this bit).

3.3 Vertical and Horizontal Processing Width Registers

(1) V Register (10 bits)

This register controls the vertical size of the processing area (from 1–1023), using processor data bus bits D9–D0.

(2) H Register (10 bits)

This register controls the horizontal size of the processing area (from 1–1023), using processor data bus bits D9–D0.

Section 4. Operating Description

4.1 Overview of Operation

The IP90C01 performs real-time histogram processing on screen units of 8-bit grayscale image data, at processing speeds (input image data transfer rates) from 500 kHz to 50 MHz. The IP90C01 can handle image types from NTSC-format to high-density image format. In addition, it provides independent vertical and horizontal register settings to set the processing area to any size from 1 x 1 to 1023 x 1023 pixels.

4.2 Histogram Processing

Histogram processing on the IP90C01 uses a special memory designed exclusively for histogram processing, along with the proprietary Dynamic Allocated Computing Method developed by Sumitomo. Once bit 1 in the mode register is set to 1, histogram processing begins when CLK falls after the next VS* pulse. The vertical dimension of the processing area is controlled by counting the number of HS* pulses according to the setting of the V register. Similarly, the horizontal dimension is controlled by counting CLK signals according to the setting of the H register. Once the IP90C01 has processed the area set by the V and H registers, it resets bit 1 of the mode register, and histogram processing ends.

Once bit 1 of the mode register is set to 1, the IP90C01 sets BUSY* low. After histogram processing is completed, BUSY* is reset to high at the next VS* pulse. For the relationship of the I/O signals used in histogram processing, see Section 4.6, "System Operation Timing Charts."

4.3 Reading Histogram Processing Results

The results of histogram processing can be read from a special memory table. Giving the MSB of the address the value 1, the lower 8 bits can be read as 20-bit histogram data values, corresponding to each bit in the 8-bit grayscale image. Table 2 shows how memory space is assigned.

A8	A7	A6	A5	A4	A3	A2	A1	A0
1	8-bit grayscale data							

Table 2: Memory Space Assignment

To read the lower 16 bytes of the histogram data, the address verification signal AS* must be enabled while DS0* is enabled. To read the upper 4 bits, DS1* must be enabled. However, AS* must be enabled for every word. Table 3 shows the relation between address and grayscale values.

Grayscale Value	Address A8–A0	Histogram data (20-bit)
0	100h	D3–D0, D15–D0
15	10Fh	(upper word), (lower word)
255	1FFh	

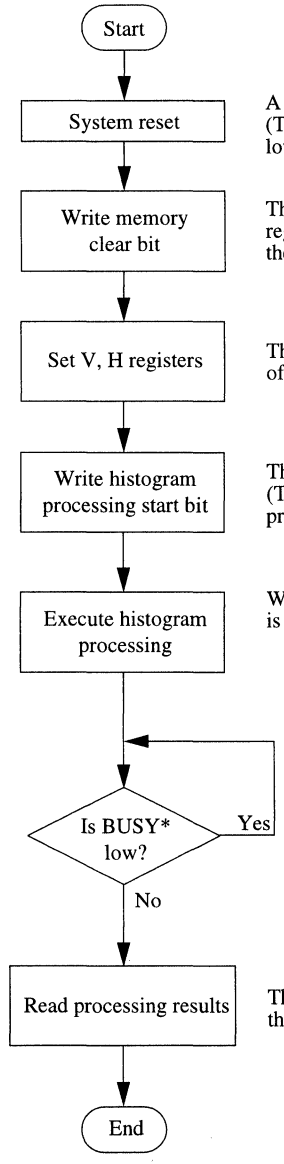
Table 3: Relation Between Addresses and Grayscale Values in Reading Results of Histogram Processing

4.4 Clearing Histogram Processing Results

The results of histogram processing can be cleared from memory by sending a low-level reset signal RST*, or by writing 1 to bit 0 of the mode register.

4.5 Operating Flow

The following chart shows the flow of IP90C01 operation.



A minimum 200-ns low-level pulse is applied. (The IP90C01-LC requires a minimum 300-ns low-level pulse.)

The CPU writes the value 0001(h) to the mode register to clear memory. (The value 0000(h) cancels the clearing operation.)

The CPU sets the horizontal and vertical dimensions of the processing area.

The CPU writes the value 0002(h) to the mode register. (The IP90C01 chip automatically resets the histogram processing start bit to 0 after the process ends.)

When the histogram processing start bit (described above) is set, the IP90C01 chip does the following:

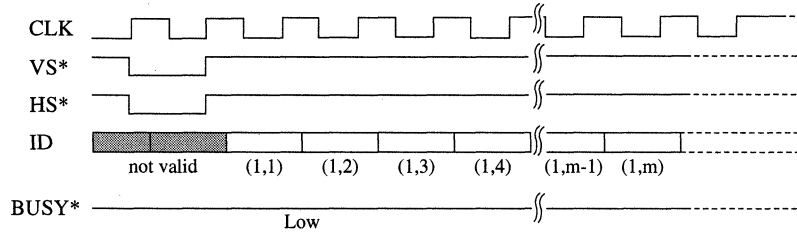
(1) Sets BUSY* low.
(2) Starts histogram processing simultaneously with VS*.
(3) Stops processing after the number of CLK signals determined by the horizontal processing width register, and the number of HS* pulses determined by the vertical processing width register.
(4) With the next VS*, BUSY* returns to high.

The CPU verifies that BUSY* is high, then reads the memory contents.

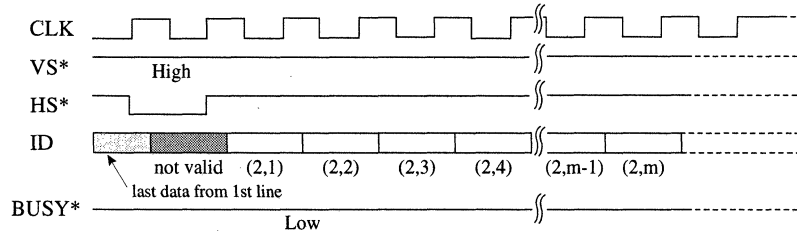
4.6 System Operation Timing Charts

Figure 4-1 shows the signals and timing relationships necessary for IP90C01 system operation. The figure also shows the signals and dummy cycles used for histogram processing.

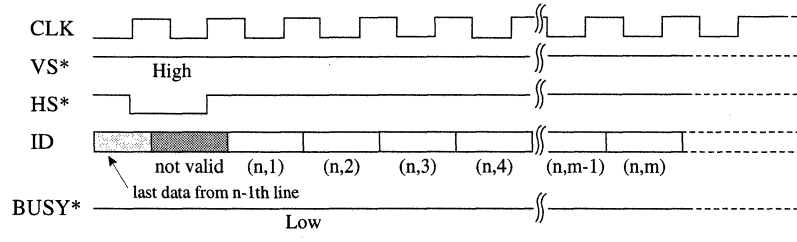
(1) First row of field



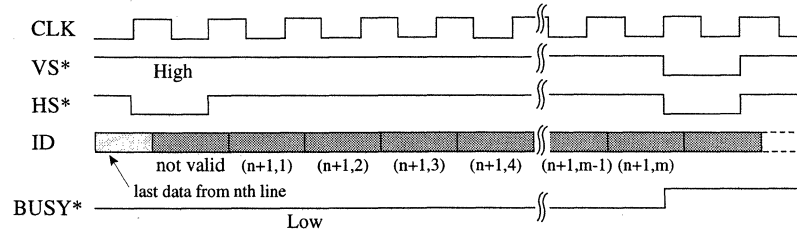
(2) Second row



(3) n^{th} row (last row of valid data)



(4) $n+1^{th}$ row (terminator row)



Note: At the next VS*, BUSY* goes high and the IP90C01 switches into a mode that allows processing results to be read.

Figure 4-1: Histogram Processing Timing Charts

4.7 Dummy Cycle HS and VS Input

Figure 4-1 (4) shows cycles that are required for reasons other than data input. These cycles are defined in units of one clock pulse width. This means that an HS* cycle is a cycle of one clock pulse width in which HS* goes low, and a data input cycle is one clock cycle in which one set of grayscale values is input. The dummy cycle required at the end of a field includes the VS* cycle that indicates the start of a field, the HS* cycle that marks the start of a line, and the data input cycle. This dummy cycle follows the input of the last row of data and includes one cycle beyond the width of the H register, as well as the following VS* cycle. In effect, the HS* cycle introduces one row of dummy input data followed by a VS* cycle, as shown in Figure 4-1 (4).

Images from CRT screens and CCD camera devices use blanking, and normally present no problem as input. However, images transferred from CPU space or DMA transfer may come without blanking or in compressed format.

4.8 Image Clock Input (CLK)

The image clock signal is normally present, and systems should be designed accordingly.

4.8.1 Conditions for Stopping the Clock While the CPU Reads Results

The image clock signal is not involved in reading histogram processing results, and so can be stopped while results are read. To assure control stability, however, observe the same conditions as for histogram processing without the clock signal, as described below.

4.8.2 Stopping the Clock Signal During Histogram Processing

Histogram processing at high speeds is made possible by incorporating dynamic circuits into the IP90C01. For this reason, the technical documentation places maximum limits on the clock signal frequency (Tcc). To ensure circuit stability, input the image clock signal at all times. When this is not possible, or when operating close to the maximum frequency, make sure the image clock signal CLK remains high.

4.9 HS* and VS* Input at Start of Field

4.9.1 Basic Timing Pattern - Figure 4-2 (1)

The basic timing pattern (as described in Figure 4-2 (1) and Section 5.4, "AC Characteristics") assumes that a field's first VS* and HS* pulses are input in the same cycle of the clock signal.

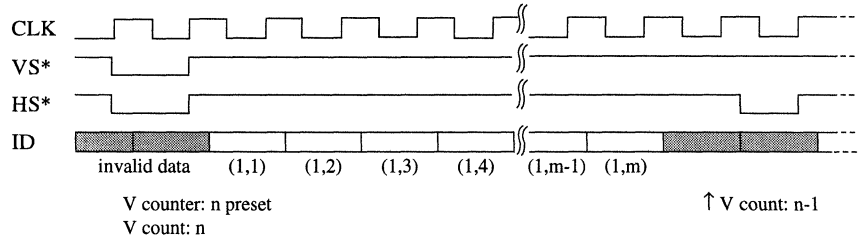
4.9.2 VS* Pulse and HS* Pulse Not Timed Together - Figure 4-2 (2)

Note the timing of the VS* and HS* pulses in Figure 4-2 (2). When the VS* pulse is input at the start of a field, but the HS* pulse is not input at the same time, the V counter that determines the vertical size of the image area is 1 higher than in the basic pattern described in Section 4.9.1.

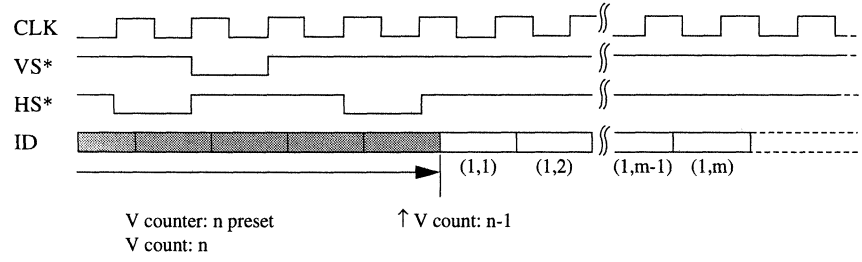
This affects the sequence of internal processing like this: Histogram processing starts with the data that accompanies the next clock signal received after the HS* pulse input following the VS* pulse. By the time this delayed HS* pulse is detected, however, the V counter has already recorded one pulse count. To allow for this, the setting written into the V register must be increased by one. In comparison, the basic pattern described in Section 4.9.1 (in which the VS* and HS* pulses arrive together) allows the loading of data from the V register to take priority over the V counter cycle, so that no V counter value is recorded before the cycle starts.

(1) Standard HS*, VS* pulse input timing (start of field)

Example: timing when VS* and HS* are input simultaneously



(2) Timing When VS* and HS* are offset (start of field)



Note: An asterisk following a signal name indicate inverse logic.

Figure 4-2: VS*, HS* Pulse Input Timing

Section 5. Electrical Characteristics

5.1 Absolute Maximum Ratings

Item	Symbol	Min.	Typ.	Max.	Unit
Power supply voltage	Vdd	-0.5	—	6.5	V
Input voltage	Vin	-0.5	—	Vdd + 0.5	V
Storage temperature	Tstg	-10	—	80	°C

5.2 Recommended Operating Conditions

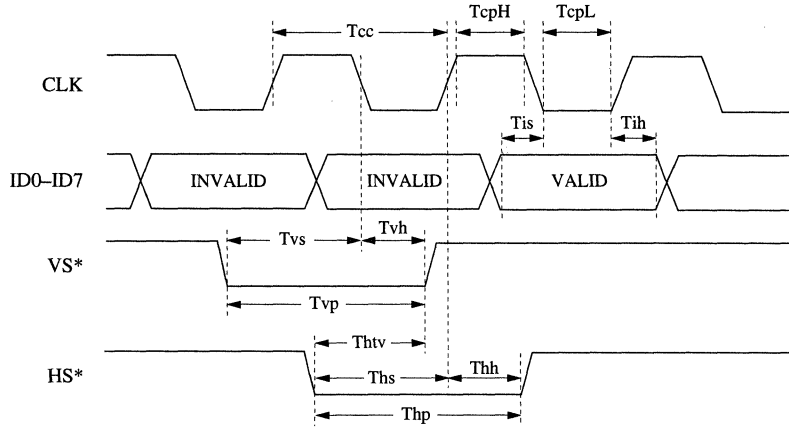
Item	Symbol	Min.	Typ.	Max.	Unit
Power supply voltage	Vdd	4.75	5.0	5.25	V
Input voltage	Vin	0	—	Vdd	V
Ambient temperature	Ta	0	—	70	°C

5.3 DC Characteristics

Item	Symbol	Conditions	Min.	Max.	Unit
High level output voltage	Voh	Vdd = Min., Ioh = 0.4 mA	2.4	—	V
Low level output voltage	Vol	Vdd = Min., Iol = 4.0 mA	—	0.4	V
High level input voltage	Vih	—	2.4	Vdd	V
Low level input voltage	Vil	—	0	0.8	V
Input leak voltage	Iix	GND ≤ Vin ≤ Vdd	-10	10	µA
Output off- leak voltage	Ioz	Vdd = Max.	-25	25	µA
Operating current	Icc	Tcc = 20 ns (IP90C01)	—	200	mA
		Tcc = 50 ns (IP90C01-LC)	—	100	—

5.4 AC Characteristics

Image data input timing



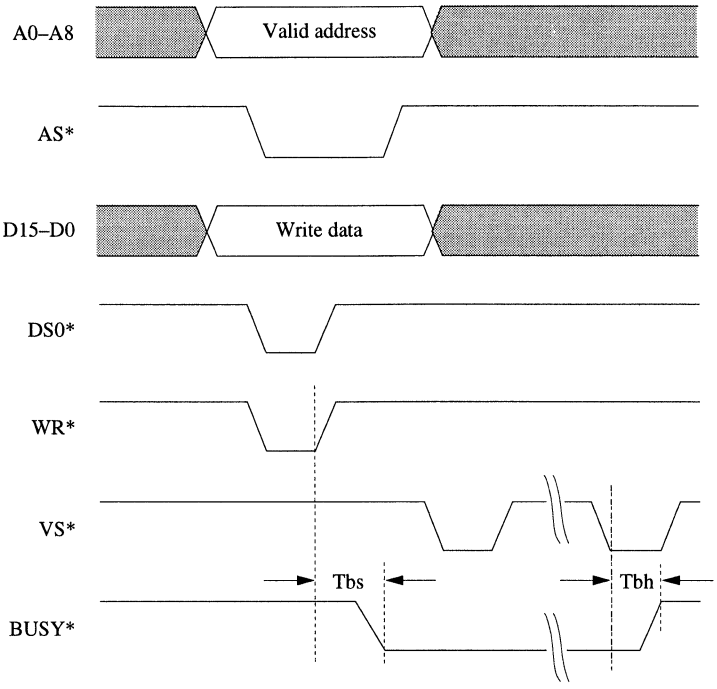
Units: ns

Item		Symbol	IP90C01			IP90C01-LC		
			Min.	Typ.	Max.	Min.	Typ.	Max.
Clock cycle		Tcc	20	—	2000	50	—	2000
Clock pulse width:	High level	Tcph	8	—	120	15	—	120
	Low level	Tcpl	8	—	120	15	—	120
ID0 to ID7 set-up time to clock fall		Tis	0	—	—	0	—	—
ID0 to ID7 hold time from clock rise		Tih	4	—	—	4	—	—
VS* fall set-up time to clock fall		Tvs	11	—	Tcc	11	—	Tcc
VS* rise hold time from clock fall		Tvh	3	—	Tcc	3	—	Tcc
VS* pulse width		Tvp	18	—	—	30	—	—
HS* fall set-up time to clock rise		Ths	12	—	Tcc	12	—	Tcc
HS* rise hold time from clock rise		Thh	0	—	Tcph	0	—	Tcph
HS* pulse width		Thp	18	—	—	30	—	—
HS* fall hold time to VS* rise		Thtv	4	—	—	4	—	—

Note 1: VS* and HS* should be low, with a width equivalent to one width of the CLK (Tcc) signal.

Note 2: The maximum limit on clock cycle length applies during operation. When not in operation, this value can be greater. If the clock is stopped during histogram processing, the clock signal CLK should be fixed at t_{high} level. For details, see Section 4.6, "System Operation Timing Charts."

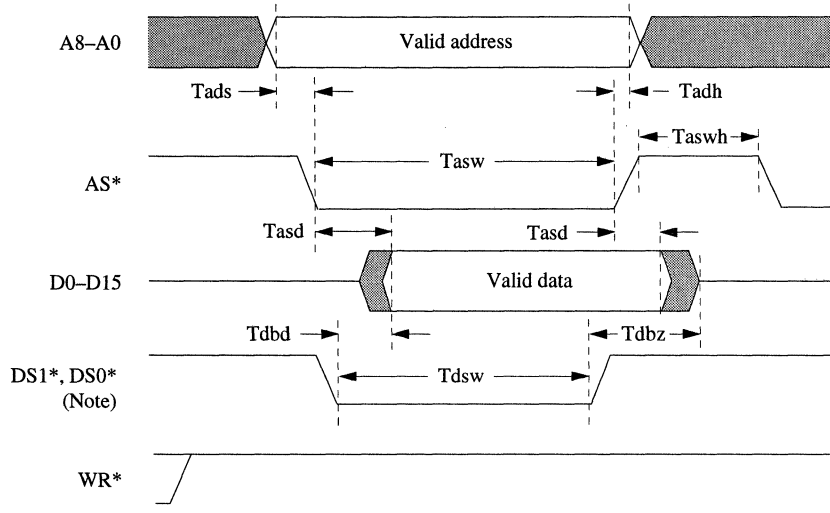
BUSY Signal Timing



Units: ns

Item	Symbol	IP90C01			IP90C01-LC		
		Min.	Typ.	Max.	Min.	Typ.	Max.
Delay from WR* rise to BUSY* fall	Tbs	—	—	20	—	—	20
Delay for VS* fall to BUSY* rise	Tbh	—	—	20	—	—	20

Data Read Cycle

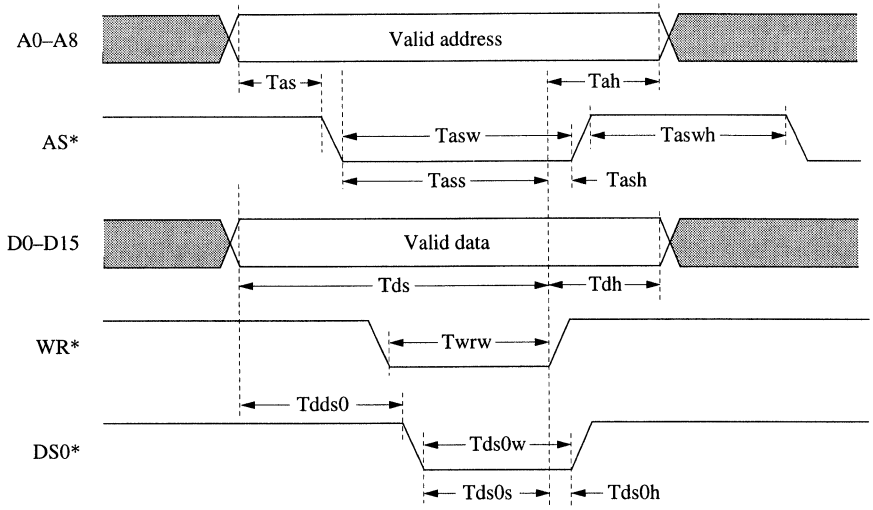


Note: DS1* represents upper word selection, and DS0* represents lower word selection

Units: ns

Item	Symbol	IP90C01			IP90C01-LC		
		Min.	Typ.	Max.	Min.	Typ.	Max.
A0 to A8 set-up time to AS* fall	Tads	2	—	—	2	—	—
AS* pulse width (low)	Tasw	60	—	10 µs	60	—	10 µs
AS* pulse width (high)	Taswh	50	—	—	50	—	—
A0 to A8 hold time from AS* rise	Tadh	7	—	—	7	—	—
Delay time from AS* rise until D0 to D15 data becomes invalid	Tasd	5	—	55	5	—	55
Delay time from DS* fall until D0 to D15 data becomes valid	Tdbd	5	—	55	5	—	55
DS0*, DS1* pulse width	Tdsw	40	—	10 µs	40	—	10 µs
Delay time from DS* rise until D0 to D15 Hi-Z mode	Tdbz	5	—	20	5	—	20

Register Setting Cycle

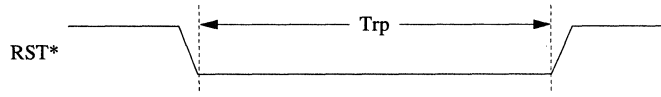


The above timing chart represents a register write cycle based on the rise of WR*. In the IP90C01's internal circuitry, AS*, DS0*, and WR* are treated the same, and the cycle is defined by the rising edge of whichever signal rises first.

Units: ns

Item	Symbol	IP90C01			IP90C01-LC		
		Min.	Typ.	Max.	Min.	Typ.	Max.
A0 to A8 set-up time to AS* fall	Tas	10	—	—	10	—	—
A0 to A8 hold time from WR* rise	Tah	2	—	—	2	—	—
AS* pulse width (low)	Tasw	50	—	1000	50	—	1000
AS* pulse width (high)	Taswh	50	—	—	50	—	—
AS* fall set-up time to WR* rise	Tass	50	—	—	50	—	—
AS* rise hold time from WR* rise	Tash	1	—	—	1	—	—
D0 to D15 set-up time to WR* rise	Tds	50	—	—	50	—	—
D0 to D15 hold time from WR* rise	Tdh	5	—	—	5	—	—
WR* pulse width	Twrw	50	—	1000	50	—	1000
DS0* pulse width	Tds0w	50	—	1000	50	—	1000
D0–D15 set-up time to DS0* fall	Tdds0	0	—	—	—	—	—
DS0* fall set-up time to WR* rise	Tds0s	50	—	—	50	—	—
DS0* rise hold time from WR* rise	Tds0h	0	—	—	0	—	—

System Reset Cycle



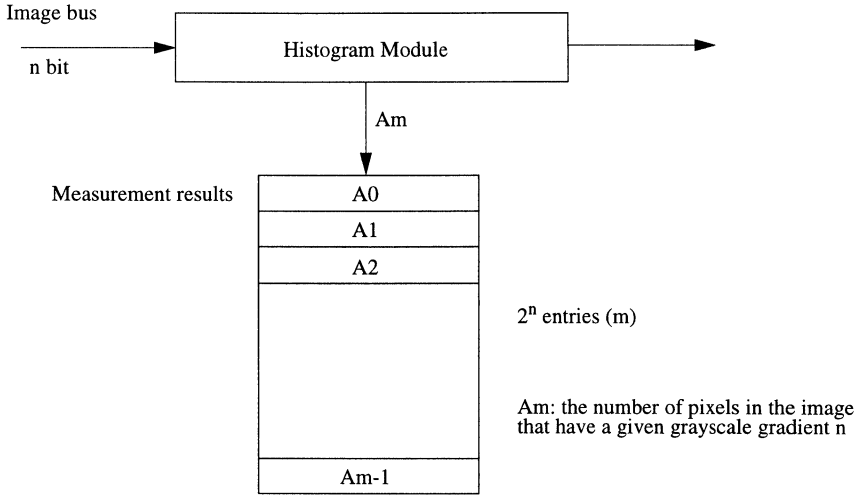
Units: ns

Item	Symbol	IP90C01			IP90C01-LC		
		Min.	Typ.	Max.	Min.	Typ.	Max.
RST* pulse width	Trp	200	—	—	300	—	—

Section 6. Sample Applications

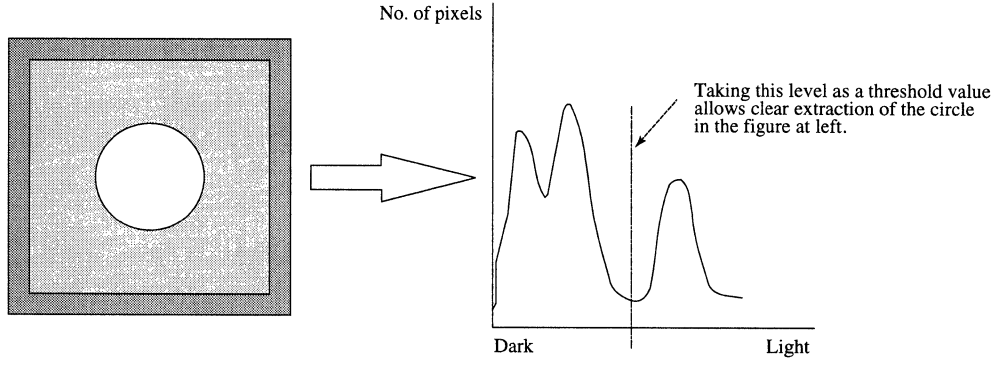
6.1 Basic Histogram

Histogram processing measures the grayscale gradient value of every pixel in the target image.



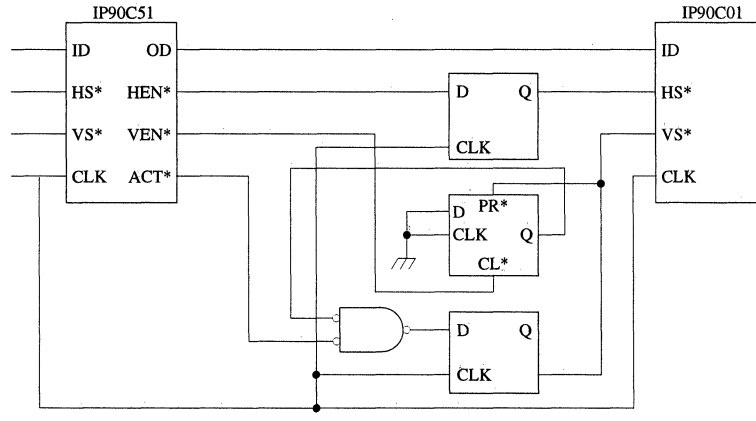
Sample Application

This process makes it possible to obtain s-tandards for quantifying the grayscale values of an image, for use in image conversion (such as binary conversion) or enhancement processing.

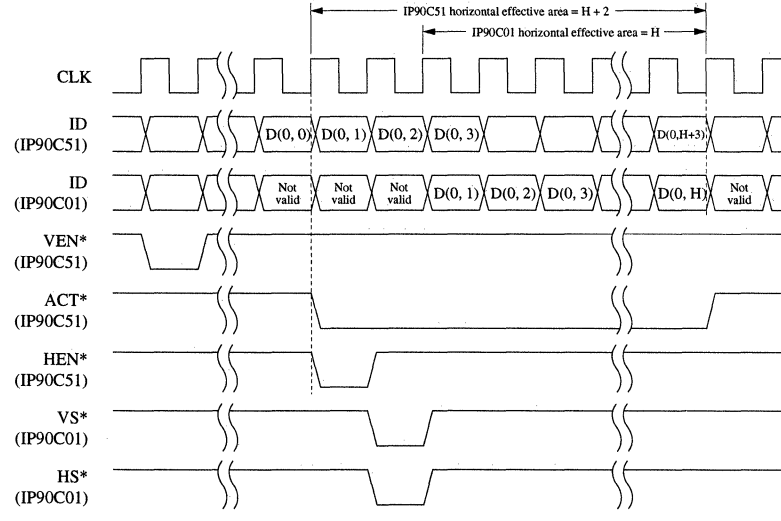


6.2 Sample Connections to the IP90C51 (non-interlaced mode)

The IP90C01 chip can be connected to an IP90C51 chip to provide simple control functions.



(a) Sample Circuit



(b) Signal Timing

Figure (a) shows a sample connection in non-interlaced mode, and Figure (b) shows the signal timing for the example. The design should provide for pulse VEN* and HEN* output from the IP90C51.

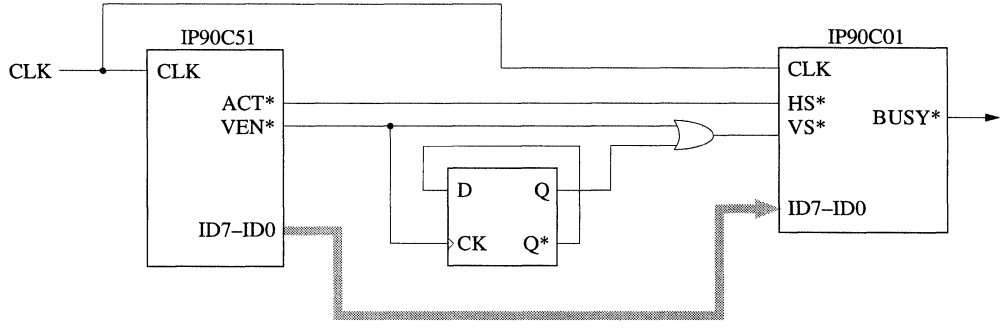
If the horizontal and vertical dimensions of the IP90C01 processing area are H and V respectively, then the Hsize and Vsize settings for the IP90C51 are H+2 and V+1. (An additional line of HS* pulses is required to accommodate a dummy cycle. See the system timing charts in Section 4.6.) Adjust the image input timing to suit the circuits used. For other IP90C51 settings, see the IP90C51's technical documentation. The above circuit is for reference only, and should not be used in applications without carefully considering all relevant conditions, connections, and timing requirements.

6.3 Histogram Processing of Interlaced Image Input: 1

Adding the circuits described below allows the IP90C01 to be connected to an IP90C51 LSI chip to handle interlaced image input. The additional circuits mask the second VS*, which enables continuous histogram processing.

Example: 1 field, 5x5 pixels

- IP90C01 register settings:
  - H register: 0100h
  - V register: If VS* and HS* arrive simultaneously, 000Ah
    If VS* and HS* do not arrive simultaneously (IP90C51 HOFS≥1), 000Bh
    (For more information, see Section 4.9 of the IP90C01 technical documentation.)
- IP90C51 register settings:
  - SMOD register: 01h, HWID register: 0000h
    ⇒ Outputs a pulse of 1 clock-cycle width from its ACT* pin, and inputs this pulse at the IP90C01 HS* pin.
  - VWID0,1 registers:
    To the VWID register corresponding to the field with the input VS*: 0005h
    To the VWID register corresponding to the field with the masked VS*: 0006h
    ⇒ This is because the IP90C01 requires a dummy line.
- The third-field VS* is input to the IP90C01, allowing BUSY* to rise and processing results to be read.

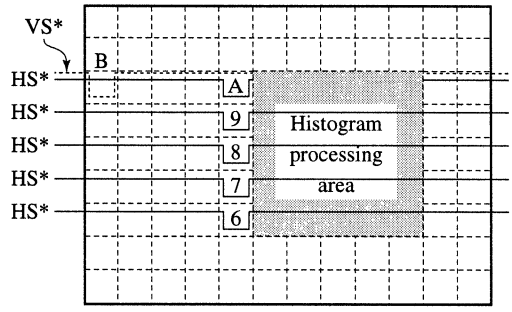


Note: The above sample circuits are provided for reference only. Thoroughly study all circuits before attempting to use them in actual applications.

Interlace Mode Histogram Processing Example 1: IP90C01 VS* and HS*

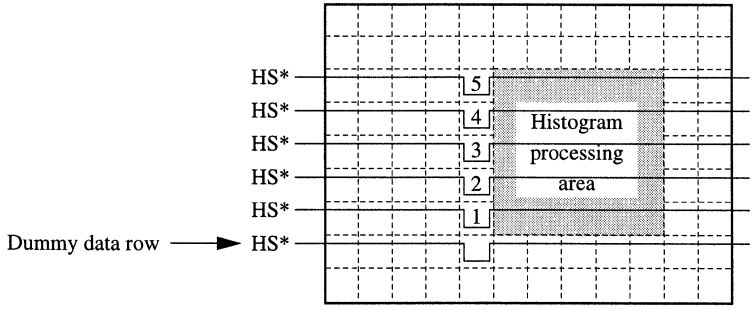
(Note: The IP90C51 HOFS register is set to 4, and VS* and HS* are not input simultaneously.)

First field (VS* input)

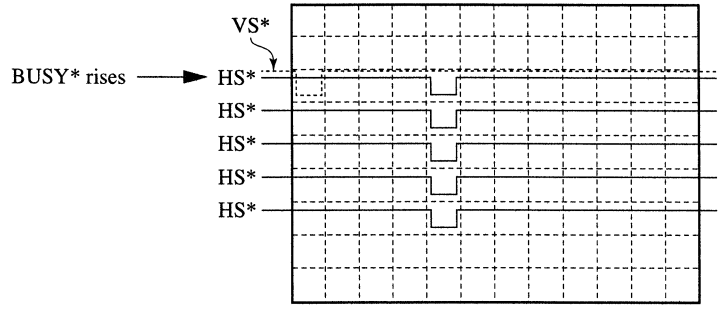


Note: The values B-1 in the illustration represent IP90C01 V counter values.

Second field (VS* masked, dummy data row input)



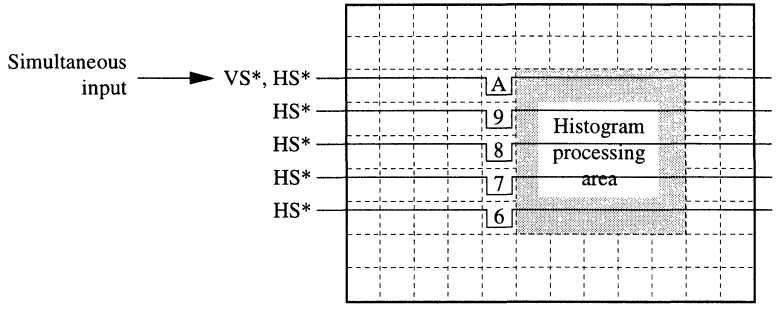
Third field (VS* input, BUSY* rises from low to high)



Interlace Mode Histogram Processing Example 1: IP90C01 VS* and HS* signals.

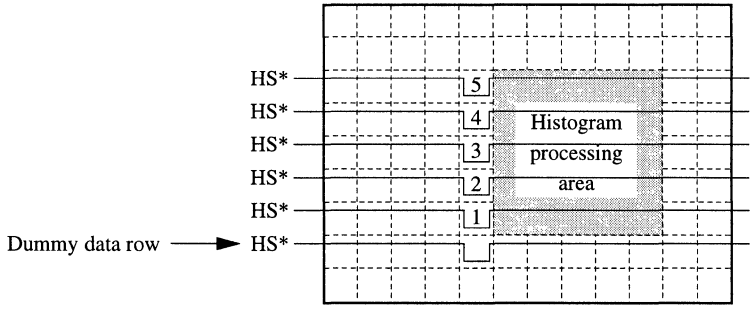
(Note: The IP90C51 VEN* and ACT* signals are adjusted for simultaneous input.)

First field (VS* input)

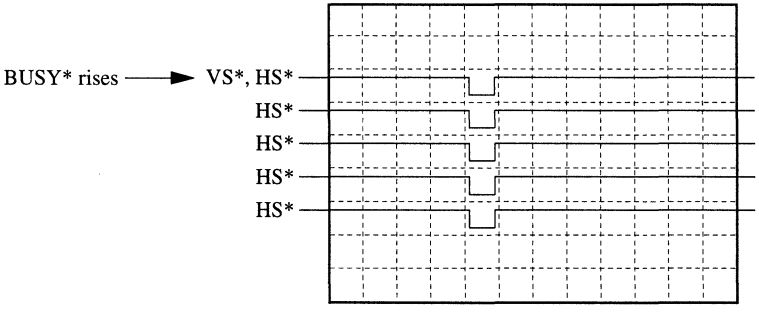


Note: The values A–1 in the illustration represent IP90C01 V counter values.

Second field (VS* masked, dummy data row input)



Third field (VS* input, BUSY* signal rises from low to high)

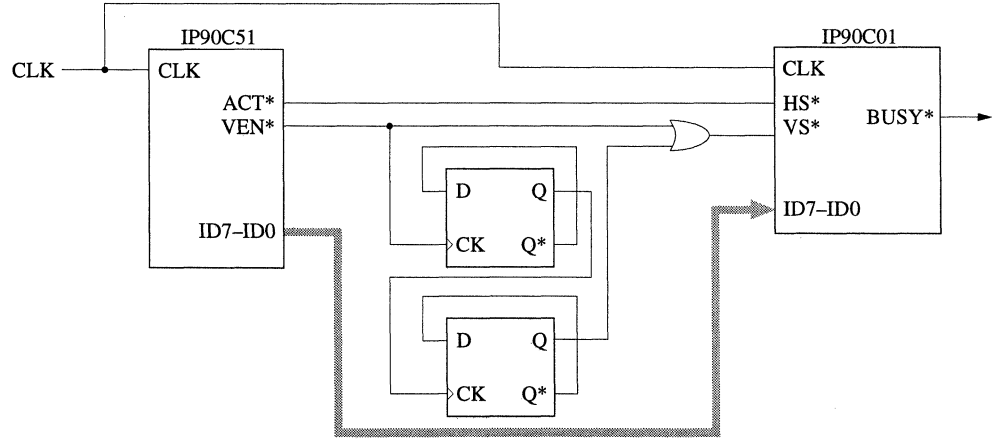


6.4 Histogram Processing of Interlaced Image Input: 2

Adding the circuits described below allows the IP90C01 to be connected to an IP90C51 chip to handle interlaced image input. The additional circuits mask the second and third VS* pulses, enabling continuous histogram processing.

Example: 1 field, 5x5 pixels

- IP90C01 register settings:
  - H register: 0100h
  - V register: If VS* and HS* arrive simultaneously, 000Ah
    If VS* and HS* do not arrive simultaneously (IP90C51 HOFS ≥ 1), 000Bh
    (For more information, see Section 4.9 of the IP90C01 technical documentation.)
- IP90C51 register settings:
  - SMOD register: 01h, HWID register: 0000h
    ⇒ Outputs a pulse of 1 clock-cycle width from its ACT* pin, and inputs this pulse at the IP90C01 HS* pin.
  - VWID0,1 registers: 0005h
- The third-field VS* is masked, so that the IP90C01 enters a dummy data line instead of VS*.
- The fourth-field VS* is input to the IP90C01, allowing BUSY* to rise and histogram processing results to be read.

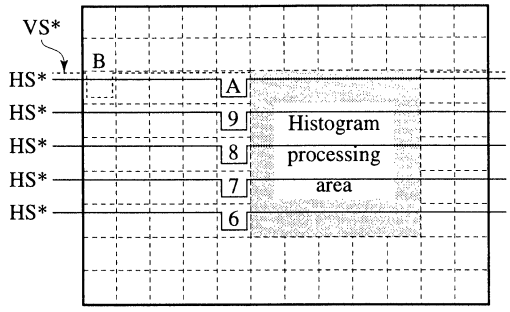


Note: The above sample circuits are provided for reference only. Thoroughly study all circuits before attempting to use them in actual applications.

Interlace Mode Histogram Processing Example 2: IP90C01 VS* and HS*

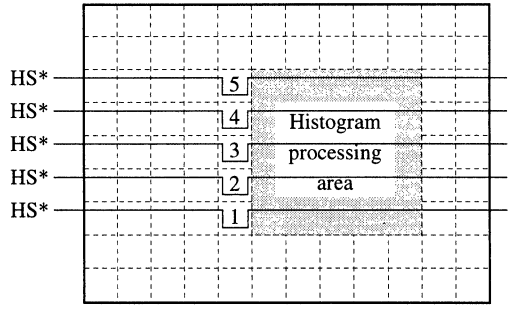
(Note: The IP90C51's HOFS register is set to 4, and VS* and HS* are not input simultaneously.)

First field (VS* input)

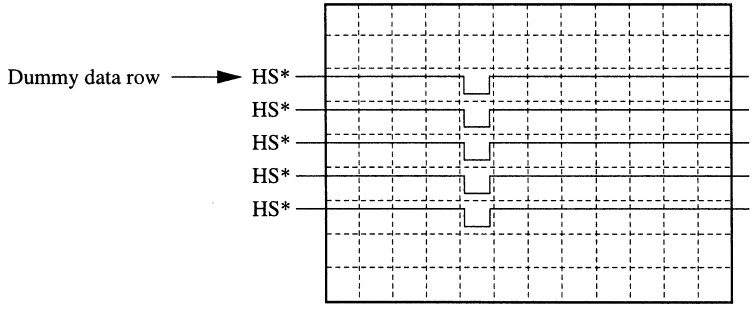


Note: The values A–1 in the illustration represent IP90C01 V counter values.

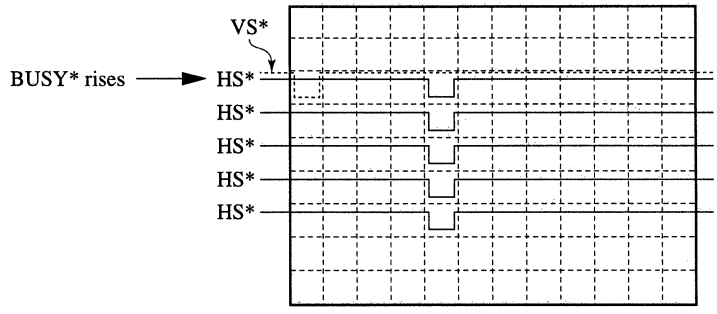
Second field (VS* masked)



Third field (VS* masked, dummy data row input)



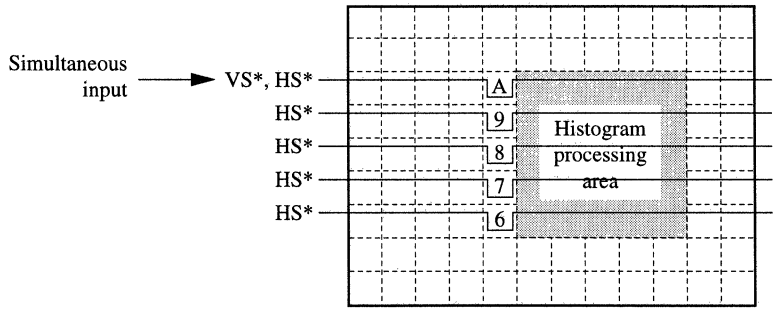
Fourth field (VS* input, BUSY* signal rises from low to high)



Interlace Mode Histogram Processing Example 2: IP90C01 VS* and HS* signals

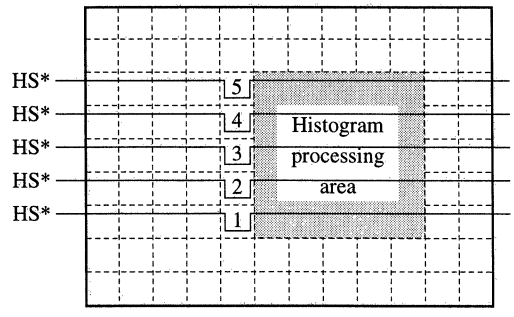
(Note: The IP90C51 VEN* and HEN* signals are adjusted for simultaneous input.)

First field (VS* input)

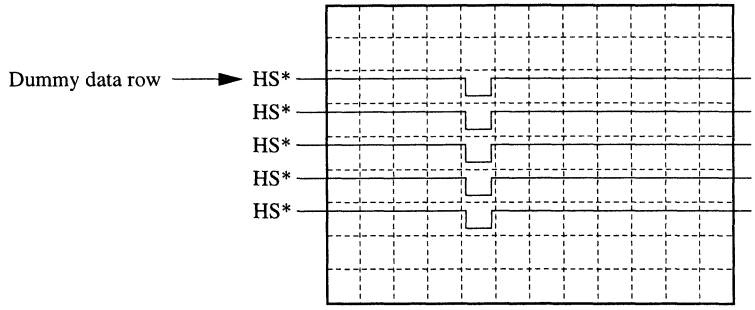


Note: The values A–9 in the illustration represent IP90C01 V counter values.

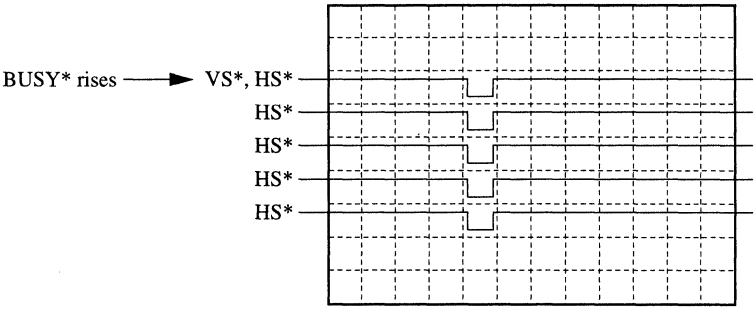
Second field (VS* masked)



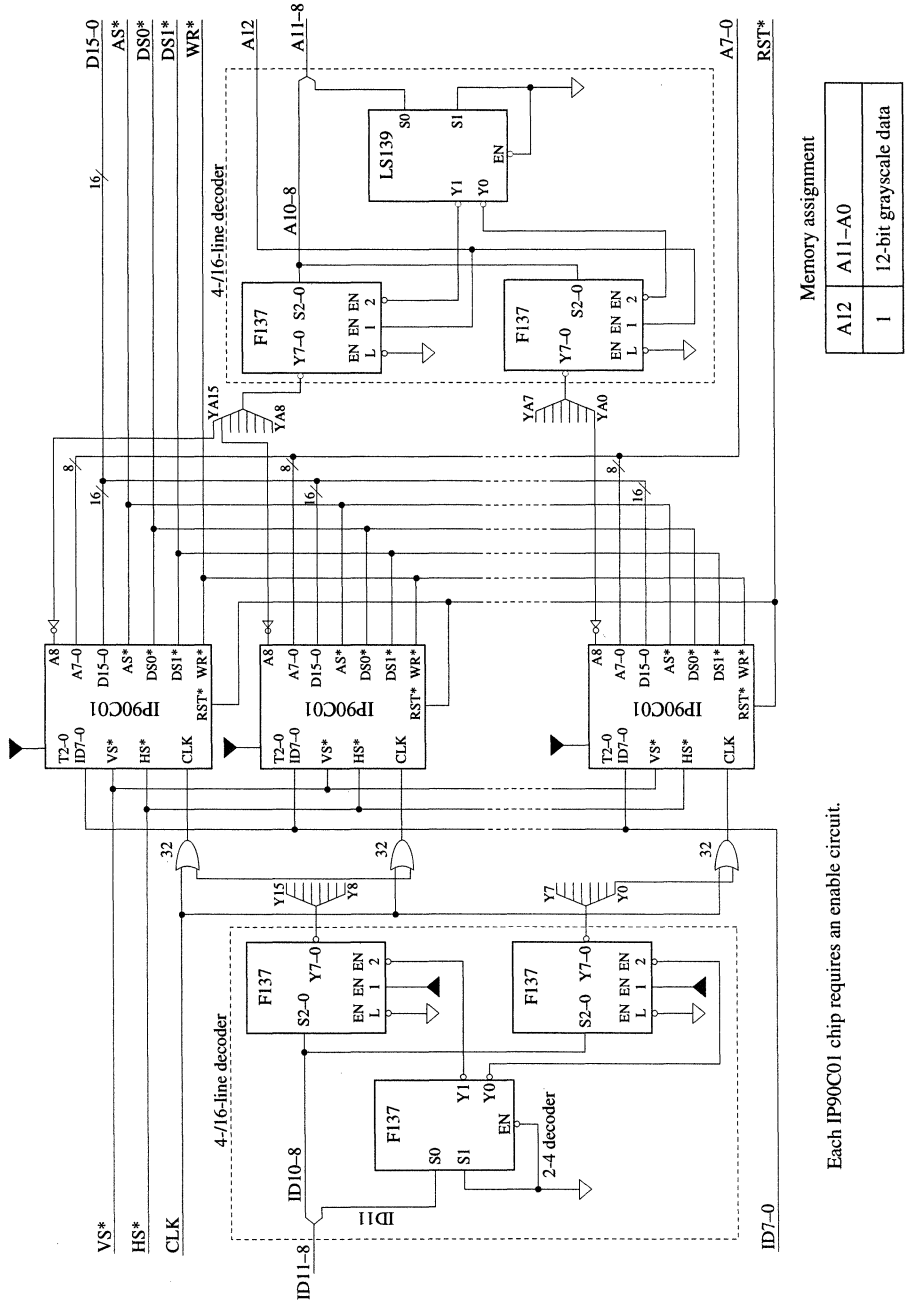
Third field (VS* masked, dummy data row input)



Fourth field (VS* input, BUSY* signal rises from low to high)



6.5 Circuit Diagram: 12-Gradient Grayscale Image Data



6.6 Pixel Control During Histogram Processing with Clock Signal Stopped

1. Histogram Start Sequence

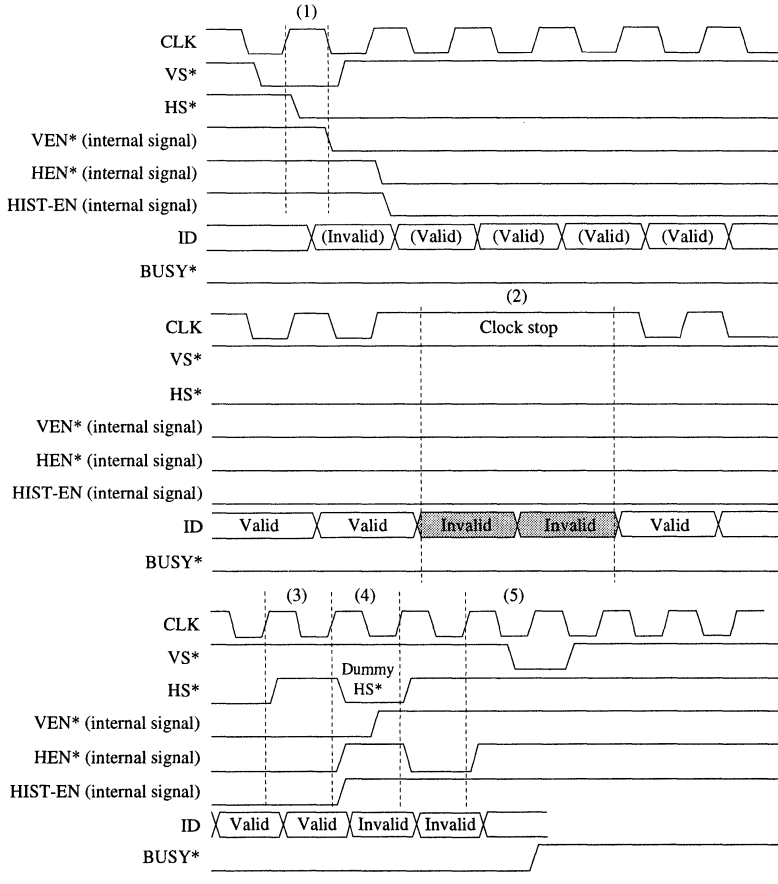
1. Set the H and V registers to 0001h.
2. Set the mode register to histogram processing mode (0002h).
3. Low VS* (pulse signal) and HS* (level signal) to start histogram processing (1).

2. Histogram Pixel Control Sequence with Clock Signal Stopped

4. Hold CLK high for all pixels not to be processed (2).

3. End Histogram Processing, Read Results

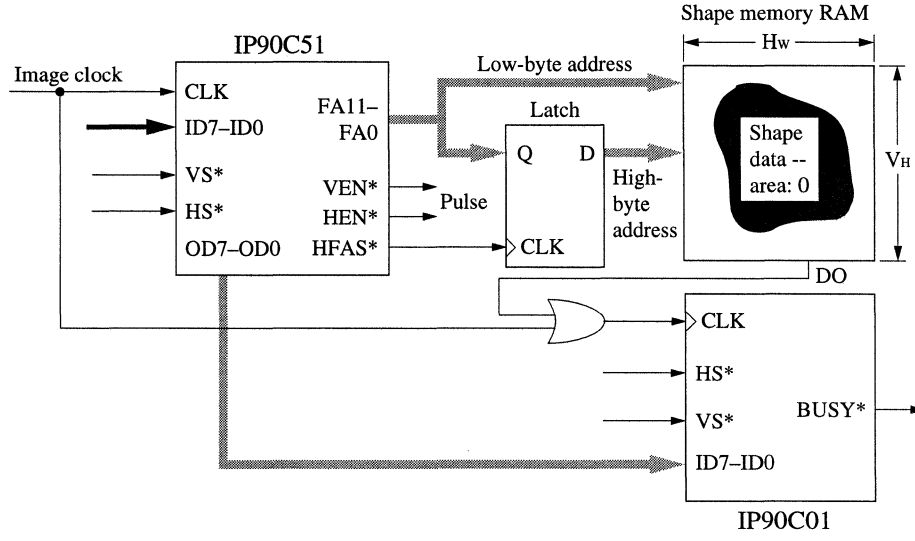
5. Raise HS* 1 clock count before the final valid ID (3).
6. Input a dummy HS* (one or more clock counts) to cancel the internal VEN* status (4).
7. Input VS* to cancel BUSY* status (5).
8. Read data (same as normal processing sequence).



6.7 Histogram Processing of Designated Shapes

The IP90C01 and IP90C51 can be used to process images having any shape.

As shown in the following illustration, mask data for the desired shape is stored in RAM, and the RAM output data is used to control the clock.



Operating Description

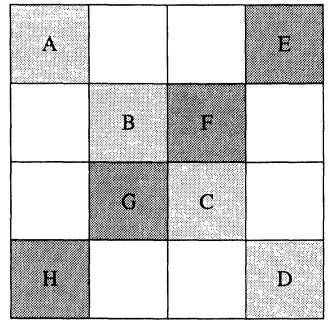
Shape mask data previously stored in RAM is read using the IP90C51 frame address generator function. The IP90C01 input clock can be controlled by the RAM output data, so that while the mask data stored in RAM is being read, the IP90C01 input clock is stopped at high level, and histogram processing is skipped. Thus, by applying histogram processing only to the non-masked RAM data, histogram processing can be performed on selected areas of any shape. (See Section 6.6 for information about clock stopping.)

Note: The above example is not an actual sample circuit, and is intended for reference only. The diagrams above are conceptual diagrams only. Be sure to thoroughly study all circuits before attempting to use them in actual applications.

6.8 Histogram Processing of Separate Local Areas

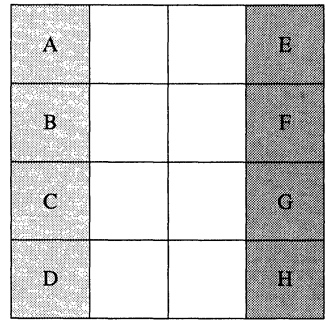
In interlace mode, a screen can be divided into multiple local areas, and histogram processing applied to each of these areas separately. An IP90C51 can be set up to partition the image signal into even- and odd-numbered fields, so that histogram processing can then be applied to each local area.

Example 1: Histogram processing of 8 of 16 areas



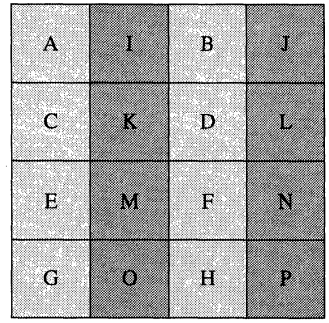
First field: A, B, C, D
Second field: E, F, G, H
IP900C51 x 1
IP90C01 x 1

Example 2: Histogram processing of 8 of 16 areas



First field: A, B, C, D
Second field: E, F, G, H
IP900C51 x 1
IP90C01 x 1

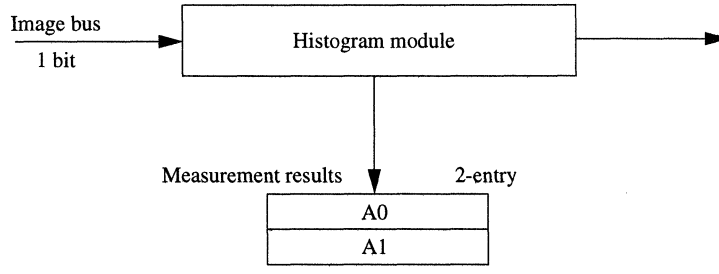
Example 3: Histogram processing of all 16 areas



First field: A, C, E, G
First field: B, D, F, H
Second field: I, K, M, O
Second field: J, L, N, P
IP900C51 x 2
IP90C01 x 2

6.9 Area Measurement

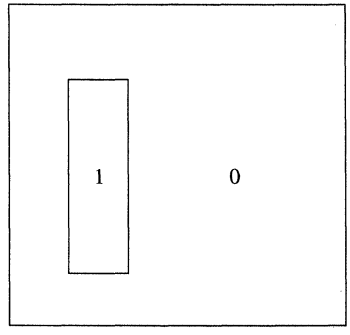
Histogram processing of binary images can be used to measure the area of the subject image.



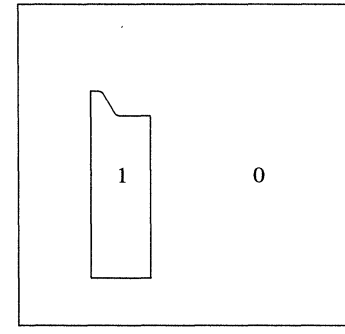
Sample Applications

Histogram processing can allow detection of form defects, surface scratches, and flaws, as well as presence or absence of items, by measuring the surface area of the subject item and comparing the result to a predetermined standard value.

Binary images



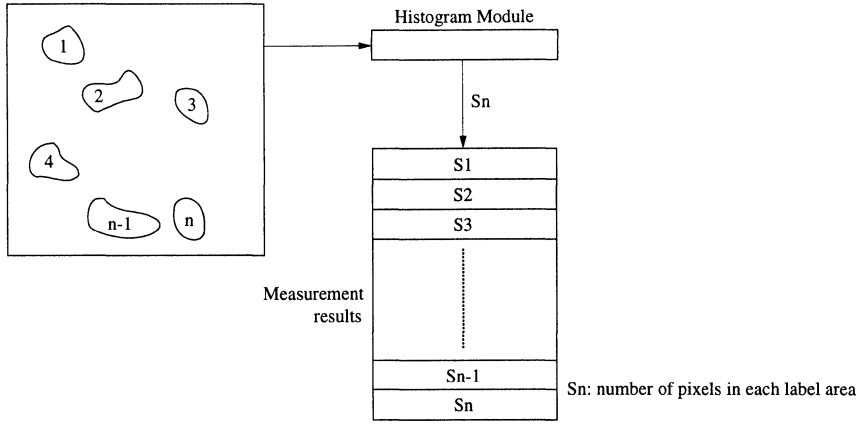
(Good item)



(Flaw or missing item)

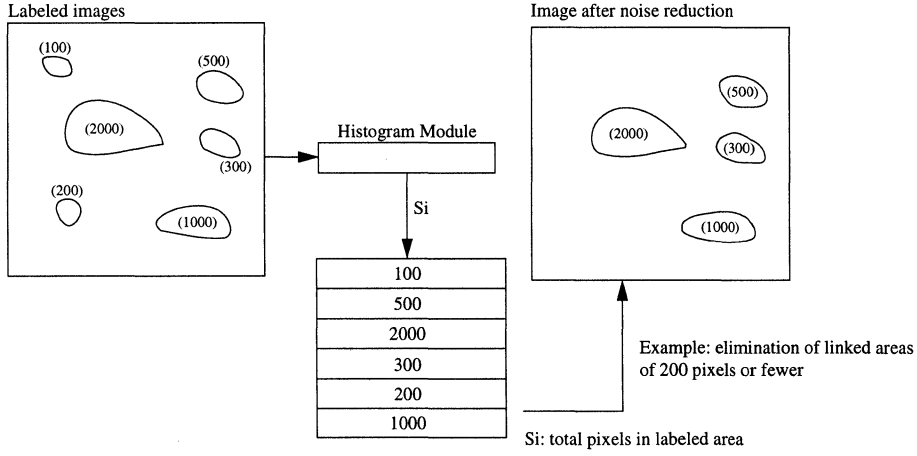
6.10 Area Measurement of Linked Areas

Histogram measurement of labeled images (images after label processing, in which different gradient values are assigned to each linked area) can be used to measure the surface area (in pixels) of each linked area in the image.



Sample Applications

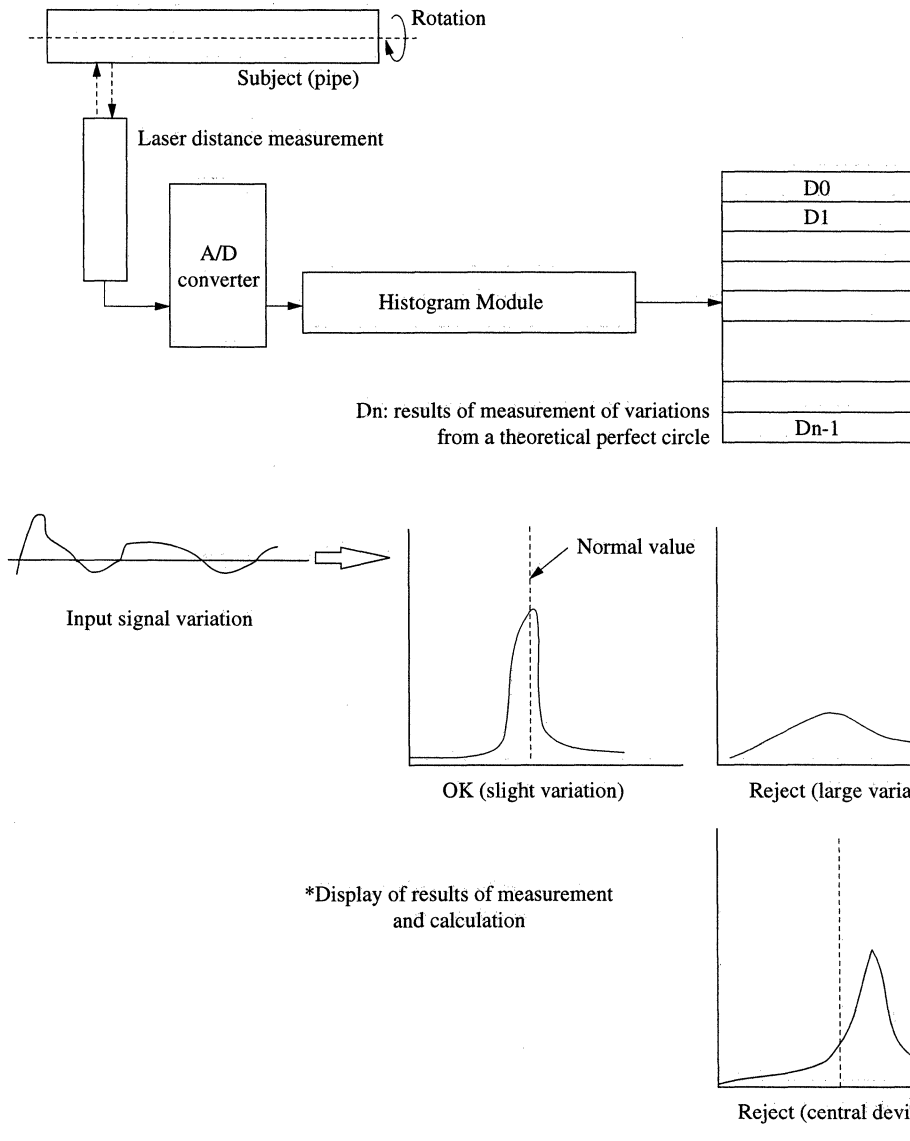
Histogram processing can enable analysis of particle size and definition in metal composition, counting spherical bodies, noise reduction by threshold area value, etc.



6.11 Sample Applications Outside of Image Processing

The IP90C01 is a general-purpose statistical histogram processor, and as such can naturally be used in applications other than image processing.

Measurement of Distribution of External Pipe Diameters



SMI ASSP

Image Processing LSI Series

IP90C05A

Projection Processor (proJ)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 2.2

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Overview

1.1 Functional Outline and Features

The IP90CO5A performs horizontal and vertical gray-level projection processing of 8-bit grayscale raster-scanned images, and simultaneously stores the results in its horizontal and vertical projection memories. The IP90C05A can process areas of up to 512 x 512 pixels; this area can be expanded by using multiple chips, since the IP90C05A uses 18-bit-wide memory.

- Projection processing function
  - -- Simultaneously executes vertical and horizontal gray-level projection processing of raster-scanned input images.
  - -- Performs projection processing of 8-bit grayscale image data (256 gray levels).
  - -- Contains two 18-bit x 512-pixel memories.
  - -- Mask processing can replace any pixel with any mask data.
  - -- Interlace/non-interlace compatible.
  - -- The processing area can be expanded by using multiple chips.

- Maximum clock frequency: 30 MHz (IP90C05A-HS)
  20 MHz (IP90C05A without rank code)

  Note: The clock duty ratio of the IP90C05A-HS is not 50% due to its high-speed operation. Please see Section 6.5, "AC Characteristics."

- External interface
  - -- The CPU data bus is 8/16 bits wide.

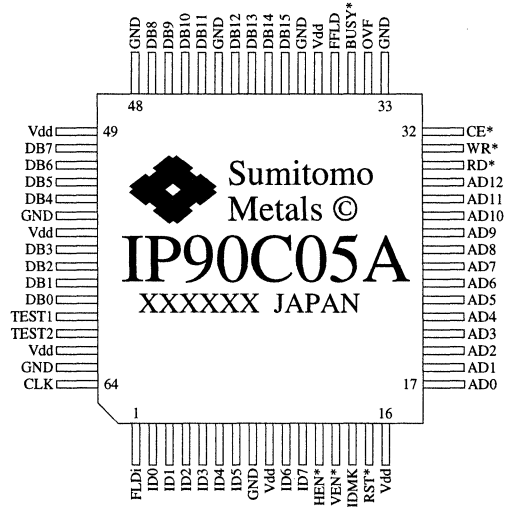
- Other
  - -- Process: CMOS
  - -- Power supply: 5V single power supply
  - -- Input/output level: TTL compatible
  - -- Package: 64-pin QFP
    Molded section (14 mm square; pin pitch = 0.8 mm)

- Application examples
  - -- Detecting positions
  - -- Recognizing shapes
  - -- Detecting centers of gravity
  - -- Recognizing box frames
  - -- Performing multiwindow-based processing when connected to the IP90C51

Section 2: Pin Assignment

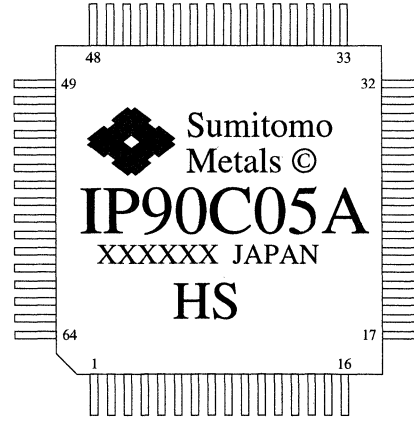
2.1 Pin Configuration Diagram

IP90C05A



64-pin QFP package (molded section = 14 mm square; pin pitch = 0.8 mm)

IP90C05A-HS



64-pin QFP package (molded section = 14 mm square; pin pitch = 0.8 mm)

Note: The IP90C05A and IP90C05A-HS have the same pin assignments. The only difference in their appearances is the marking stamped on each.

Table of Pin Assignments

Pin No.	Name
1	FLDi
2	ID0
3	ID1
4	ID2
5	ID3
6	ID4
7	ID5
8	GND
9	Vdd
10	ID6
11	ID7
12	HEN
13	VEN
14	IDMK
15	RST*
16	Vdd
17	AD0
18	AD1
19	AD2
20	AD3
21	AD4
22	AD5

Pin No.	Name
23	AD6
24	AD7
25	AD8
26	AD9
27	AD10
28	AD11
29	AD12
30	RD*
31	WR*
32	CE*
33	GND
34	OVF
35	BUSY*
36	FFLD
37	Vdd
38	GND
39	DB15
40	DB14
41	DB13
42	DB12
43	GND
44	DB11

Pin No.	Name
45	DB10
46	DB9
47	DB8
48	GND
49	Vdd
50	DB7
51	DB6
52	DB5
53	DB4
54	GND
55	Vdd
56	DB3
57	DB2
58	DB1
59	DB0
60	TEST1
61	TEST2
62	Vdd
63	GND
64	CLK

Notes:
1. The IP90C05A and IP90C05A-HS have the same pin assignments. The only difference in their appearances is the marking stamped on each.
2. An asterisk following a signal name indicates negative logic.

2.2 Pin Descriptions

Pin group	Symbol	No. of Pins/Type		Function	Description
Image bus	CLK	1	I	Image Clock	
	ID0–ID7	8	I	8-bit image data input	MSB: ID7; LSB: ID0
	HEN*	1	I	Horizontal data enable signal	
	VEN*	1	I	Vertical data enable signal	
	IDMK	1	I	Masking signal	Masks image data: When this pin is high, data is replaced by the mask value (Note 1).
	FLDi	1	I	Field discriminator signal	In interlace mode, this pin must input the discriminator signal of the currently processed field (Note 2). High: odd field; Low: even field.
Control bus	WR*	1	I	Write signal	
	RD*	1	I	Read signal	
	CE*	1	I	Chip enable	
	AD0–AD12	13	I	Address bus	MSB: AD12, LSB: AD0
	DB0–DB15	16	I/O	Data bus	MSB: AD15, LSB: DB0
	FFLD	1	O	Field discriminator signal retention output at the start of processing	In interlace mode, this pin outputs the discriminator signal of the field to which the first line belongs (Note 3). This pin is high in non-interlace mode and after a reset.
	OVF	1	O	Overflow signal	This pin is high whenever an overflow occurs (Note 4), and is low after a reset.
	BUSY*	1	O	Processing execution flag	This pin is low during processing and high after a reset.
	RST*	1	I	Reset	
Other	TEST1, 2	2	I	Test pin	This signal must remain high when in use (Note 5).
Power supply	Vdd	6	PW	5V	
and GND	GND	7	PW	Ground	
		54			

Note 1: Image data replacement can be controlled pixel by pixel. For details, see Section 4.4, "Mask-Value Registers."

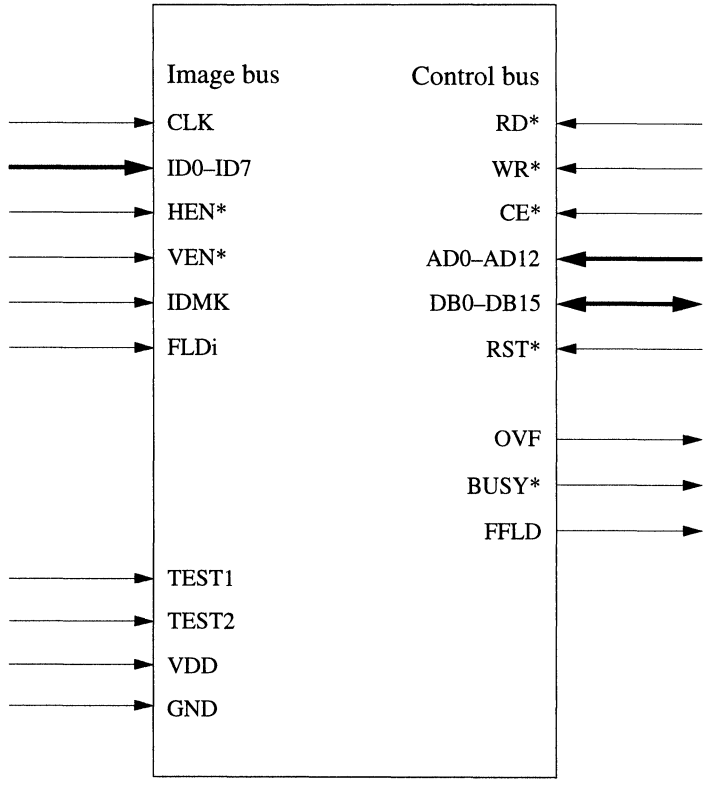
Note 2: Inputs a high signal in non-interlace mode.

Note 3: This pin retains the FLDi signal for the field (1st field) input immediately after projection processing is initiated in interlace mode. If this FLDi signal is high, the FFLD signal also goes high ; if FLDi is low, the FFLD signal also goes low. For details, see "7) FFLD Pin Output" in Section 5.3.

Note 4: If the processing range in the horizontal or vertical direction exceeds 512 pixels, the device may run out of memory, driving OVF high. For processing in only one direction, see Section 5.4, "Overflow."

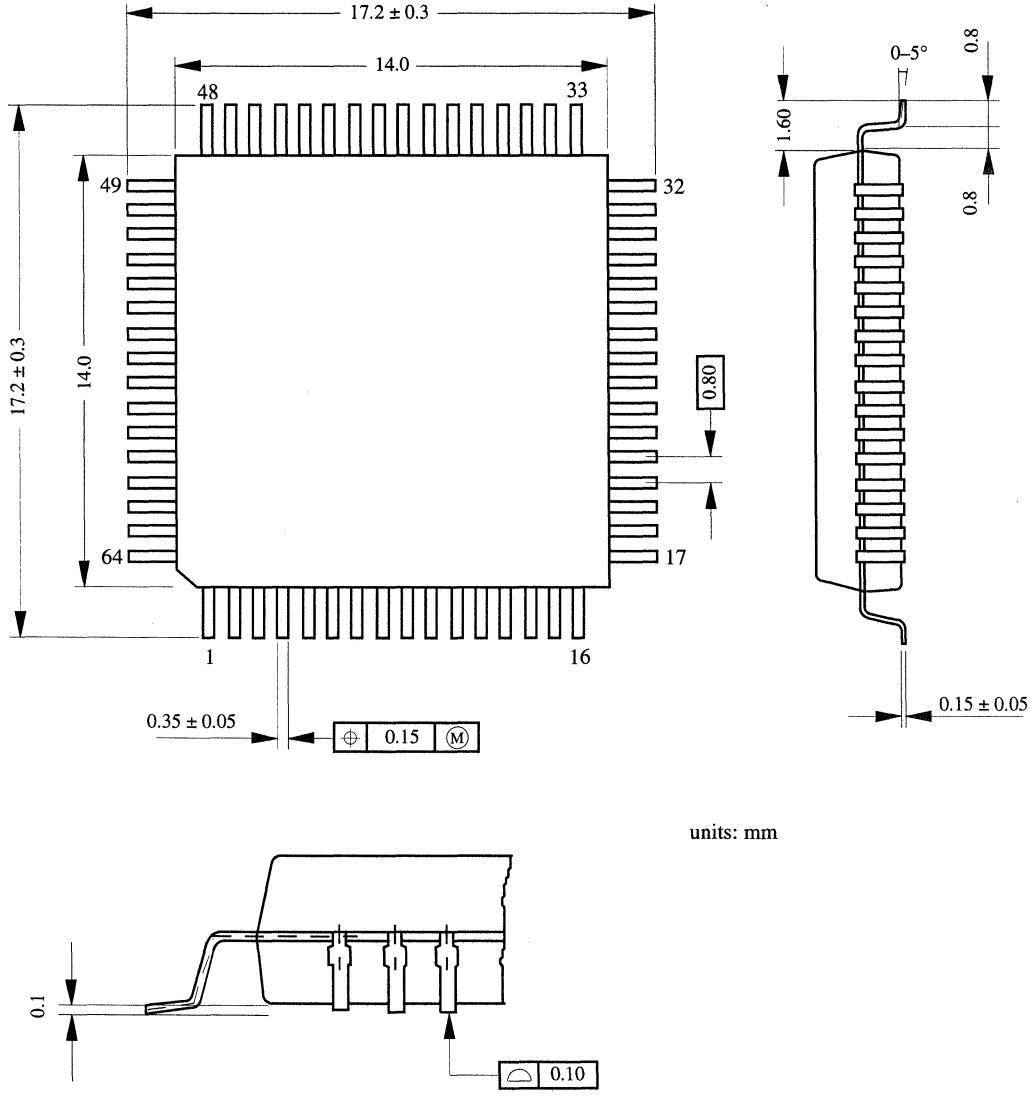
Note 5: This pin is used to test the device's internal logic, and must be kept high during normal use.

2.3 Logic Pin Diagram



2.4 Package Dimensions

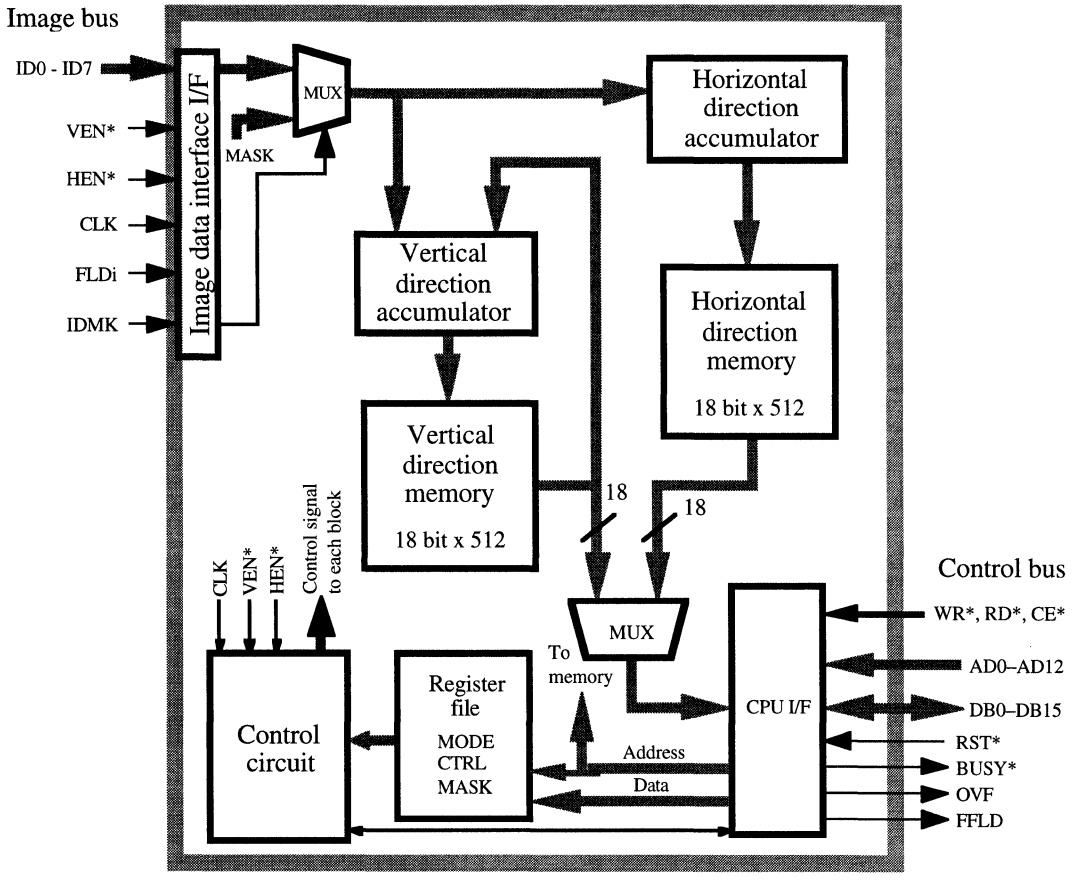
The IP90C05A has a 64-pin plastic QFP package, as shown below.



units: mm

Section 3: Block Diagram

3.1 Block Diagram



Note: This block diagram is for functional description purposes only, and does not show all the IP90C05A's functions. For details about functionality, timing, etc., refer to other sections of this manual.

Section 4. Registers

4.1 Outline of Registers

All registers are write-only.

1) Register Outlines

Address (hex)	Register	Symbol	Width bit	Description
0000h	Mode register	MODE	8	Sets projection processing mode
0004h	Control register	CTRL	8	Sets control, etc. during execute and memory read cycles
0008h	Mask-value register	MASK	8	Sets the mask values

2) Functional Outline and Bitmap

		Bit								
		15.....8	7	6	5	4	3	2	1	0
Data bus		DB15–DB8	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
Mode register	Bit name	—	—	—	—	—	vproj	hproj	vmstr	scan
	Outline	00h	0	0	0	0	Execute processing in the vertical direction	Execute processing in the horizontal direction	Mode in which data is stored in the vertical projection memory.	Scanning mode
Control register	Bit name	—	—	—	—	mdru	fl	hmru	exec	sftrst
	Outline	00h	0	0	0	Units from which memor y data is read	Field to which the start line belongs	Horizontal memory read control.	Execute	Software reset
Mask-value register		00h	Arbitrary data							

Note 1: All registers can be cleared to 0 by resetting the hardware or software.

Note 2: The software reset bit (sftrst) is automatically set to 0 after software reset.

Note 3: If the execute bit (exec) and software reset bit (sftrst) are simultaneously set to 1, the device is reset, because software reset takes priority.

4.2 Mode Register

Use the mode register to set the IP90C05A's processing mode.

	b7	b6	b5	b4	b3	b2	b1	b0	Address
MODE	0	0	0	0	vproj	hproj	vmstr	scan	0000h

- This register resets to 00h.
- Bits b7, b6, b5, and b4 must always be set to 0.

b0: scan bit

This bit determines whether only one field of image data (non-interlace mode) or two (interlace mode) are processed.

- b0 = 0: Non-interlace mode (one field of image data is processed). This is the default.
- b0 = 1: Interlace mode (two fields of image data are processed).

b1: vmstr bit

During interlace mode (scan = 1), this bit determines whether the results of vertical projection processing are stored in the vertical projection memory for two fields of image data (i.e., one frame of data) or for one field of data. This bit has no effect in non-interlace mode (scan = 0). For details, see Section 5.3. "Projection Processing in Interlace Mode."

- b1 = 0: Two fields of processed data (one frame of data) are stored. This is the default.
- b1 = 1: One field of processed data is stored.

The table below shows the relationship of the scan and vmstr bits to the input image scan mode (non-interlaced or interlaced).

Mode register / Input image					
	scan	0	0	1	1
	vmstr	0	1	0	1
Non-interlace		One frame of image data (one field) is processed.		Note 1	Note 2
Interlace		Only the 1st field is processed.		One frame of image data (two fields) is processed.	Note 3

Note 1: Two frames of image data (two fields of data) are processed, though only 256 lines of each field can be processed. Watch for overflows when using this mode.

Note 2: Do not use this mode since it does not affect processing.

Note 3: This is a split-field store mode, which allows the results of horizontal projection processing to be stored in separate locations for each field. Do not use this mode during normal processing. For details, see Section 5.3, "Projection Processing in Interlace Mode."

b2: hproj bit

This bit enables execution of horizontal projection processing.

- b2 = 0: Executes horizontal projection processing (this is the default).
- b2 = 1: Does not execute horizontal projection processing (Note 4).

b3: vproj bit

This bit enables execution of vertical projection processing.

- b3 = 0: Executes vertical projection processing (this is the default).
- b3 = 1: Does not execute vertical projection processing (Note 5).

Note 4: If projection processing is executed in only the vertical direction, the device enters an expanded vertical processing area mode. In this situation, since the device contains 18-bit x 512 memory, the maximum processing area expands to 512 pixels horizontally by 1024 pixels vertically. (Make sure the vertical processing area does not exceed 1024 pixels.) A sample expanded vertical processing area is shown in Section 7.2, "Expanded Processing Range."

Note that in this mode, even if the processing area exceeds 512 or 1024 pixels in the vertical direction, the OVF signal does not output an overflow status. However, it does output an overflow status when the area exceeds 512 pixels in the horizontal direction.

Note 5: If projection processing is executed in only the horizontal direction, the device enters an expanded horizontal processing area mode. In this situation, since the device contains 18-bit x 512 memory, the maximum processing area expands to 1024 pixels horizontally by 512 pixels vertically. (Make sure the horizontal processing area does not exceed 1024 pixels.) A sample expanded horizontal processing area in the horizontal processing area is shown in Section 7.2, "Expanded Processing Range."

Note that in this mode, even if the processing area exceeds 512 or 1024 pixels in the horizontal direction, the OVF signal does not output an overflow status. However, it does output an overflow status when the area exceeds 512 in the vertical direction.

4.3 Control Register

Use the control register to set software reset, execution, and controls during a memory read.

	b7	b6	b5	b4	b3	b2	b1	b0	Address
CTRL	0	0	0	mdru	fl	hmru	exec	sftrst	0004h

- This register resets to 00h.
- Bits b7, b6, b5, and b4 must always be set to 0.

b0: sftrst bit

Writing 1 to this bit executes the software reset. This bit is automatically cleared to 0 after reset, so there is no need to write 0 to it.

b1: exec bit

Writing 1 to this bit readies the device for execution. This bit is automatically cleared to 0 after processing ends.

- b1 = 0: Idle (this is the default).
- b1 = 1: Execute.

b2: hmru bit

This bit affects horizontal projection memory address mapping in interlace mode (scan = 1). It determines how the results of horizontal projection processing are read from the horizontal projection memory: sequentially in order of coordinate address, or sequentially in order of input address, field by field.

This bit has no effect in non-interlace mode. For details, see Section 5.5, "Reading Processing Results."

- b2 = 0: In the order of the coordinates (this is the default).
- b2 = 1: In the order of the input address.

b3: fl bit

This bit specifies the field to which the start line belongs. When reading the order of the coordinates (hmru = 0) in interlace mode (scan = 1), this bit specifies whether the field of the processing area's first line is even or odd.

This bit has no effect in non-interlace mode (scan = 0), or when reading the order of the inputs in interlace mode (scan = 1, hmru = 1). For details, see Section 5.5, "Reading Processing Results."

- b3 = 0: Even field (this is the default).
- b3 = 1: Odd field.

b4: mdru bit

When reading 18-bit projection processing results from memory, this bit sets the units used: in word units (16 bits) or byte units (8 bits).

- b4 = 0: In word units (this is the default).
- b4 = 1: In byte units (Note 1).

Note 1: When accessing the memory in byte units (by setting mdru to 1), make sure the 8 high-order bits of the data bus (DB8–DB15) are fixed either high or low through a resistor. Only 8 low-order bits (DB0–DB7) are required to write to the register. Therefore, use the low order bits (DB0–DB7) to interface the device with, for example, an 8-bit CPU.

4.4 Mask Value Register

This register is used to mask selected data (pixel by pixel if desired). Any image data entered when the IDMK pin is set high is replaced by the corresponding data from this register.

	b7	b6	b5	b4	b3	b2	b1	b0	Address
CTRL									0008h

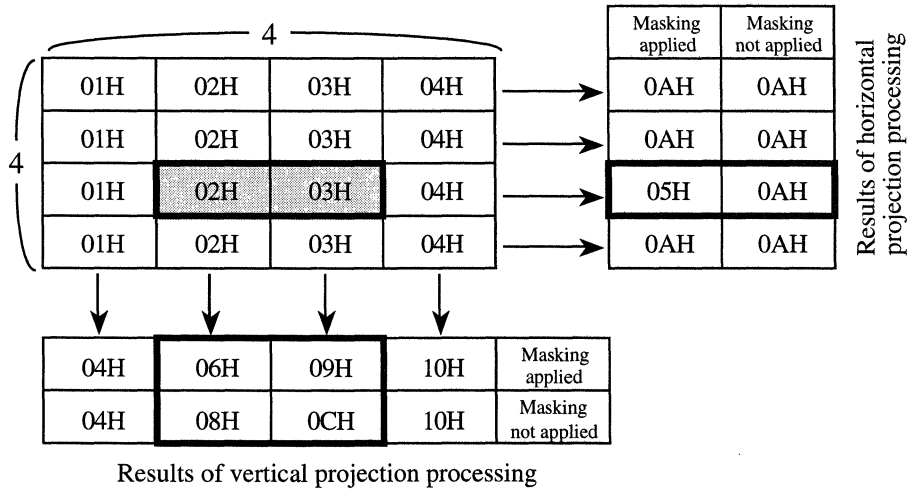
- This register resets to 00h.

b7–b0: Mask value (any data).

Typical example of masking: processing area 4 x 4, mask value = 00h

The diagram below shows the results of masking the areas enclosed by thick lines (masking applied). The numbers in the processing area represent the value of the image data (gray level) of each pixel.

Processing area

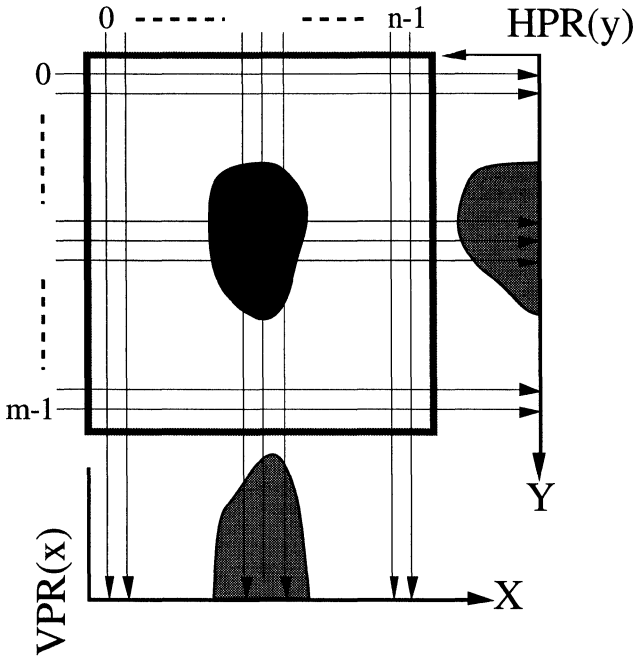


Results of vertical projection processing

Section 5: Circuit Operation and Control Method

5.1 Operation Outline

The IP90CO5A performs horizontal and vertical gray-level projection processing of raster-scanned 8-bit grayscale images, and simultaneously stores the results in its horizontal and vertical projection memories.



Gray level projection in the horizontal direction (Y-axis): $HPR(y) = \sum_x f(x,y)$

Gray level projection in the vertical direction (X-axis): $VPR(y) = \sum_x f(x,y)$

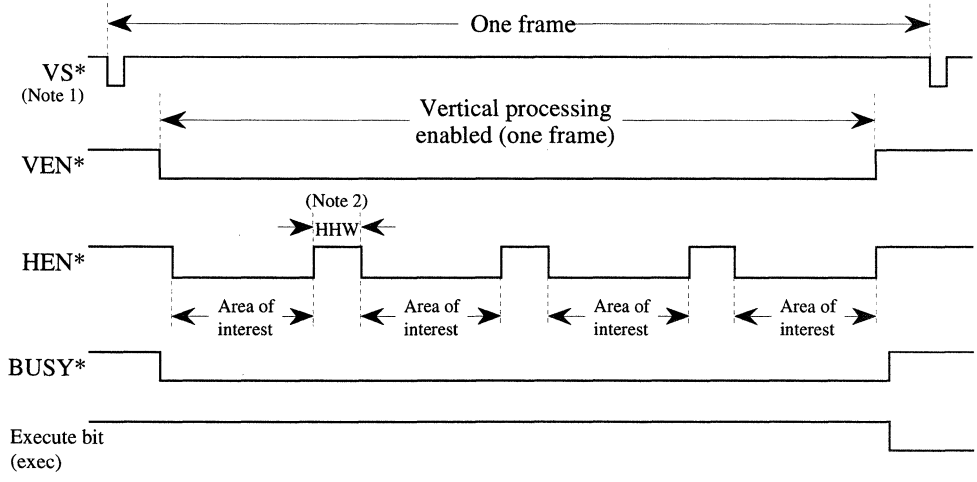
When the control register's execute bit (exec) is set to 1, the IP90C05A starts projection processing the grayscale image data of the effective area (rectangular) indicated by VEN* and HEN* after the first fall of VEN* (detected at the falling edge of image clock CLK).

5.2 Projection Processing in Non-Interlace Mode

1) Projection Processing

In non-interlace mode (scan = 0), each frame is configured with a single field, and the IP90C05A processes one field of image data at a time.

The diagram below shows the timing relationships between the VS*, VEN*, HEN*, and BUSY* signals and the execute bit (exec) in non-interlace mode. When VEN* goes low after the execute bit (exec) is set to 1, the IP90C05A drives BUSY* low and executes projection processing for the area of interest in one frame. After processing is complete, the execute bit automatically resets to 0 and drives BUSY* high. When BUSY* goes high, the results can be read from the internal memory.

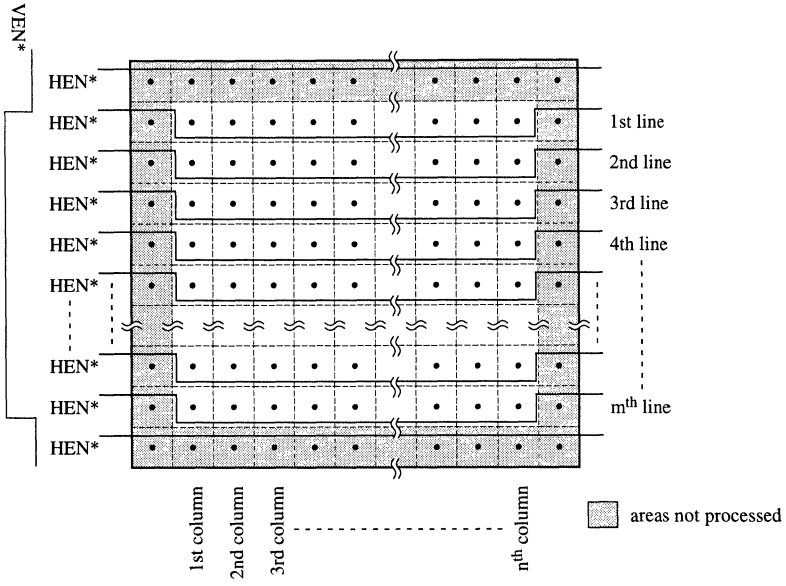


Note 1: VS* is shown here to indicate the start of a field. VS* does not need to be input, since the IP90C05A recognizes a low period with consecutive VS* pulses as a single field.

Note 2: Make sure that the period when HEN* is high (HHW) and VEN* is low lasts at least three cycles of the input clock.

2) Memory Mapping of Projection Processing Results

The diagram below shows the relationship between the projection processing results of a selected image area (m x n) and the memory addresses.



Horizontal projection memory

	MSB		LBS
Address mapping during external access (Note 2)	AD1, AD0 (Note 1)		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1000	Result of the 1st line		
1004	Result of the 2nd line		
1008	Result of the 3rd line		
100C	Result of the 4th line		
1000 + HEX [4(m-1)]	Result of the m^th line		
1000 + HEX (4m) ⋮ 17FC	Indeterminate data		

Vertical projection memory

	MSB		LBS
Address mapping during external access (Note 2)	AD1, AD0 (Note 1)		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1800	Result of the 1st column		
1804	Result of the 2nd column		
1808	Result of the 3rd column		
180C	Result of the 4th column		
1000 + HEX [4(n-1)]	Result of the n^th column		
1000 + HEX (4n) ⋮ 17FC	Indeterminate data		

Note 1: AD0 is ignored when memory is accessed using word units.

Note 2: Addresses are represented in hexadecimal notation.

5.3 Projection Processing in Interlace Mode

In interlace mode (scan = 1), the IP90C05A can process either two fields (one frame) of image data or one field. This is controlled by setting bit 1 (vmstr) in the mode register.

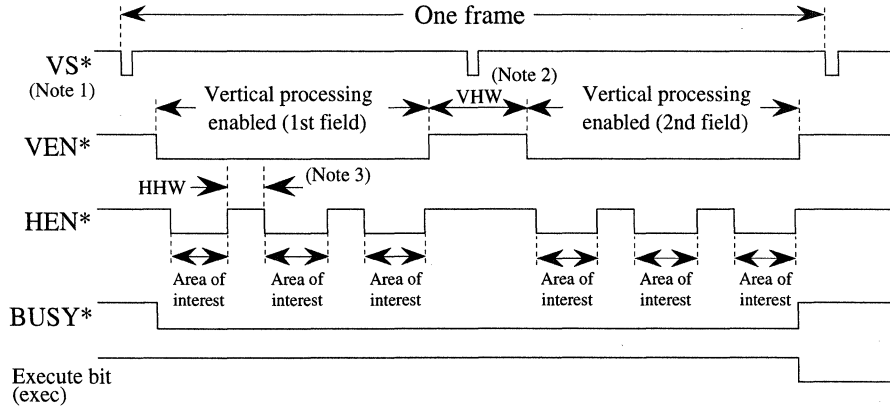
1) Projection Processing One Frame (Two Fields) of Image Data

To process two fields of image data, set the mode register's scan bit (bit 0) to 1, and the vmstr bit (bit 1) to 0.

The vertical projection processing results for two fields of image data consist of the vertical projection processing results of the 1st field and 2nd field. These results are stored in the vertical projection memory.

The horizontal projection processing results are stored in separate locations of the horizontal projection memory: the results of the 1st field are stored in locations word 0–255, while the results of the 2nd field are stored in locations word 256–511. These results can be read by programming bit 2 (hmru) and bit 3 (fl) of the control register to convert address mapping so that a frame of data can be read in the order of the coordinates or in the order of inputs, field by field. For details, see "2) Method for Reading Horizontal Projection Processing Results in Interlace Mode" in Section 5.5.

The diagram below shows the timing relationships between VS*, VEN*, HEN*, BUSY*, and the execute bit (exec) when two fields are processed in interlace mode. When VEN* goes low after the execute bit (exec) is set to 1, the I90C05A drives BUSY* low and executes projection processing for the effective area within one frame (two fields). After processing is complete, the IP90C05A automatically resets exec to 0 and drives BUSY* high. When BUSY* goes high, processing results can be read from the internal memory.



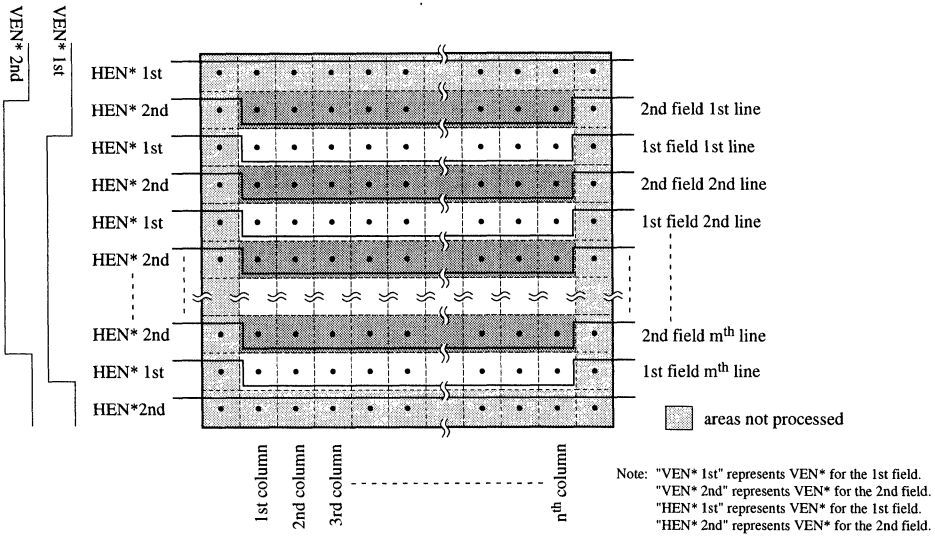
Note 1: VS* is shown to indicate the beginning of a field. VS* does not need to be input, since the IP90C05A recognizes a low period with consecutive VEN* pulses as a single field.

Note 2: Make sure that the period when VEN* is high (VHW), which is between the 1st and 2nd fields, lasts at least three cycles of the input clock.

Note 3: Make sure that the period when HEN* is high (HHW) and VEN* is low lasts at least three cycles of the input clock.

2) Memory Mapping the Projection Processing Results for One Frame (Two Fields)

The figure below shows an example of the relationship between the results of processing a selected image area (2m x n) and memory addresses. For details on how to read the horizontal projection processing results, see "2) Method for Reading Horizontal Projection Processing Results in Interlace Mode" in Section 5.5.



Horizontal projection memory

Address mapping during external access (Note 2)	AD1, AD0 (Note 1)		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1000	Result of the 1st field 1st line		
1004	Result of the 1st field 2nd line		
⋮	⋮		
1000 + HEX [4(m-1)]	Result of the 1st field m^th line		
⋮ 13FC	Indeterminate data		
1400	Result of the 2nd field 1st line		
1404	Result of the 2nd field 2nd line		
⋮	⋮		
1400 + HEX [4(m-1)]	Result of the 2nd field m^th line		
⋮ 17FC	Indeterminate data		

MSB LBS

Vertical projection memory

Address mapping during external access (Note 2)	AD1, AD0 (Note 1)		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1800	Result of the 1st column		
1804	Result of the 2nd column		
1808	Result of the 3rd column		
180C	Result of the 4th column		
	⋮		
1800 + HEX [4(n-1)]	Result of the n^th column		
1800 + HEX (4n) ⋮ 1FFC	Indeterminate data		

MSB LBS

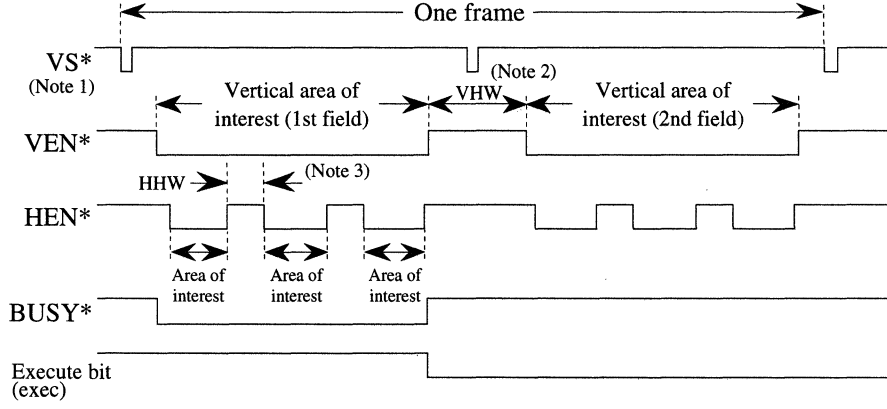
Note 1: AD0 is ignored when memory is accessed using word units.

Note 2: Addresses are represented in hexadecimal notation.

3) Single-Field Projection Processing

Execute projection processing for a single field by setting bit 1 (vmstr) of the mode register to 1 in interlace mode (scan = 1). More specifically, if the execute bit (exec) is not set to 1 when VEN* is high (VHW) before starting the 2nd field (after finishing the 1st field), the device processes only the 1st field.

The diagram below shows the timing relationships between VS*, VEN*, HEN*, BUSY*, and the execute bit (exec). When VEN* goes low after the execute bit (exec) is set to 1, the IP90C05A drives BUSY* low, and executes projection processing for only the area of interest within the field. After processing is complete, the IP90C05A automatically resets the execute bit to 0 and drives BUSY* high. When BUSY* goes high, processing results can be read from the internal memory.



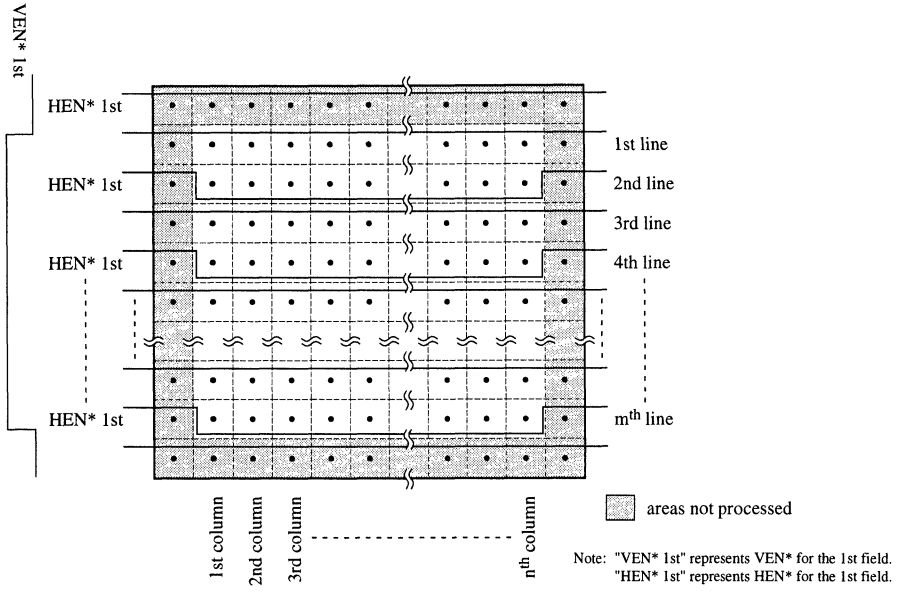
Note 1: VS* is shown here to indicate the beginning of a field. VS* does not need to be input, since the IP90C05A recognizes a low period with consecutive VEN* pulses as a single field.

Note 2: Make sure that the period when VEN* is high (VHW), which is between the 1st and 2nd fields, lasts at least three cycles of the input clock.

Note 3: Make sure that the period when HEN* is high (HHW) and VEN* is low lasts at least three cycles of the input clock.

4) Memory Mapping of Single-Field Projection Processing Results

The diagram below shows the relationship between the results of processing the selected image area (m lines x n pixels) and the memory addresses. For details on how to read the horizontal projection processing results, see "2) Method for Reading Horizontal Projection Processing Results in Interlace Mode" in Section 5.5.



Horizontal projection memory

Address mapping during external access (Note 2)	MSB AD1, AD0 (Note 1) LBS		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1000	Result of the 1st field 1st line		
1004	Result of the 2nd field 1st line		
1008	Result of the 3rd field 1st line		
100C	Result of the 4th field 1st line		
⋮	⋮		
1000 + HEX [4(m-1)]	Result of the 1st field m^{th} line		
1000 + HEX (4m)	Indeterminate data		
⋮			
17FC			

Vertical projection memory

Address mapping during external access (Note 2)	MSB AD1, AD0 (Note 1) LBS		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1800	Result of the 1st column		
1804	Result of the 2nd column		
1808	Result of the 3rd column		
180C	Result of the 4th column		
⋮	⋮		
1000 + HEX [4(n-1)]	Result of the n^{th} column		
1000 + HEX (4n)	Indeterminate data		
⋮			
1FFC			

Note 1: AD0 is ignored when memory is accessed using word units.

Note 2: Addresses are represented in hexadecimal notation.

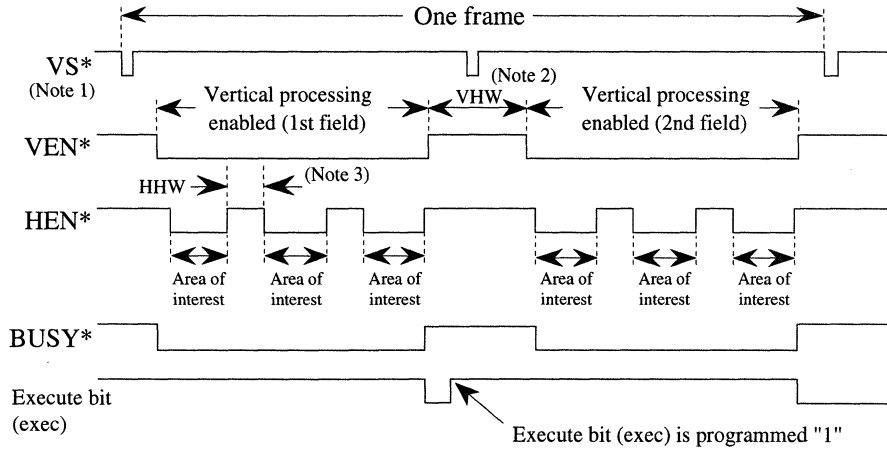
5) Single-Field Projection Processing (Split-Field Store Mode)

When processing only one field (of two) in interlace mode, the device can be placed in split-field store mode by setting bit 0 (scan) and bit 1 (vmstr) of the mode register to 1.

In split-field store mode, the results of each processed field are stored separately in the horizontal projection memory. The results of the first processed field (which is processed after the execute bit is set following a reset) are stored in horizontal projection memory, beginning with word 0. The results of processing the 2nd field (which is processed by setting the execute bit but without resetting after the 1st field has been processed) are stored in the horizontal projection memory, beginning with word 256.

In contrast, the results of vertical projection processing are stored directly without being split into separate locations. Therefore, these results are always stored in the vertical projection memory beginning with word 0, and this data is overwritten for each subsequent processed field.

The diagram below shows the timing relationships between VEN*, HEN*, BUSY*, and the execute bit (exec), when exec is set to 1 while VEN* is high (VHW) before starting the 2nd field (but after processing the 1st field).



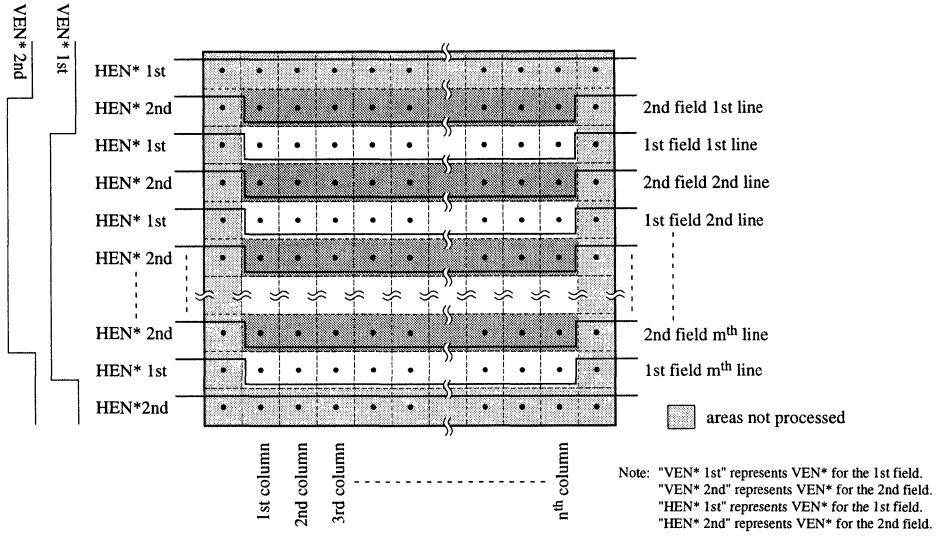
Note 1: VS* is shown here to indicate the beginning of a field. VS* does not need to be input, since the IP90C05A recognizes a low period with consecutive VEN* pulses as a single field.

Note 2: Make sure that the period when VEN* is high (VHW), which is between the 1st and 2nd fields, lasts at least three cycles of the input clock.

Note 3: Make sure that the period when HEN* is high (HHW) and VEN* is low lasts at least three cycles of the input clock.

6) Memory Mapping of the Projection Processing Results in Split-Field Store Mode

The diagram below shows an example of the relationship between the results of processing a selected image area (2m x n) and the memory addresses. The horizontal projection processing results for the 1st field are stored from word 0–255 in the horizontal projection memory, and the results for the 2nd field are stored from word 256–511. For details on reading horizontal projection processing results, see "2) Method for Reading Horizontal Projection Processing Results in Interlace Mode" in Section 5.5.



Horizontal projection memory

Address mapping during external access (Note 2)	MSB AD1, AD0 (Note 1) LBS		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1000	Result of the 1st field 1st line		
1004	Result of the 1st field 2nd line		
⋮	⋮		
1000 + HEX [4(m-1)]	Result of the 1st field m^th line		
⋮ 13FC	Indeterminate data		
1400	Result of the 2nd field 1st line		
1404	Result of the 2nd field 2nd line		
⋮	⋮		
1400 + HEX [4(m-1)]	Result of the 2nd field m^th line		
⋮ 17FC	Indeterminate data		

Vertical projection memory

Address mapping during external access (Note 2)	MSB AD1, AD0 (Note 1) LBS		
	10	01	00
	b17 b16	b15 - - - - - b8	b7 - - - - - b0
1800	Result of the 1st column		
1804	Result of the 2nd column		
1808	Result of the 3rd column		
180C	Result of the 4th column		
⋮	⋮		
1800 + HEX [4(n-1)]	Result of the n^th column		
1800 + HEX (4n) ⋮ 1FFC	Indeterminate data		

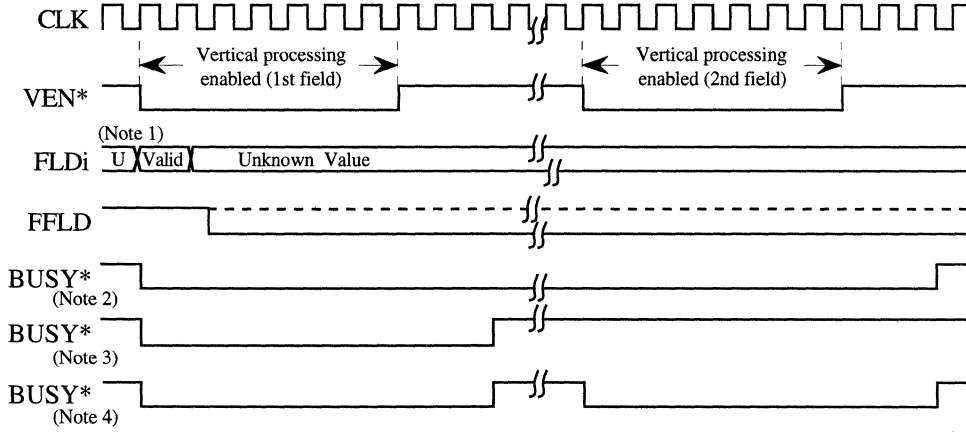
Note 1: AD0 is ignored when memory is accessed using word units.

Note 2: Addresses are represented in hexadecimal notation.

7) FFLD Pin Output

In interlace mode, the IP90C05A's FFLD pin outputs the 1st field's field discrimination signal regardless of whether the device is processing a frame (two fields) of image data or only a field. (In non-interlace mode, the FFLD bit is always high.) After setting the execute bit (exec) to 1, the IP90C05A monitors VEN* in the 1st field: when VEN* goes low, the device latches FLDi (field identification) signal to the first fall of CLK. It then outputs the latched field discrimination signal for the 1st field from the FFLD pin one clock cycle later. Even in the 2nd field, the FFLD pin holds and outputs the field identification signal for the 1st field. The FFLD pin retains and outputs the field identification signal for the 1st field even if BUSY* is driven high. When reset, it goes high.

The diagram below shows the timing relationships of CLK, VEN*, FLDi, FFLD, and BUSY*.



Note 1: U = unknown value

Note 2: This shows one frame (two fields) of image data being processed.

Note 3: This shows only one field being processed.

Note 4: This shows fields of data being processed in split-field store mode.

5.4 Overflow

1) Overflow During Projection Processing in Both Directions (hproj = 0, vproj = 0)

If the horizontal or vertical processing range exceeds 512 pixels, the IP90C05A drives the OVF signal high to indicate overflow status. It also simultaneously resets the execute bit to 0 to stop processing and drive BUSY* high. The results of processing performed before the overflow are still valid, and can be read from the internal memory. Overflow status can be cleared by resetting the device, which sets OVF back to low. The timing charts below show typical overflow conditions in the horizontal and vertical directions.

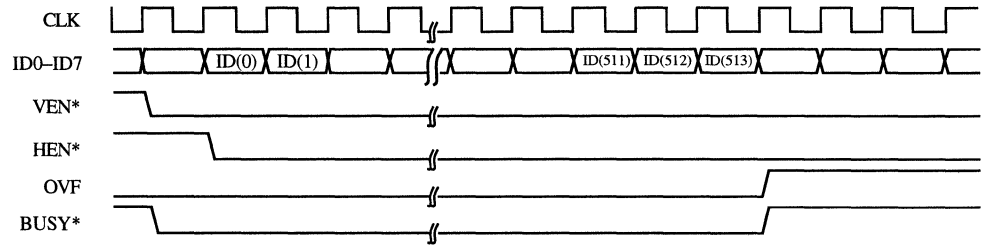
2) Overflow In Vertical Processing Area Expanded Mode (hproj = 1, vproj = 0)

Setting bit 2 (hproj) of the mode register to 1 and bit 3 (vproj) to 0 puts the device into vertical processing area expanded mode (with a maximum range of 1024 pixels). In this mode, OVF does not output overflow status even if the vertical processing range exceeds 512 pixels; instead, it stays low, as it was when processing began. It also does not output overflow status when the horizontal processing range exceeds 512 pixels.

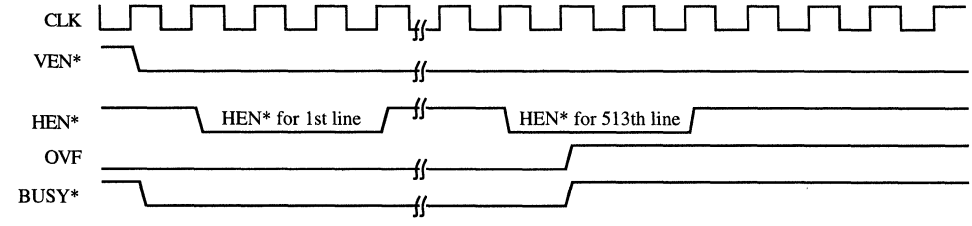
3) Overflow In Horizontal Processing Area Expanded Mode (hproj = 0, vproj =1)

Setting bit 2 (hproj) of the mode resister to 0 and bit 3 (vproj) to 1 puts the device into horizontal processing area expanded mode (with a maximum range of 1024 pixels). In this mode, OVF does not output overflow status even if the horizontal processing range exceeds 512 pixels; instead, it stays low, as it was when processing began. It also does not output overflow status even when the horizontal processing range exceeds 1024 pixels. Note, however, that it does output overflow status when the vertical processing range exceeds 512 pixels.

Overflow in the horizontal direction



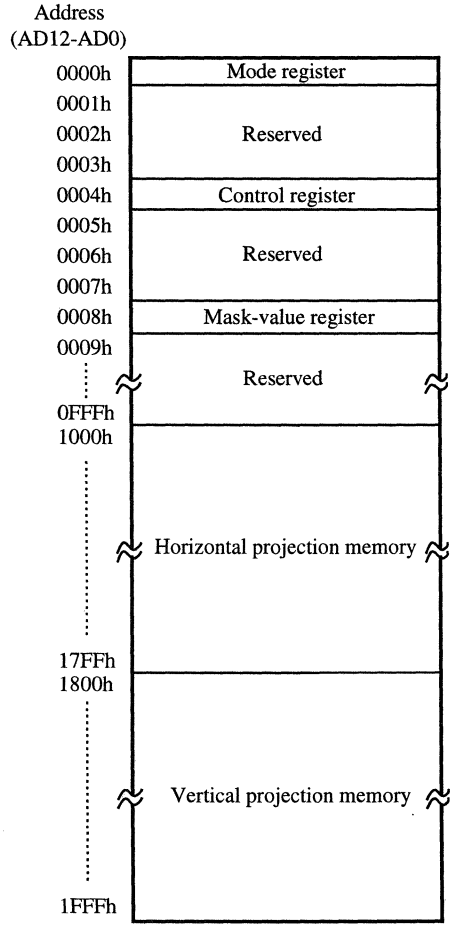
Overflow in the vertical direction



5.5 Reading Processing Results

1) Setting the Address Bus

The diagram below shows the IP90C05A's memory mapping.



Note: Register addresses are decoded only for AD2 and AD3. (Register addresses are not fully decoded.) Do not use addresses 0001h–0003h, 0005h–0007h, or 0009h–0FFFh.

When reading projection processing results from memory, each bit in control address bus AD12–AD0 has the following settings:

AD12: Chooses between memory access and register access:

AD12 = 1: Memory access

AD12 = 0: Register access

AD11: Specifies whether horizontal or vertical projection results are read:

AD11 = 1: Reads vertical projection results

AD11 = 0: Reads horizontal projection results

AD10–AD2: Internal memory word addresses

The control address bus, when bits AD0 and AD1 are ignored, consists of the 18 bits in the combined horizontal and vertical projection memories. (Data for addresses outside the processing area are indeterminate and not guaranteed.)

Control address bus										
AD10	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	Word addresses in horizontal projection memory (hex)	Word addresses in the vertical projection memory (hex)
0	0	0	0	0	0	0	0	0	000h	000h
0	0	0	0	0	0	0	0	1	001h	001h
0	0	0	0	0	0	0	1	0	002h	002h
0	0	0	0	0	0	0	1	1	003h	003h
0	0	0	0	0	0	1	0	0	004h	004h
0	0	0	0	0	0	1	0	1	005h	005h
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0	0	0	0	0	1	1	1	1	00Fh	00Fh
0	0	0	0	1	0	0	0	0	010h	010h
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
1	1	1	1	1	1	1	1	0	1FFh	1FFh
1	1	1	1	1	1	1	1	1	1FFh	1FFh

AD1-AD0: Identifies 18-bit memory data for word or byte access.

For word access

Control address bus		Data bus														
AD1	AD0	DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8	DB7	DB6	DB5	DB4	DB3	DB1	DB0
0	x	Sixteen low-order bit are output														
1	x	0	0	0	0	0	0	0	0	0	0	0	0	0	Two high-order bits are output	

Note: x means "Don't care;" it may be logic or 0. AD0 is ignored.

For byte access

Control address bus		Data bus														
AD1	AD0	DB15	DB14	DB13	DB12	DB11	DB10	DB9	DB8	DB7	DB6	DB5	DB4	DB3	DB1	DB0
0	0	x	x	x	x	x	x	x	x	Eight low-order bits are output						
0	1	x	x	x	x	x	x	x	x	Eight middle-order bits are output						
1	0	x	x	x	x	x	x	x	x						Two high-order bits are output	
1	1	Inhibited														

Note: x = Unknown. Values cannot be guaranteed. Logic 0 or 1 is output.

The access mode (word access or byte access) is determined by bit 4 (mdru) of the control register.

2) Method for Reading Horizontal Projection Processing Results in Interlace Mode

The following describes how to read the results of processing two fields of image data in interlace mode as described in Section 5.3, "Projection Processing in Interlace Mode."

Horizontal projection processing results are stored in the horizontal projection memory, as shown in Section 5.3. To read these results, set bit 0 (scan) of the mode register to 1, then set bit 2 (hmru) of the control register to 0 or 1 to map the addresses in horizontal projection memory in either (respectively) the order of the frame coordinates or of the inputs for each field, as shown below.

a) Addresses in order of the frame coordinates

In this mode, the horizontal projection processing results for the 1st and 2nd fields are alternately mapped in the read address space of the horizontal projection memory. To enter this mode, set bit 0 (scan) of the mode register to 1, and bit 2 (hmru) of the control register to 0. Then, to arrange the addresses in order of the frame coordinates, set bit 3 (fl) of the control register to specify whether the first line to be projection processed in the horizontal direction is on an even field or an odd field. The IP90C05A then maps the results on the memory addresses as shown below.

Horizontal projection memory

	MSB		LBS	
Address mapping during external access (Note 2)	AD1, AD0 (Note 1)			
	10	01	00	
	b17- -b16	b15 - - - -b8	b7 - - - -b0	
1000				→ 1st line of the field specified by the fl-bit
1004				→ 1st line of the field not specified by the fl-bit
1008				→ 2nd line of the field specified by the fl-bit
100C				→ 2nd line of the field specified by the fl-bit
⋮	⋮	⋮	⋮	⋮
17F8				→ 256th line of the field specified by the fl-bit
17FC				→ 256th line of the field not specified by the fl-bit

Note 1: AD0 is ignored when memory is accessed using word units,

Note 2: Addresses are represented in hexadecimal notation.

b) Addresses in the order of the inputs for each field (scan = 1, hmru = 1)

In this mode, horizontal projection processing results are mapped in read address space of the horizontal projection memory in the order in which processing is performed, and the results are stored. To enter this mode, set bit 0 (scan) of the mode register to 1 and bit 2 (hmru) of the control register to 1. Bit 3 (fl) of the control register does not affect this mode.

First half of the memory: results of the 1st field
Second half of the memory: results of the 2nd field

Horizontal projection memory

Address mapping during external access (Note 2)	AD1, AD0 (Note 1)		
	10	01	00
	MSB b17 b16	b15 - - - - b8	b7 - - - - b0 LBS
1000			
1004			
⋮		⋮	
13FC			
1400			
1404			
⋮		⋮	
17FC			

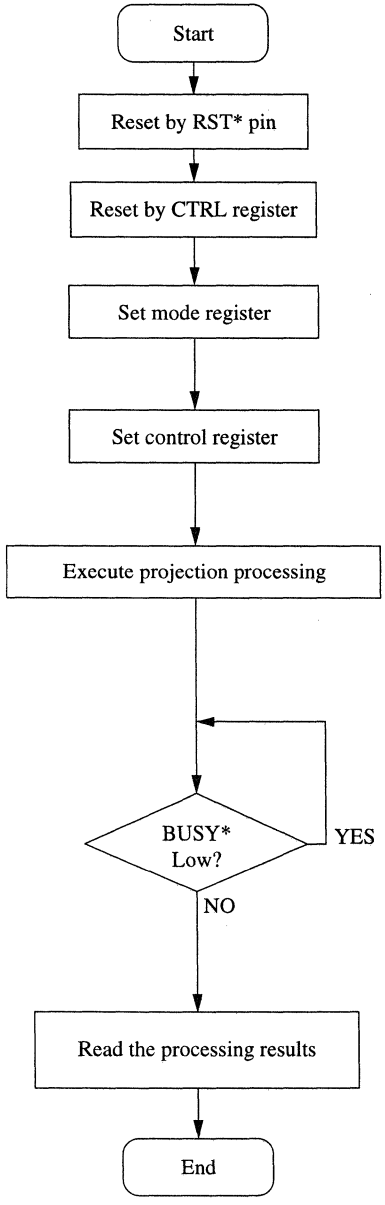
Results of the 1st field

Results of the 2nd field

Note 1: AD0 is ignored when memory is accessed using word units,
Note 2: Addresses are represented in hexadecimal notation.

5.6 Control Method

The following flowchart shows the IP90C05A's control method.



Select the desired mode.

Select the execute bit (exec). (The IP90C05A automatically clears exec to 0 after processing is finished.

After exec is set, do the following:

1. When VEN* goes low, it drives BUSY* low.
2. The image data is processed in the area of interest marked by VEN* and HEN*, and the results are stored in memory.
3. Processing ends three clock cycles after VEN* goes high in interlace or 1-field interlaced mode, and BUSY* is driven high. In 1-frame (2-field) interlaced mode, the device ends processing three clock cycles after VEN* indicates that the 2nd field becomes high, and drives BUSY* high.

After confirming that BUSY* is high, read the results from memory.

Section 6. Electrical Characteristics

6.1 Absolute Maximum Ratings

Item	Symbol	Min.	Typ.	Max.	Unit
Supply voltage	Vdd	-0.5		6	V
Input/output voltage	Vin/Vout	-0.5		Vdd+0.5	V
Operating temperature	Topt	0		70	°C
Storage temperature	Tstg	-10		80	°C

6.2 Recommended Operating Conditions

Item	Symbol	Min.	Typ.	Max.	Unit
Supply voltage	Vdd	4.75	5.0	5.25	V
Input voltage	Vin	0		Vdd	V
Ambient temperature	Ta	0		70	°C
Input rise time	Tri	0		50	ns
Input fall time	Tfi	0		50	ns

6.3 Pin Capacitance (Vdd = Vin = 0 V)

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input capacitance	Cin	f = 1 MHz		10	25	pF
Output capacitance	Cout	f = 1 MHz		10	25	pF

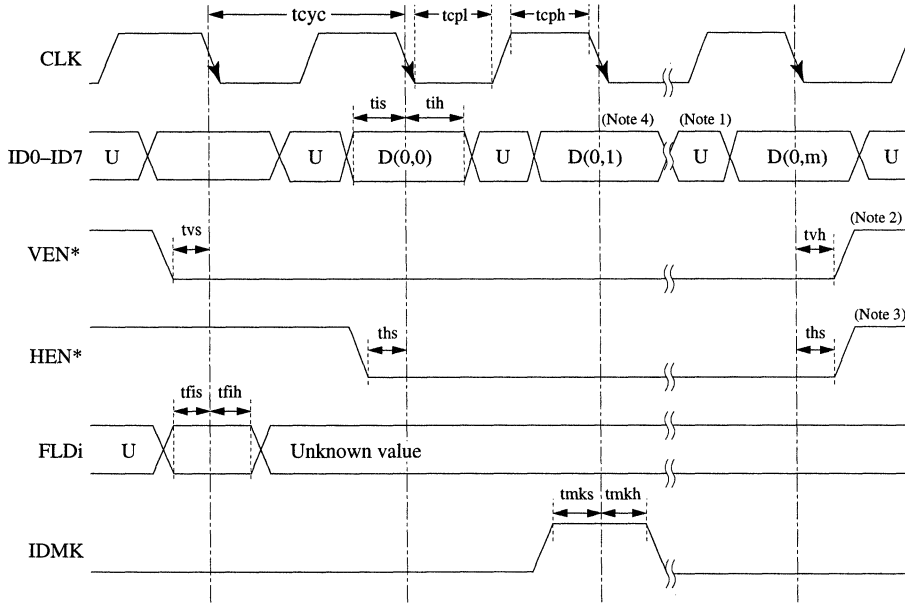
6.4 DC Characteristics

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input high voltage	Vih		2.4		Vdd	V
Input low voltage	Vil		0		0.8	V
Output high voltage	Voh	Vdd = 4.75, Ioh = -0.5 mA	2.4			V
Output low voltage	Vol	Vdd = 4.75 V, Iol = 4.0 mA			0.4	V
Output shorting current (Note 1)	Ios	Vout = 0V			250	mA
Input leakage voltage	Iix	GND ≤ Vin ≤ Vdd	-20		20	µA
Output off leakage voltage	Ioz	GND ≤ Vout ≤ Vdd	-25		25	µA

Note 1: The output shorting current shown above is the absolute maximum voltage that can be applied to one pin for one second without causing a short.

6.5 AC Characteristics (load capacitance 30 pF)

1) Image Input Signal



(units: ns)

		IP90C05A			IP90C05A-HS		
Parameter	Symbol	Min.	Typ.	Max.	Min.	Typ.	Max.
CLK cycle	tcyc	50.0			33.3		
CLK high period	tcph	20.0			11.0		
CLK low period	tcpl	20.0			16.0		
ID0-ID7 setup time	tis	2.0			1.0		
ID0-ID7 hold time	tih	7.5			7.0		
VEN* setup time	tvs	2.0			1.0		
VEN* hold time	tvh	7.5			7.0		
HEN* setup time	ths	2.0			1.0		
HEN* hold time	thh	7.5			7.0		
FLDi setup time	tfis	2.0			1.0		
FLDi hold time	tfih	7.5			7.0		
IDMK setup time	tmks	2.0			1.0		
IDMK hold time	tmkh	7.5			7.0		

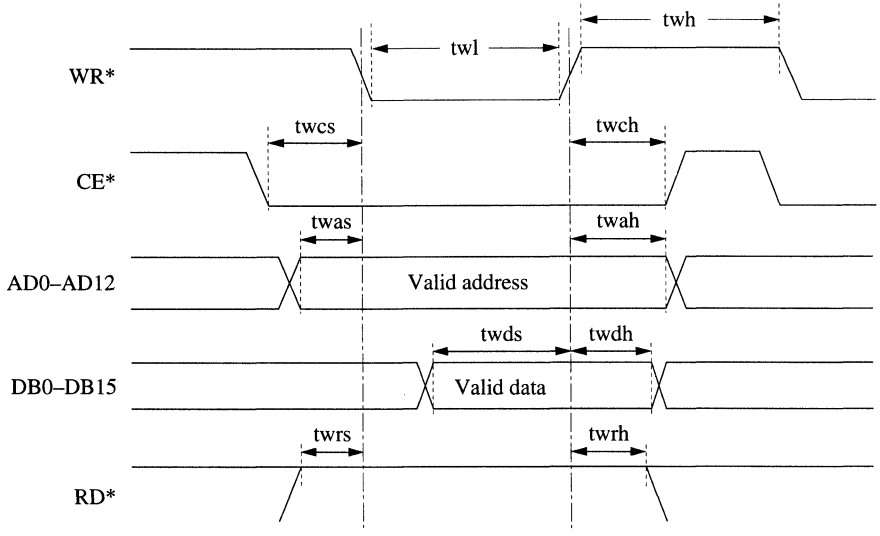
Note 1: U = unknown value

Note 2: The VEN* recovery period (time between the 1st and 2nd fields when VEN* is high) must be at least three CLK cycles.

Note 3: The HEN* recovery period (time when HEN* is high and VEN* is low) must be at least three CLK cycles.

Note 4: Data D (0,1) are overwritten by the value of the mask-value register because IDMK is high during this period.

2) CPU Interface Timing (Write to the Register)



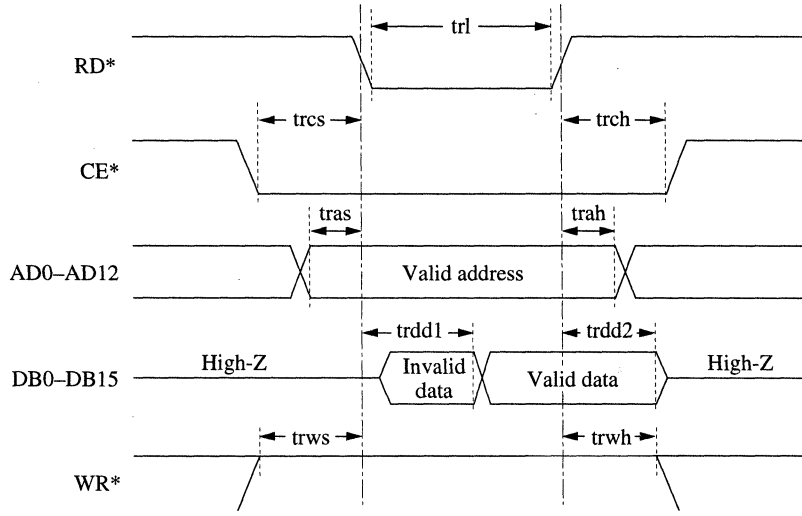
(units: ns)

Parameter	Symbol	Min.	Typ.	Max.
WR* low period	twl	1.5tcyc	—	—
WR* high period	twh	0.5tcyc	—	—
CE* setup time relative to the fall of WR*	twcs	3	—	—
CE* hold time relative to the rise of WR*	twch	3	—	—
AD setup time relative to the fall of WR*	twas	3	—	—
AD hold time relative to the rise of WR*	twah	3	—	—
DB setup time relative to the rise of WR*	twds	1.5tcyc	—	—
DB hold time relative to the rise of WR*	twdh	3	—	—
RD* setup time relative to the fall of WR*	twrs	3	—	—
RD* hold time relative to the rise of WR*	twrh	3	—	—

Note 1: Although the CPU need not write to each register in synchronization with the image data clock (CLK), make sure that CLK is input even during write cycles.

Note 2: Values internally set in the register require at least three CLK cycles after WR* goes high to become valid.

3) CPU Interface Timing (Memory Read)



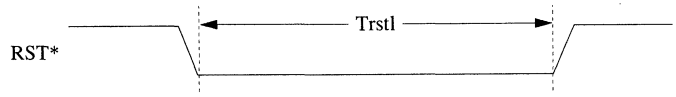
(units: ns)

Parameter	Symbol	Min.	Typ.	Max.
RD* low period	trl	1.5tcyc	—	—
CE* setup time relative to the fall of RD*	trcs	3	—	—
CE* hold time relative to the rise of RD*	trch	3	—	—
AD setup time relative to the fall of RD*	tras	3	—	—
AD hold time relative to the rise of RD*	trah	3	—	—
DB setup time relative to the fall of RD*	trdd1	—	—	1.0tcyc+30
DB hold time relative to the rise of RD*	trdd2	5	—	0.5tcyc+30
WR* setup time relative to the rise of RD*	trws	3	—	—
WR* hold time relative to the fall of RD*	trwh	3	—	—

Note 1: Although the CPU need not write to each register in synchronization with the image data clock (CLK), make sure that CLK is input even during write cycles.

Note 2: When RD* or CE* is high, the IP90C05A's data bus lines DB0–DB15 are placed in a high-impedance state.

4) CPU interface timing (reset)



(units: ns)

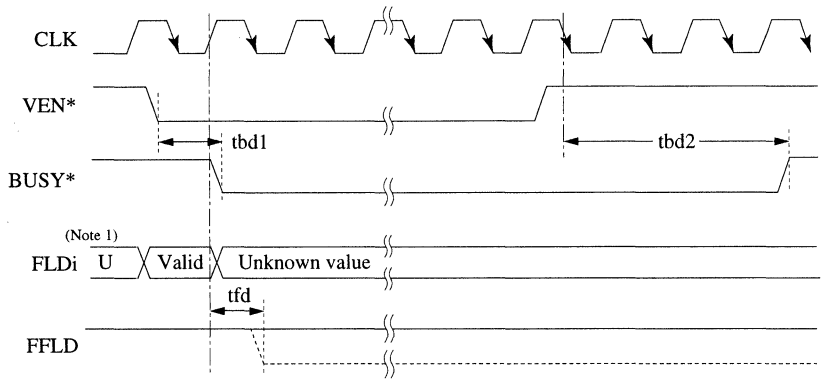
Parameter	Symbol	Min.	Typ.	Max.
RST* low period	trstl	3tcyc	—	—

Note 1: Although RST* need not be synchronized with the image data clock (CLK), make sure that CLK is input even during reset cycles.

Note 2: The device's internal logic reset requires at least four CLK cycles after the falling edge of RST* is detected.

5) Output timing

a) BUSY and FFLD) outputs



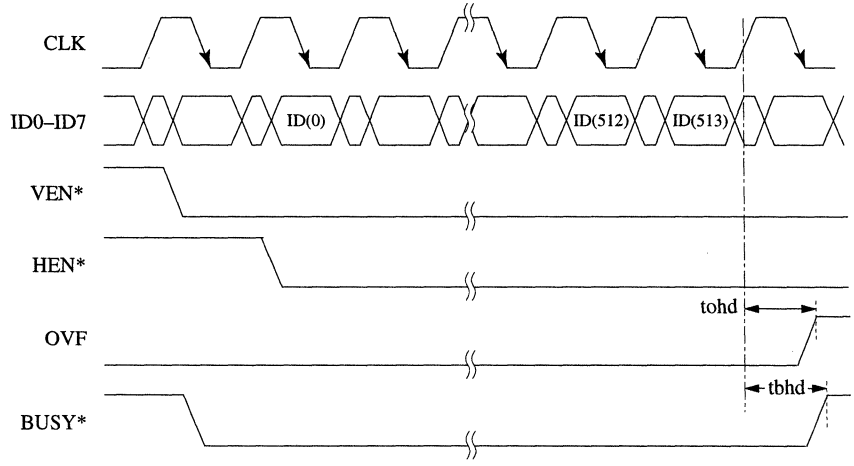
Note 1: U = Unknown

(units: ns)

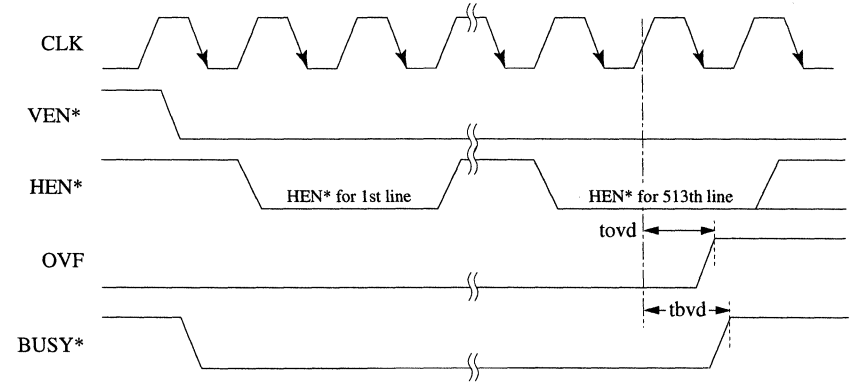
Parameter	Symbol	Min.	Typ.	Max.
Delay time from VEN* fall till BUSY* is driven low	tdb1	2	—	0.5tcyc+25
Delay time from VEN* rise till BUSY* goes high	tdb2	2.5tcyc	—	2.5tcyc+25
FFLD delay time from CLK rise	tfd	2	—	25

b) OVF output

Horizontal overflow



Vertical overflow



(units: ns)

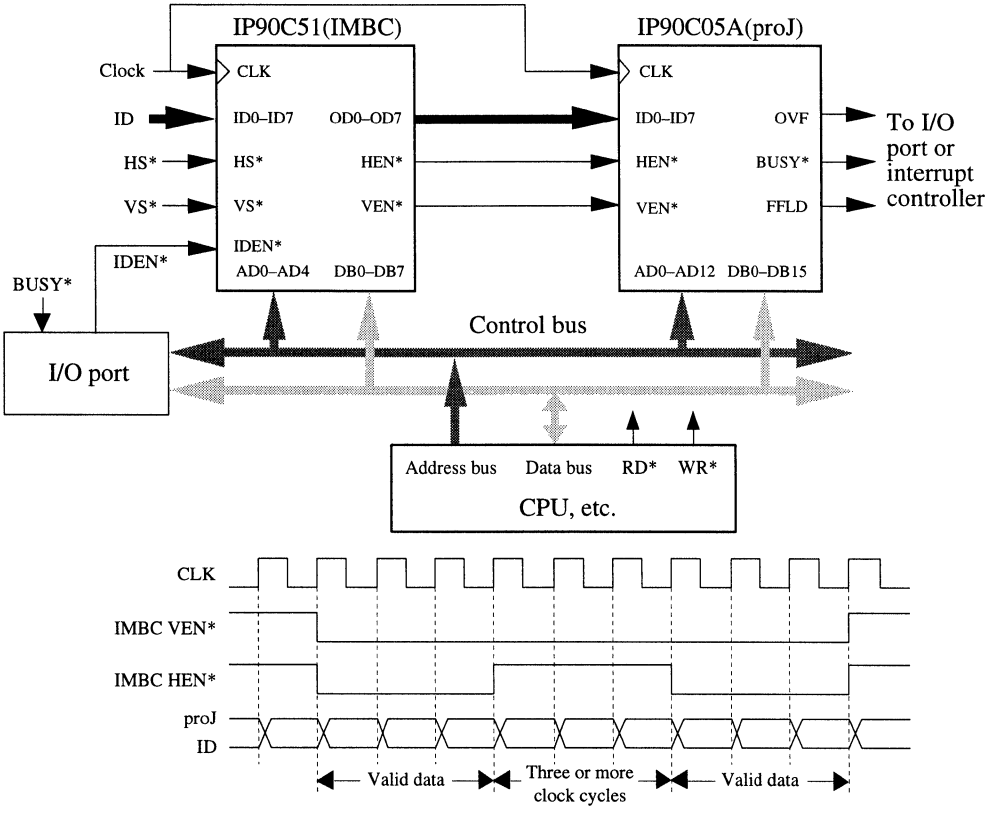
Parameter	Symbol	Min.	Typ.	Max.
OVF delay time from CLK rise during horizontal overflow	tohd	2	—	25
BUSY* delay time from CLK rise during horizontal overflow	tbhd	2	—	25
OVF delay time from CLK rise during vertical overflow	tovd	2	—	25
BUSY* delay time from CLK rise during vertical overflow	tbvd	2	—	25

Section 7: Sample Applications

7.1 Connecting the IP90C05A and IP90C51

Sumitomo Metal Industries' IP90C51 (IMBC) LSI processes images by clipping each area of the raster-scanned digital image data as an "area of interest" (AOI). The diagram below shows a sample application using the IP90C05A together with the IP90C51 to process images input from a camera. Note that this example does not guarantee clock frequency timings for any specific application, and that these timings must be checked carefully when designing an actual system.

Example: Using the IP90C05A with the IP90C51 and CPU.



- The IP90C05A's VEN* and HEN* inputs are level-sensitive, but not edge triggered. Therefore, the IP90C51's VEN* and HEN* generate level output.

  The recovery period for HEN* is three clock cycles or more.

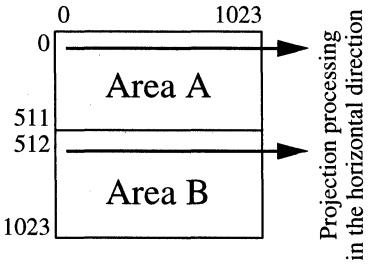
The following shows the setting and operation procedures:

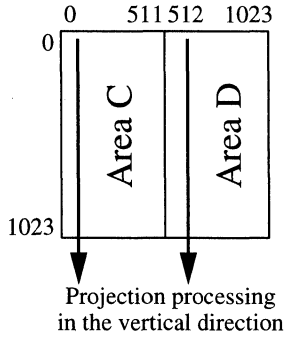
IMBC	Drives IDEN* high (IMBC stops).
↓	
IMBC	Sets the operation mode (VS* sync mode OFF).
↓	
proJ	Sets the operation mode.
↓	
proJ	Sets the execute bit.
↓	
IMBC	Sets the execute bit.
↓	
IMBC	Drives IDEN* low (IMBC starts execution).
	The IP90C05A proJ chip starts processing the next time VEN* or HEN* is asserted.

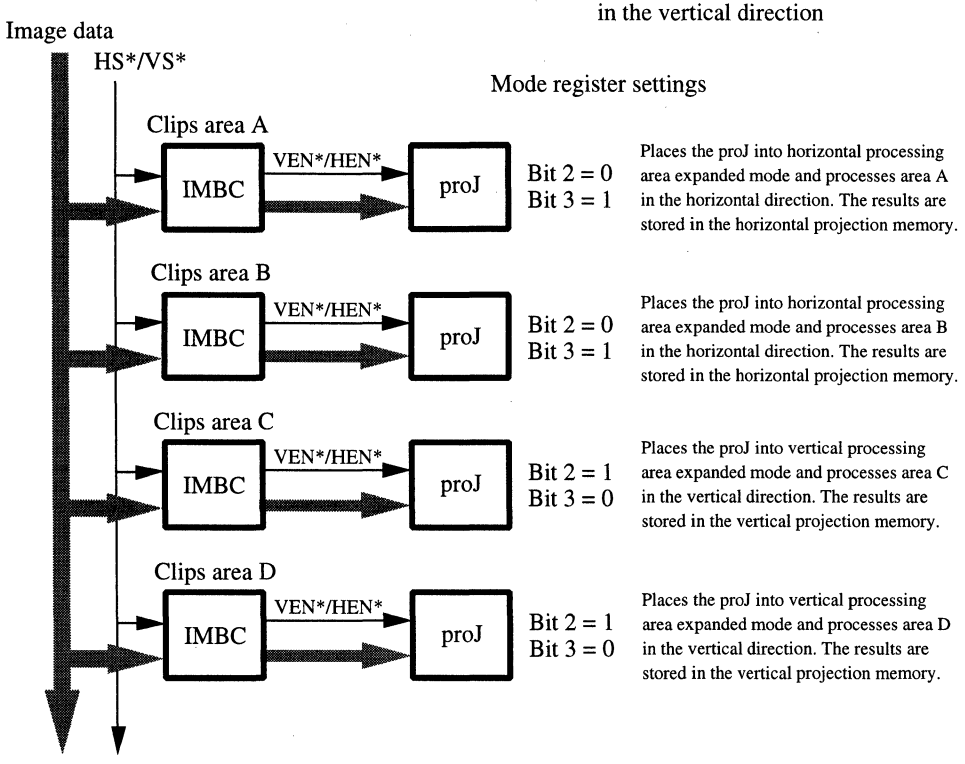
7.2 Expanding the Processing Range (1024 x 1024 pixels)

Extended operation requires four IP90C05A (proJ) chips. Place each IP90C05A into horizontal or vertical processing area expanded mode by setting bit 2 (hproj) and bit 3 (vproj) of the mode register.

The IP90C51 allocates each proJ's processing areas (areas A, B, C, and D) in correspondence with expansion mode.

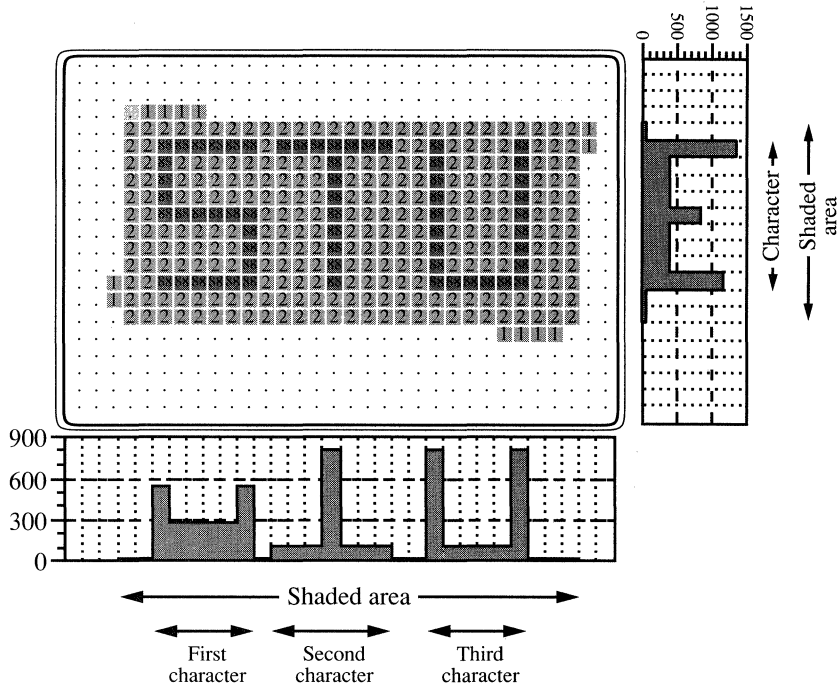






7.3 Character Area Recognition

Projection processing results make it possible to recognize simple character areas and symbols.



7.4 High-Speed Character Recognition by Encoding and a Recognition Table Search

(Source: Character Recognition Technique by Image Processing by Yamauchi, Electronics [Ohm Company], February 1992, pages 54-55.)

In most industrial applications, such as inspecting IC marking (i.e., package character recognition), the type of character, font, and size remain almost constant. In cases where the characters are limited as in such industrial applications, the device provides an easy way to recognize characters quickly by "integration encoding the projection processing results."

The following describes the execution procedures for this application:

(1) Executing projection processing

Projection process the characters to be recognized, as shown on the following page, in the horizontal and vertical directions using and appropriate resolution (32 lines in Figure 7-1). The projection processing described here requires summing binary values of the pixels on each horizontal and vertical line. (The horizontal and vertical lines in Figure 7-1 are summed to obtain a cumulative total.)

(2) Integration and encoding

After projection processing is completed, integrate the results at appropriate line intervals (every 4 lines in the example shown here). Quantize this integrated value at a threshold of up to 32, 64, 96, or 128 into one of four values: 00, 01, 10, or 11. Convert these quarternary-quantized eight values in each direction into a 2-byte code (2 bytes each for the horizontal and vertical directions, for a total 4 bytes). Thus, the character to be recognized is translated into a 4 -byte code.

(3) Creating a table for character recognition by the table

After encoding is completed, create a recognition table for the horizontal and vertical directions as shown below. After this is done, characters can be recognized by comparing the code generated by the same method from the characters against the values in this table.

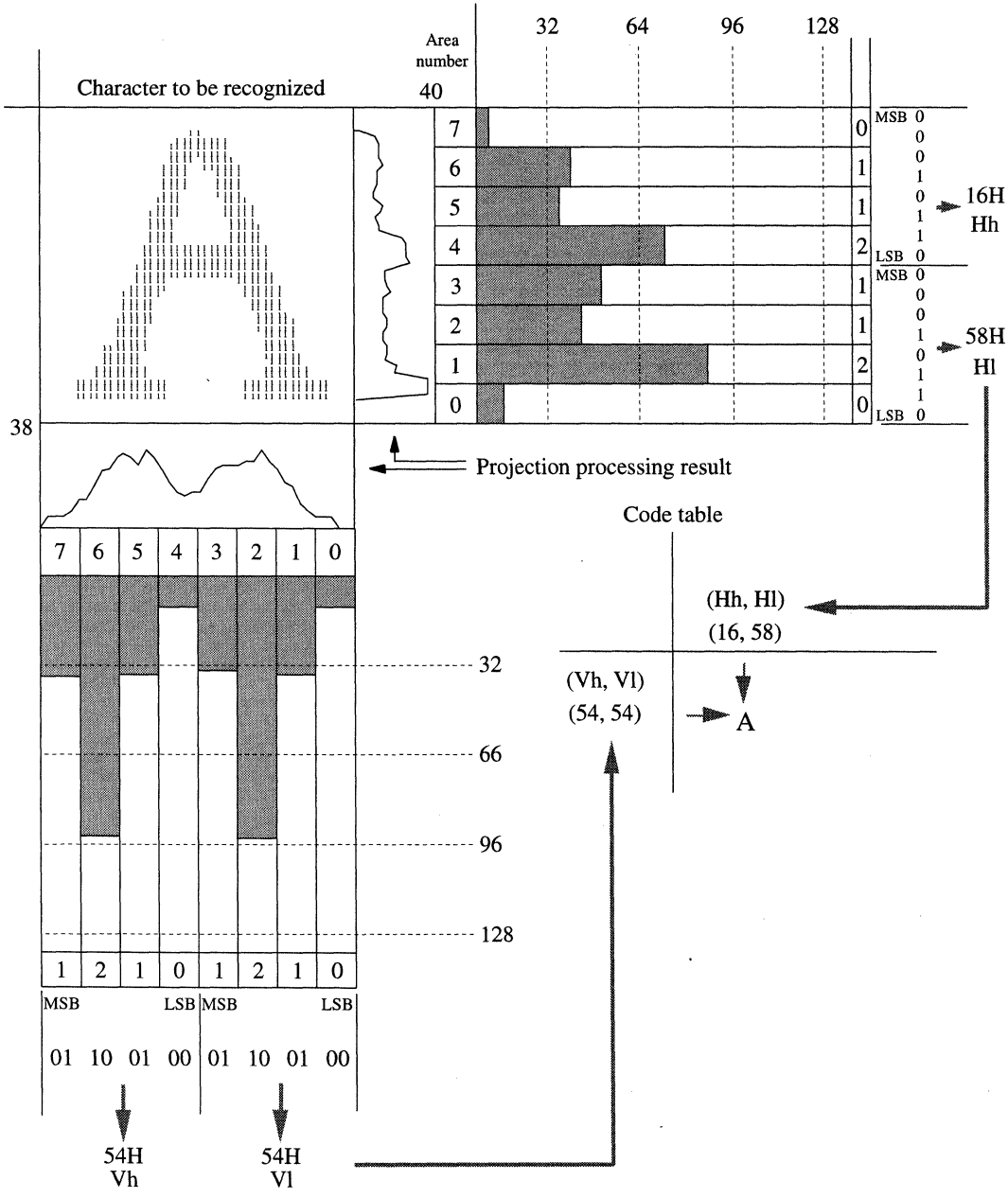


Figure 7-1: Integrating and encoding the projection processing result

After the processing described above is executed, the results shown in Table 7-1 can be obtained. Table 7-2 shows experimental results obtained when the technique described here was applied to Japanese characters in the "mincho" font. These results reveal that the technique is highly accurate, and can even discriminate between very similar characters such as "1" and "I".

Although this technique can only be applied to certain types of characters, it provides the following advantages over conventional methods because it uses the integration encoded results of projection processing:

- This technique is not affected by noise.
- This technique is not affected by the inclination of the characters to be recognized.
- The algorithm is very simple and executes at high speed.

Character to be recognized: B								Character to be recognized: 1								Character to be recognized: I							
Horizontal direction (15h, 55h)				Vertical direction (18h, 60h)				Horizontal direction (00h, 00h)				Vertical direction (02h, 00h)				Horizontal direction (00h, 00h)				Vertical direction (02h, 40h)			
PROJ	SUM	CODE	Hh Hl	PROJ	SUM	CODE	Hh Hl	PROJ	SUM	CODE	Hh Hl	PROJ	SUM	CODE	Hh Hl	PROJ	SUM	CODE	Hh Hl	PROJ	SUM	CODE	Hh Hl
0				0				0				0				0				0			
0				0				0				0				0				0			
5				1				0				0				0				0			
16	21	0		3	4	0		2	2	0		0	0	0		0	0	0		0	0	0	
13				3				4				0				9				0			
9				4				6				0				8				0			
8				21				6				0				5				0			
8	38	1		32	60	1		4	20	0		0	0	0		4	26	0		0	0	0	
8				32				3				0				4				0			
8				32				3				1				4				1			
8				21				3				2				4				2			
8	32	1		7	92	2		3	12	0		3	6	0		4	16	0		3	6	0	
9				7				3				4				4				4			
9				7				3				26				4				21			
13				7				3				31				4				31			
14	45	1	15h	7	28	0	18h	3	12	0	00h	32	93	2	02h	4	16	0	00h	31	87	2	02h
10				8				3				19				4				31			
9				10				3				2				4				15			
9				13				3				1				4				3			
9	37	1		19	50	1		3	12	0		1	23	0		4	16	0		3	52	1	
9				26				3				0				4				2			
9				26				3				0				4				1			
9				22				3				0				4				0			
9	36	1		15	89	2		3	12	0		0	0	0		4	16	0		0	3	0	
9				6				3				0				4				0			
9				1				3				0				4				0			
12				0				4				0				4				0			
17	47	1		0	7	0		9	19	0		0	0	0		6	18	0		0	0	0	
6				0				4				0				11				0			
0				0				0				0				0				0			
0				0				0				0				0				0			
0	6	1	55h	0	0	0	60h	0	4	0	00h	0	0	0	00h	0	11	0	00h	0	0	0	40h

Table 7-1: Experimental Results

Characters to be recognized	Horizontal code	Vertical code
B	15h, 55h	18h, 60h
1	00h, 00h	02h, 00h
I	00h, 00h	02h, 40h

Table 7-2: Experimental Results When the Technique is Applied to Mincho Characters

SMI ASSP

Image Processing LSI Series

IP90C08

Template Matching Processor (templa)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.3

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1. Characteristics and Features

1.1 Overview of Specifications

[Basic Functions]

- Single-chip architecture for template matching of 16 x 7-pixel image areas.
- Template matching of grayscale images in 0 to 255 gradients.
- Real-time processing at image data input rates up to 30 MHz.
- Results of template matching are determined by summing the differences in grayscale values between the template image and the pixels in an area of the input image corresponding to predetermined coordinates on the template.
- Within each image field, four sets of coordinates of areas that have low sum-of-differences values (that is, areas very similar to the template) are ranked in order of similarity to the template and stored with their sum-of-differences values.
- Sum-of-differences values are defined as the matched filter output from the image areas to which they relate, and are output from the IP90C08 in real time.

[Area Expansion Functions]

- The template and image areas can be expanded by using multiple IP90C08 templa chips simultaneously and externally summing the sum-of-differences (matched filter) output from all chips.
- The template matching area can be expanded by using multiple IP90C08s in a cascade connection. This essentially creates an expanded matched filter (sum of differences calculator), with the respective sum-of-differences outputs connected to the expanded input from the main IP90C08 to function as an expanded template matching area. For example, three IP90C08s in a cascade connection can output 17-bit sum-of-differences values for an area as large as 16 x 21 pixels.

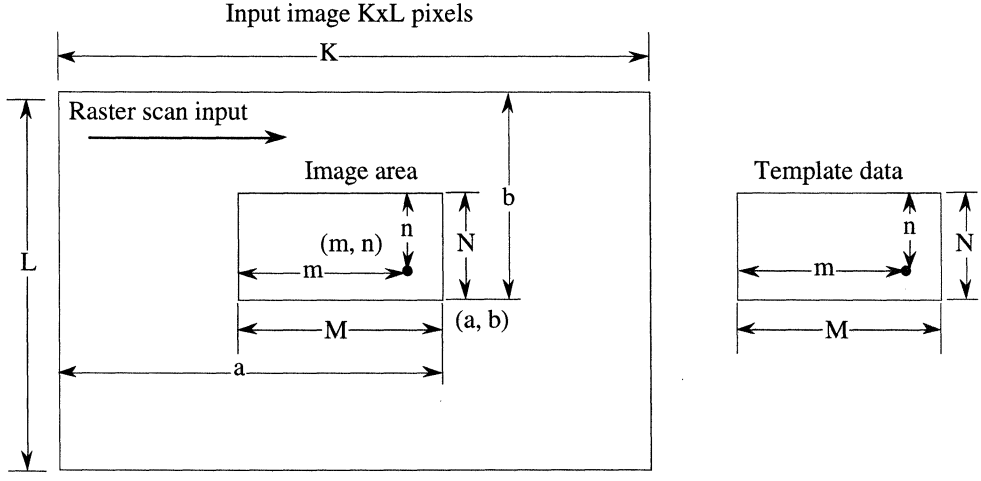
[Supplementary Functions]

- Field controls can be used to provide matched coordinate measurement for one field only. Also, the VSEN* pin can be used to control separate enable/disable settings for the field start signal VS*, and thus synchronize multiple image processing procedures field by field.
- The IP90C08 provides two sets of template registers, and allows access to the non-selected register at all times, even during processing. The template can be switched through the template register set selection pin TMPSEL, or through the selection register.
- The IP90C08 also includes these easy-to-use features:
  - -- CPU interface in either byte (8-bit) or word (16-bit) units
  - -- Single 5-volt power supply, with low-demand CMOS process
  - -- Input and output are TTL-level compatible, with CMOS-level output
  - -- 160-pin QFP (molded unit 28 x 28 mm, with a pin pitch of 0.65 mm)
  - -- Operating temperature range T_a = 0° to 70°C

1.2 Algorithms

Template matching is the process of searching raster-scan input images for the areas within those images that most closely match the pixel values stored in the IP90C08's internal template data registers. Usually, the closeness of the match is determined by comparing the binary images pixel by pixel.

The IP90C08 also expands this method to provide template matching of grayscale image data. This involves finding the pixel-to-pixel differences between the incoming grayscale image and the template data (which is stored as grayscale image data on the IP90C08), then calculating the sum of these differences. This sum indicates how closely the input image matches the template. Equation (1) shows a calculation using an input image and template data located at the same position within an image area.



$$S(a,b)=\sum_{n=0}^{N-1}\sum_{m=0}^{M-1}|I(a+m-M,\ b+n-N)-T(m,n)| \quad \text{..... (1)}$$

where:

- (a,b) are the coordinates of the lower right corner of the image area within the input image
- S(a,b) is the sum of differences with relation to the image area of M x N pixels having its lower right corner at (a, b)
- I(a + m – M, b + n – N) is the pixel grayscale value of the point with coordinates (m, n) within the image area
- T(m, n) is the image data for the corresponding template point at coordinates (m, n)

The lower the sum-of-differences is, the more closely the template data matches the image area. The IP90C08 calculates the sum-of-differences with respect to the input image, and outputs the result at the OD pin. This output can be further augmented using output compensation functions (including addition of bias values or division by means of a shift operation), which allows the IP90C08 to act as a matched filter or average-smoothing filter, and to create output images constructed by mapping the sum-of-difference values.

The IP90C08 can also locate the coordinates having the lowest sum-of-differences values (which give the positions with the best fit), rank these sets of coordinates in order of closeness-of-fit, and save them and their corresponding output values in the matching coordinates register.

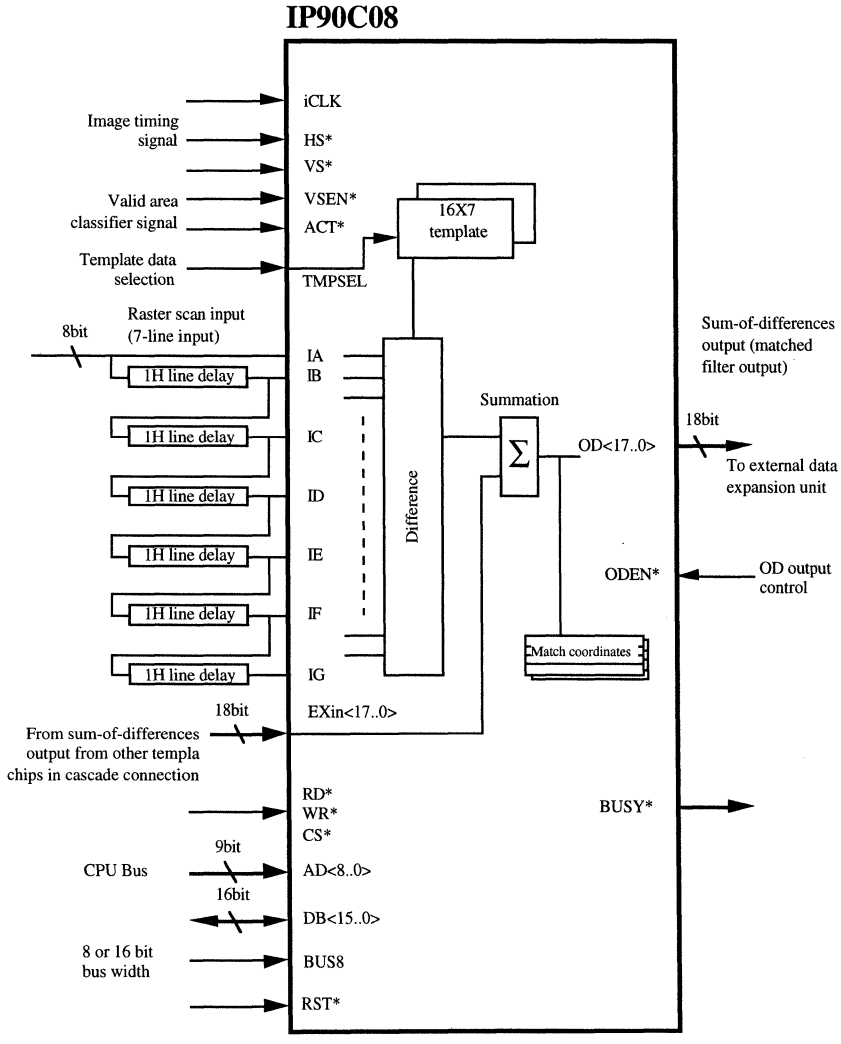
The IP90C08 contains template data in two sets of template registers. This allows the chip to load one set of template data while already performing template matching on another set, so that the matching process can continue without interruption. The IP90C08 can perform template matching for areas as large as 16 x 7 pixels; however, using multiple IP90C08s can expand the allowable size of the image area and the corresponding template data. (For further information, see Section 2.4, "Sample Expanded System Configurations.") Conversely, when processing areas smaller than 16 x 7 pixels, the template area enable register can be used to exclude selected pixels from processing.

Images to be searched enter the IP90C08 as data columns (with a maximum height of 7 pixels) in the direction of the raster scan, the same as would normal input to a 16 x 7 pixel filter. For details on the configuration of systems using this type of input, see the following section.

Section 2. Sample System Configuration and Operation

2.1 Basic System Configuration

The following diagram shows the configuration of a basic system using the IP90C08. Image data enters as raster scan input, in the form of single data columns of seven rows each. (The example also includes a line delay memory.) The images are then compared in real time to two internally stored template frames.



Note 1: An asterisk (*) following a pin symbol indicates negative logic.

Note 2: This block diagram does not include all functions of the IP90C08. Refer to appropriate sections of this manual for information on specific functions, timing, etc.

2.2 Operating Overview

2.2.1 Image Synchronization and Control Signals (VS*, HS*, ACT*)

This section outlines the operation of the image synchronization and control signals as used by the IP90C08's image input and output systems. For specific details about timing, see Section 7.5, "AC Characteristics." For an overview of pin functions, see Section 4, "Pins and Functions."

2.2.1.1 Operations Synchronized with the Pixel Clock

The IP90C08's template-matching circuits are designed to operate in synchronization with the pixel clock signal (iCLK). Other functions, however, are not synchronized with the pixel clock: these include selection of template data settings and CPU bus signals for internal processing settings.

The image data input signal (IA<7..0> through IG<7..0>), the image synchronization and control signals (HS*, VS*, VSEN*, ACT*, TMPSEL, RST*), and any expanded-area sum-of-differences input (EXin<17..0>) are latched with an input flip-flop at the rise of the pixel clock signal. The sum-of-differences output signal (OD<17..0>, which is the output signal for image data) and the control signal (BUSY*) output are timed to the rise of the pixel clock signal, with a delay for external load drive.

An asterisk (*) after a signal name indicates inverse logic.

2.2.1.2 Raster Scan Input

Image data is input to the IP90C08 as a simultaneously scanned signal of 7 rows of 8-bit image data, timed to the pixel clock, oriented in the direction of the raster scan, and input through the image data input pins (IA<7..0> through IG<7..0>). This signal enters the pipeline configuration of the IP90C08 for throughput processing in real time, where it can be used for template matching processing.

2.2.1.3 Image Synchronization Signals for Frame Recognition (VS*, HS*)

Image data entering as raster scan signals must be reconstructed in two dimensions to provide a frame from which patterns can be recognized. This process uses the vertical and horizontal synchronization signals VS* and HS*. VS* initiates the frame, and functions much like the vertical sync (VS) signal in a television system (and thus its name). HS* is the line initiating signal, and functions much like the horizontal sync (HS) signal in a television system.

The composition of a frame of image data begins when VS* cycles low, and each line within the frame begins when HS* cycles low. A frame of image data consists of the data input during the pixel clock cycles from one VS* signal to the next. Similarly, a line of data consists of the data input during the pixel clock cycles from one HS* to the next. Note that in this system, any data input simultaneously with VS* or HS* (data latched with the rising edge of the same pixel clock cycle as VS* or HS*) is invalid, and the data latched with the next cycle is considered as the first image data in that respective frame or line.

For internal processing, the IP90C08 uses VS* and HS* as the initial loading signals for the Vcnt and Hcnt counters, respectively. Vcnt and Hcnt recognize the vertical and horizontal coordinates of an image. VS* acts as the initial loading signal for the vertical-axis coordinate counter (Vcnt), setting the initial value of Vcnt with respect to the initial vertical base coordinate of the frame (Vbase). Together with the active area signal (ACT*, described in the following section), this VS* controls the start of the template matching process. HS* is not used to recognize image area, but acts as the initial loading signal for the horizontal-axis coordinate counter (Hcnt), setting the initial value of Hcnt with respect to the initial horizontal base coordinate of the row (Hbase). The processing area is controlled by ACT*. For further description of the IP90C08's internal configuration, see Section 3, "Internal Configuration."

VS* and HS* are inverse logic signals, and are recognized at the rise of the pixel clock. Both are simultaneously differentiated within the IP90C08, and become effective when a low-level signal on the clock rise is detected following a high-level signal. Because of this simultaneous differentiation, the low phase (low pulse width) of VS* and HS* can be any length, as long as they return to a high level before the next active phase. System timing is described further in the following section.

2.2.1.4 Active Area Signal (ACT*)

The ACT* signal indicates that data entering the IP90C08 is valid for processing. The first VS* signal that enters after the start of template matching (indicated by the internal execute register having been set) is interpreted as the start of a frame to be processed. (VSEN* is also low at this time.) ACT* is used when this frame enters the IP90C08 system, and indicates whether the local area is to be used for pixel-to-pixel template matching. The IP90C08 applies template matching processing to the 16 x 7-pixel template matching area that ends with the pixel column that is input when ACT* is taken low. Pixels entering while ACT* is high are ignored, and no record is kept of match coordinates for 16 x 7-pixel template matching areas that end with pixel columns input while ACT* is high. However, pixels entering while ACT* is high are still considered constituents of 16 x 7-pixel template matching areas, and are retained in internal shift registers until ACT* cycles low, at which time they are used for template matching.

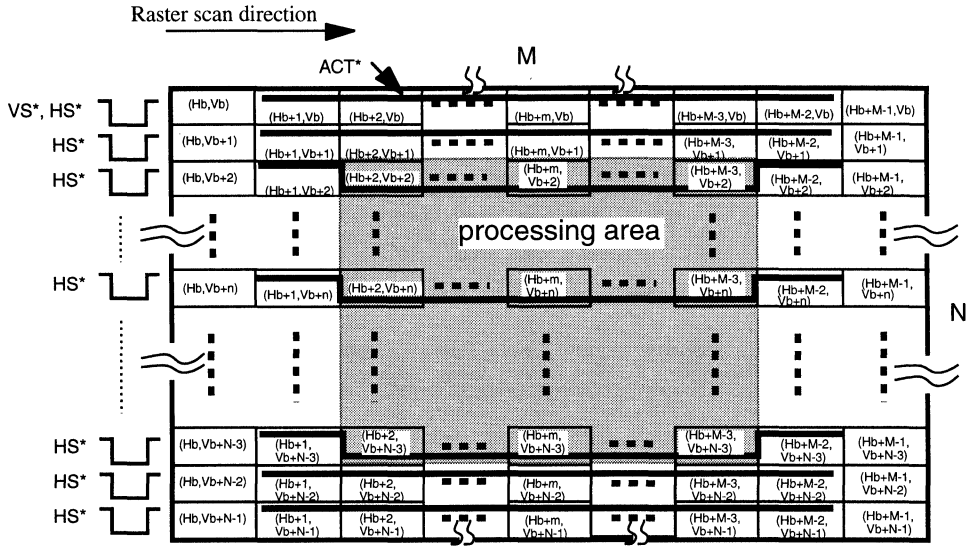
The IP90C08 does not send VS* at the end of a frame. Frame processing should be executed continuously by switching between templa register sets A and B. To end processing after the input of one frame only, set the field control flag 'fl' at bit 1 of the execute register to 1 before execution, or configure the logic circuits to hold ACT* high after the input of one frame. The IP90C55 (IMSC) chip has a function that sends a rectangular-area-valid signal (AOI-n*) for one frame only within the image space; this function can also be used in combination with these features.

HS* has no effect on whether incoming data is processed.

2.2.2 The VS*, HS*, and ACT* Signals, and Spatial Coordinates

The following figure shows the relationship between VS*, HS*, and ACT*, and the coordinates of individual pixels.

The incoming image (M x N) is input as pixel data in a raster scan image synchronized by iCLK.



Hb: horizontal coordinate base register value (Hbase)
Vb: vertical coordinate base register value (Vbase)

- Horizontal Coordinates

  The change of HS* from high to low is detected during the rise of iCLK, and the internal synchronous differentiation process generates a pulse signal HSpls having one clock width. HSpls becomes the loading signal for Hcnt (which generates horizontal coordinates), and the Hbase coordinate becomes the horizontal coordinate of the image data IA acquired on the iCLK rise following the rise at which the change of HS* from high to low was detected. Hcnt then begins to count upward with each iCLK signal.

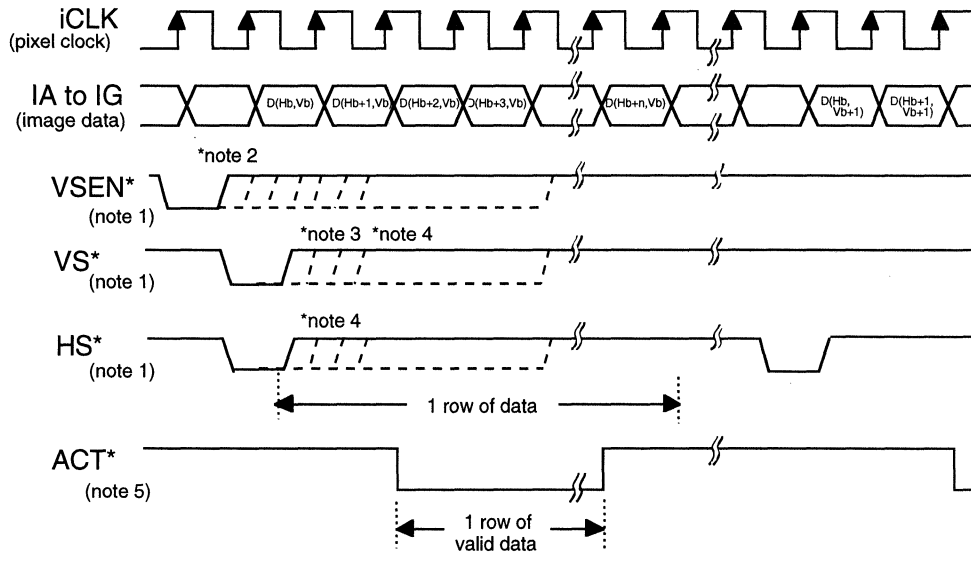
- Vertical Coordinates

  The change of VS* from high to low is detected during the rise of iCLK, and the internal synchronous differentiation process generates a pulse signal VSpls having one clock width. VSpls becomes the loading signal for Vcnt (which generates vertical coordinates), and the Vbase coordinate becomes the vertical coordinate of the image data IA acquired on the next iCLK rise following the rise at which the change of VS* from high to low was detected. Hcnt then begins to count upward with each HSpls.

- The IP90C08 processes image data that is input while ACT* is low (when its execute flag is enabled).

- As shown above, ACT* should cycle to active (low) to define the active portion of each line in the processing area. ACT* should remain high for areas not intended for processing.

2.2.3 Incoming Image Data Transfer Format



Note 1: An asterisk (*) following a signal name indicates negative logic. VS* starts the definition of an image field, VSEN* validates the VS* input, and HS* starts each row in the image.

Note 2: VS* is validated if VSEN* is low during the rise of iCLK one clock cycle before VS* changes to low. VSEN* facilitates synchronization with other image processing LSIs, image sequencers, and similar devices, and should always be kept low when synchronization with other devices is not needed.

Note 3: VS* and HS* need not be input simultaneously. If HS* precedes VS*, it will be ignored. If HS* is input simultaneously with VS*, it will be recognized.

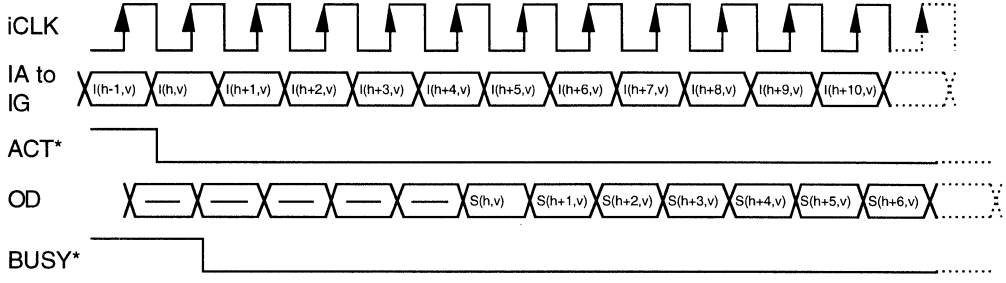
Note 4: Because VS* and HS* are synchronously differentiated by the internal clock signal, they need not be input in pulse form. Any time VS* or HS* is detected at the rise of iCLK, the system operates as though VS* or HS* has been input. After a VS* or HS* low is detected, the signal must return to high until the next VS* or HS*.

Note 5: ACT* should remain low throughout the valid data interval on each row.

2.2.4 OD and BUSY* Signal Output Timing

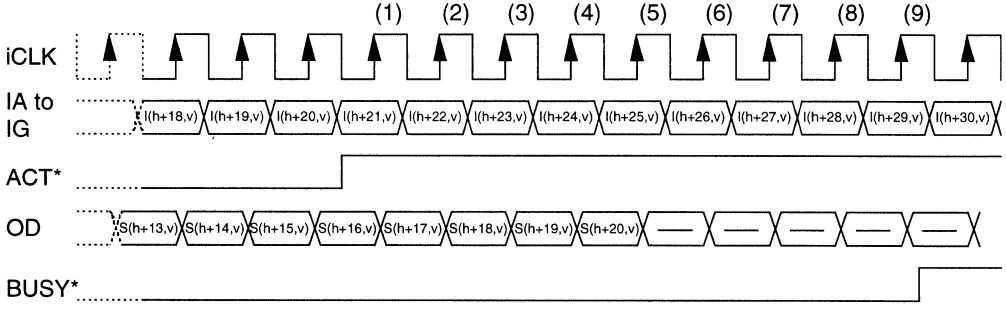
ACT* determines whether matching coordinate measurement is applied to image data entering via the IA through IG signals. In the example below, ACT* is low when input data I(h,v) enters, thereby initiating matching coordinate measurements for the 16 columns of data from I(h-15,v) to I(h,v) using sum-of-differences values (with respect to the template). The sum-of-differences output signal OD can also be used to constantly output the results of calculations on the most recent 16 x 7-pixel image data input, without referring to ACT*.

The results of sum-of-differences processing of incoming image data is delayed four clock cycles before being output through OD. This assumes that the sum-of-differences delay register value is 0, and that the output compensation circuit is inactive (that is, the I/O control register IOCTRL has its output compensation circuit enable flag "se" set to 0). BUSY* indicates that the LSI internal matching coordinate measurement is in progress. In the illustration below, BUSY* returns to high nine clock cycles after the rise of the signal in which ACT* goes high.



Continued below...

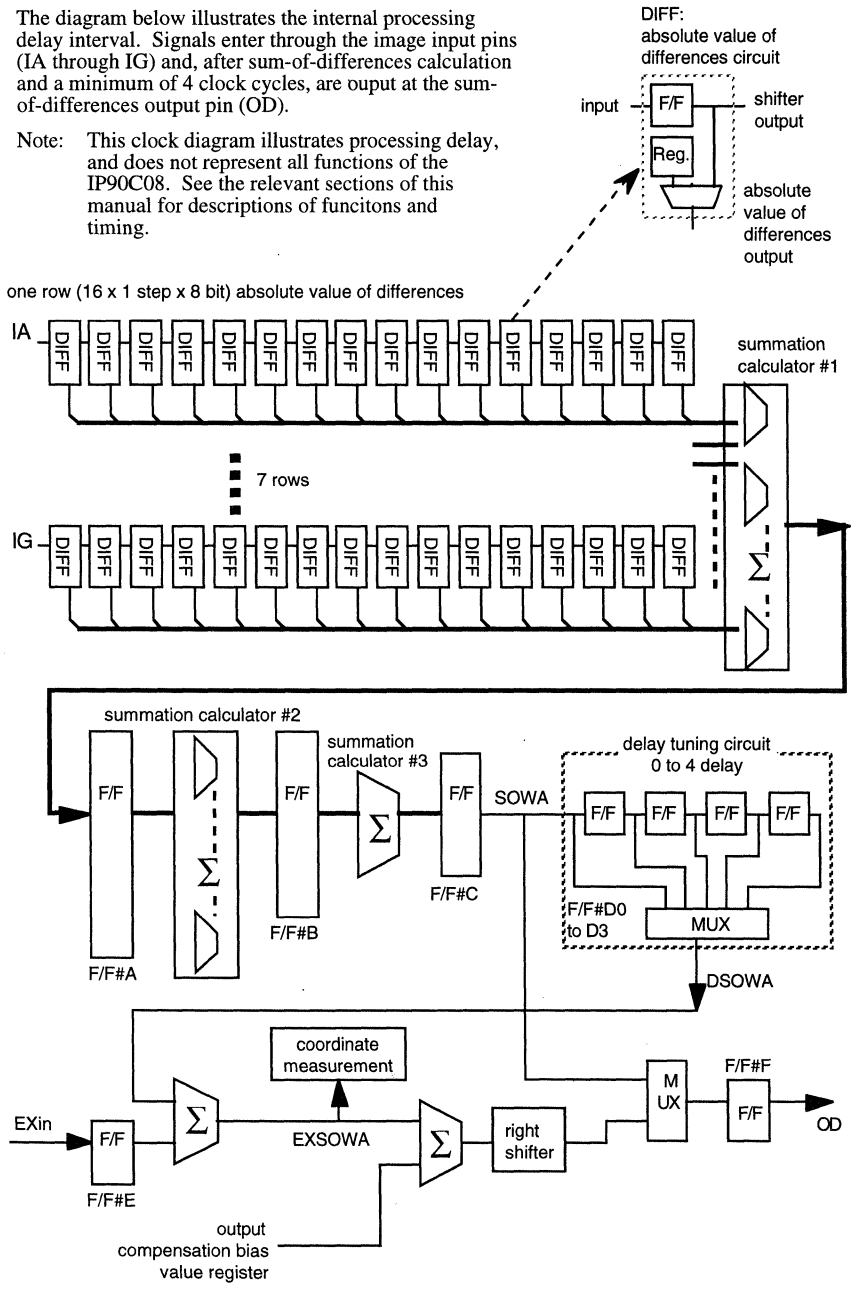
...Continued from above



2.2.5 Processing Delay

The diagram below illustrates the internal processing delay interval. Signals enter through the image input pins (IA through IG) and, after sum-of-differences calculation and a minimum of 4 clock cycles, are ouput at the sum-of-differences output pin (OD).

Note: This clock diagram illustrates processing delay, and does not represent all functions of the IP90C08. See the relevant sections of this manual for descriptions of funcitons and timing.



2.2.6 The Active Area Signal (ACT*) and Measurement Areas

The following illustrations show examples of how the active area signal (ACT*) can be used to select areas of the input image for matching coordinate measurement. Each example (sequence a) through (sequence n) assumes that image data enters as raster scan input, and shows the scanning pattern of a 16 x 7 image area as well as the time sequence of signals used to validate the matching coordinate measurement area. Thin diagonal lines indicate the measurement area (the area of the input image in which measurements are valid), and thick diagonal lines indicate the image area being scanned (the area stored in the internal shift register). The IP90C08 accepts simultaneous external feed of seven pixels of data (one column of seven rows), and stores this data in an internal shift register as an 'image area.' The IP90C08 then compares this 16-column x 7-row x 8-bit image data in the shift register with a previous set of template data, and performs differential, absolute value, and sum-of-difference calculations.

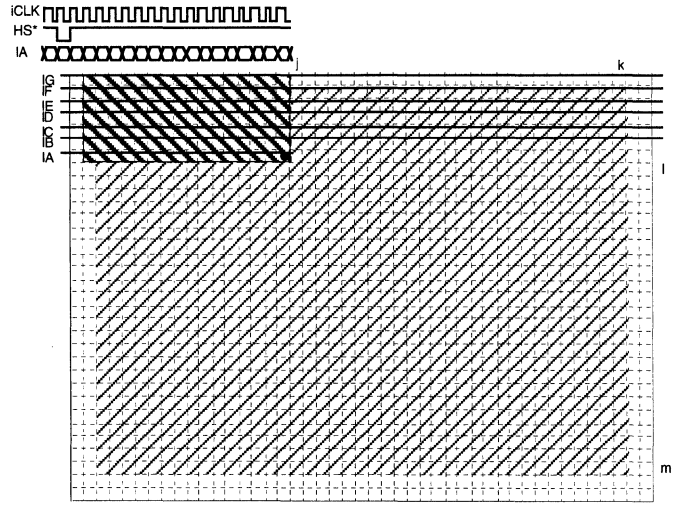
Remember the following points when referring to these examples:

- The IP90C08 constantly accepts input data and makes sum-of-differences calculations.
- ACT* is set low to control matching coordinate measurement.
- The relationship between ACT* and matching coordinate measurement is as follows. When ACT* low is detected, matching coordinate measurement is performed on the image area stored in the internal shift register. Thus, the valid data for this measurement includes not only the 7-row data input at the time ACT* low is detected, but also the previous 15 columns of 7-row data that was input to the internal shift register before ACT* low was detected. The single column input when ACT* low is detected is then added to the previous 15 columns of data, and matching coordinate measurement is performed on all 16 columns.

Sequence A

IA: (j-1,l-1). This image area does not fall within rows in which matching coordinate measurement can be applied.

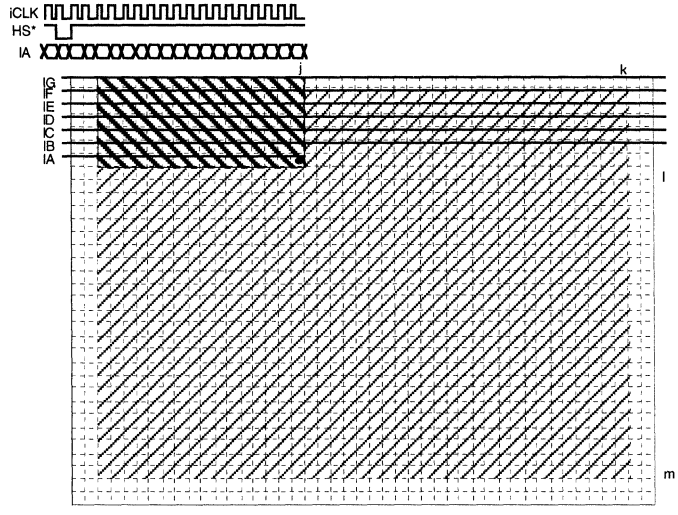
One portion of the image area (the top row and left-hand column of the image area, shown in heavy diagonal lines) is located outside the measurement area, so ACT* (the solid horizontal lines) remains high. ACT* corresponding to the position of the black dot (j-1,l-1), is high.



Sequence B

IA: (j,l-1). This image area does not fall within rows in which matching coordinate measurement can be applied.

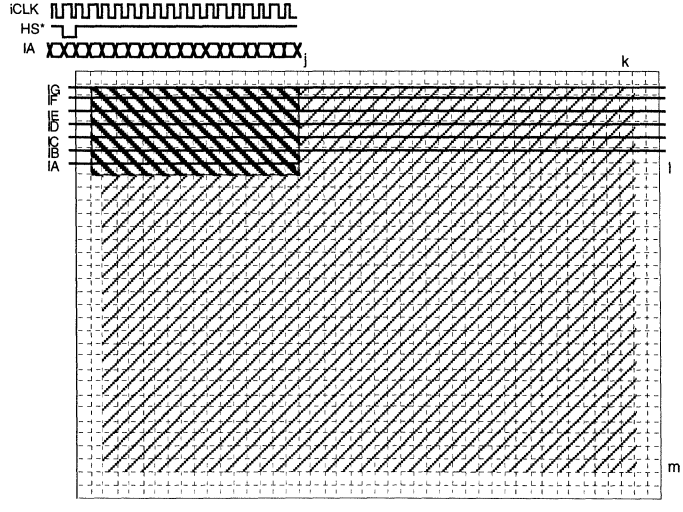
One portion of the image area (the top row of the image area, shown in heavy diagonal lines) is located outside the measurement area, so ACT* remains high.



Sequence C

IA: (j-1,l). The image area falls within rows covered by the measurement area.

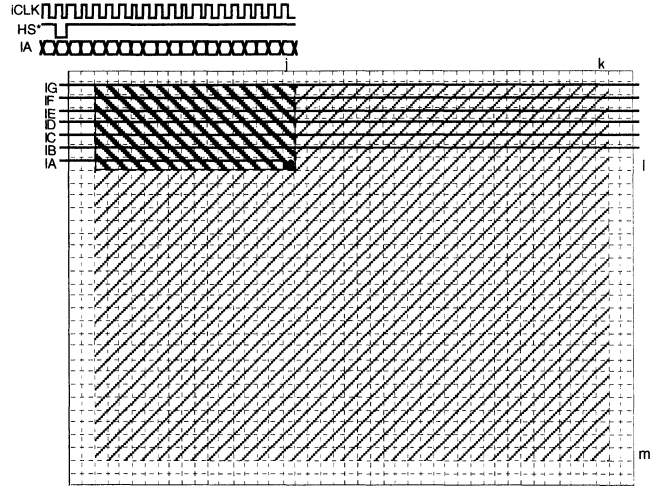
The image area is scanned within the rows in which matching coordinate measurement is feasible, but is still not entirely within it. One portion (the left edge of the image area, shown in heavy diagonal lines) still lies outside the measurement area, so ACT* remains high.



Sequence D

IA: (j,l). The image area falls completely within the matching coordinate measurement area.

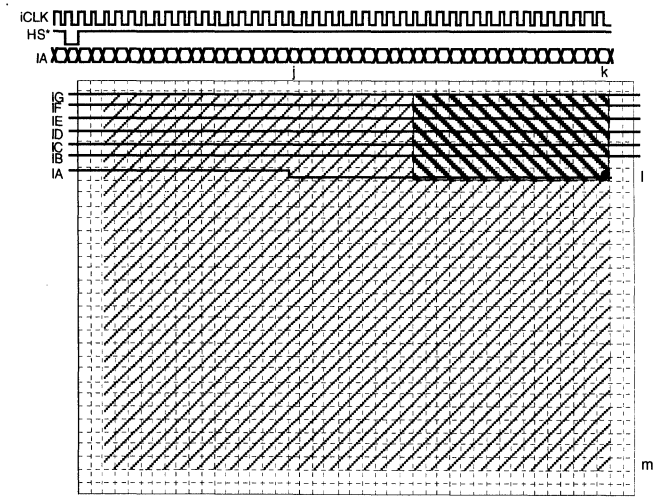
The entire image area is within the measurement area, and ACT* is asserted low. The image area (shown in heavy diagonal lines) having this pixel in its lower right corner is stored in the internal shift register, and the result of a sum-of-differences calculation for this area is used for matching coordinate measurement. The matching coordinate measurement circuit takes the sum-of-differences value from this first measurement as the best available match (the lowest sum-of-differences), and enters the sum-of-differences value and the coordinates (j,l) in the register MINPOS(0).



Sequence E

IA: (k,l). The image area is at the right edge of the matching coordinate measurement area.

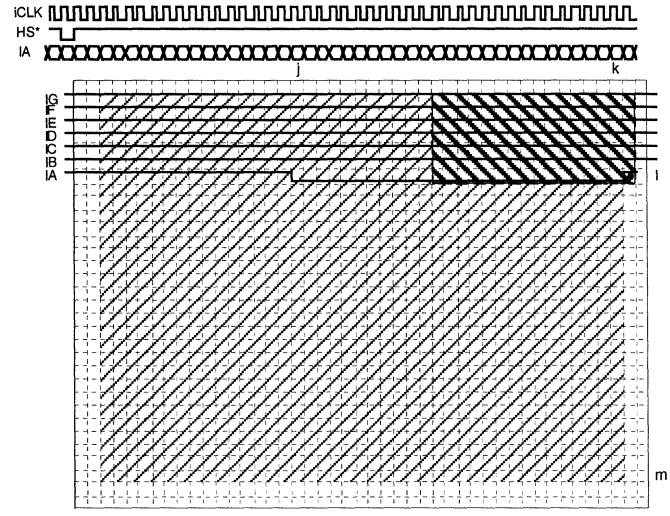
Here, the image area is moved to the right edge of the area of the frame, but is still entirely within the measurement area. Therefore, ACT* is asserted low.



Sequence F

IA: (k+1,l). The image area extends beyond the matching coordinate measurement area.

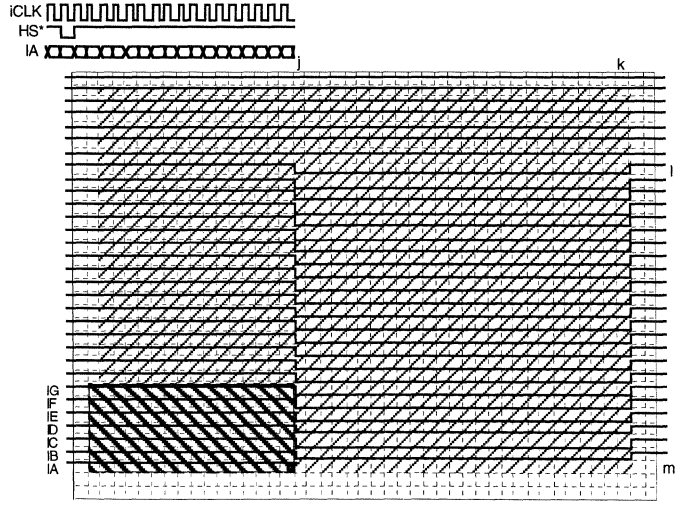
The image area extends beyond the right edge of the area of the frame and outside the measurement area. ACT* is therefore de-asserted and remains high.



Sequence G

IA: (j-1,m). The image area falls within the last rows of the matching coordinate measurement area.

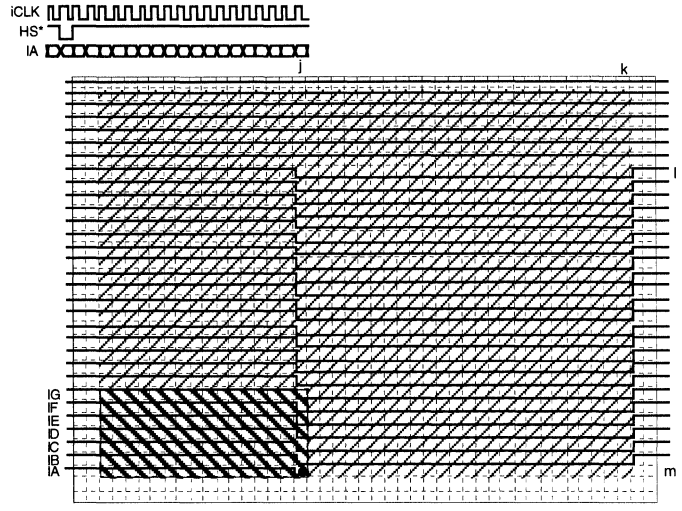
The image area has moved to the lowest row of the matching coordinate measurement area. Because measurements can still be made in this row, the position is valid for vertical measurement; however, since not all of the image area fits horizontally within the measurement area, ACT* is de-asserted and remains high.



Sequence H

IA: (j,m). The image area falls within the last rows of the matching coordinate measurement area.

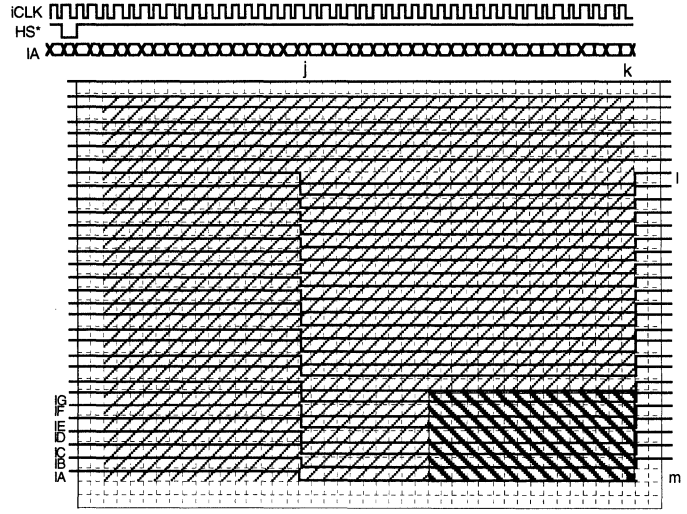
The image area is scanned within the lowest rows in which matching coordinate measurement is feasible. In this position, the entire image area falls within the measurement area, so ACT* is asserted low.



Sequence I

IA: (k,m). The image area is located at the last pixel of the matching coordinate processing area.

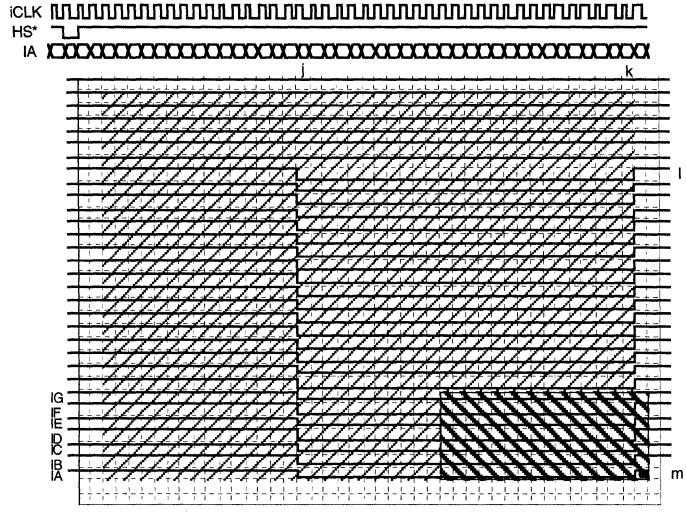
Here, the image area has moved to the last pixel at the lower right corner of the measurement area. This area is still measurable, so ACT* is asserted low.



Sequence J

IA: (k+1,m). The image area attempts to measure the entire matching coordinate measurement area.

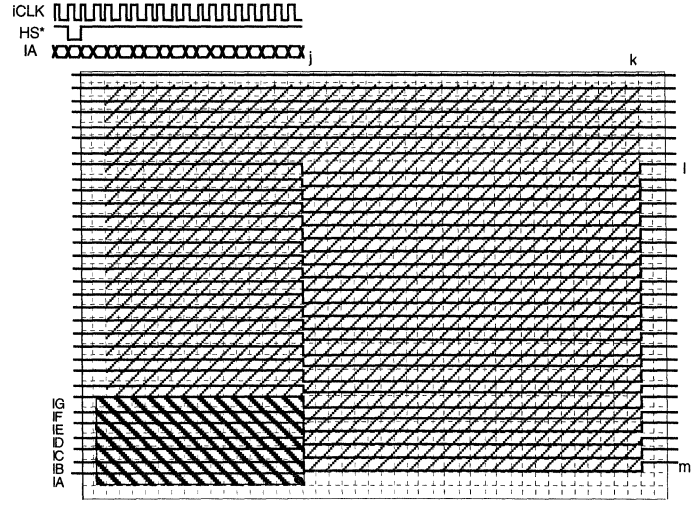
The image area is positioned after the last pixel in the lower right corner of the measurement area. This requires scanning areas not previously measured, so ACT* is de-asserted and remains high.



Sequence K

IA: (j-1,m+1). The image area is located on the row following the last row of the matching coordinate measurement area.

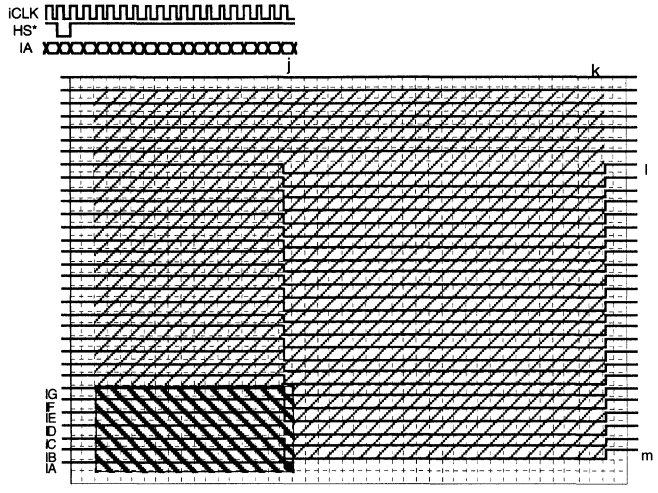
The image area is positioned on the row following the last row of the measurement area. This requires data input from a line not previously measured, and therefore ACT* is de-asserted and remains high.



Sequence L

IA: (j,m+1). The image area is located on the row following the last row of the matching coordinate measurement area.

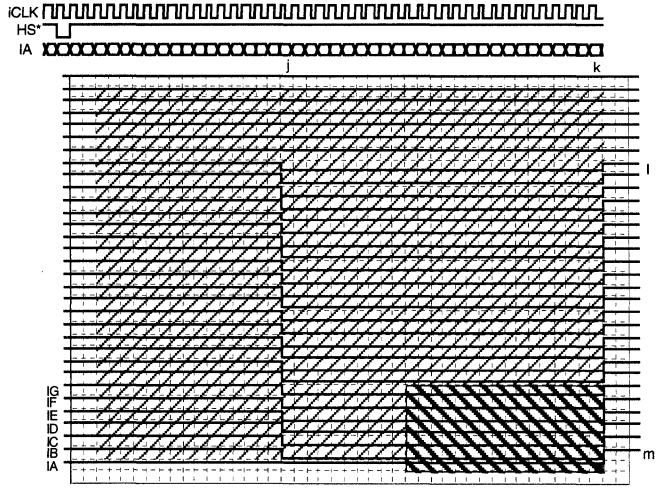
The image area is positioned on the row following the last row of the measurement area. The data simultaneously input through pins IB–IG still falls within the measurement area, but the entire image area involves areas not previously measured, so ACT* is de-asserted and remains high.



Sequence M

IA: (k,m+1). The image area is located on the row following the last row of the matching coordinate measurement area.

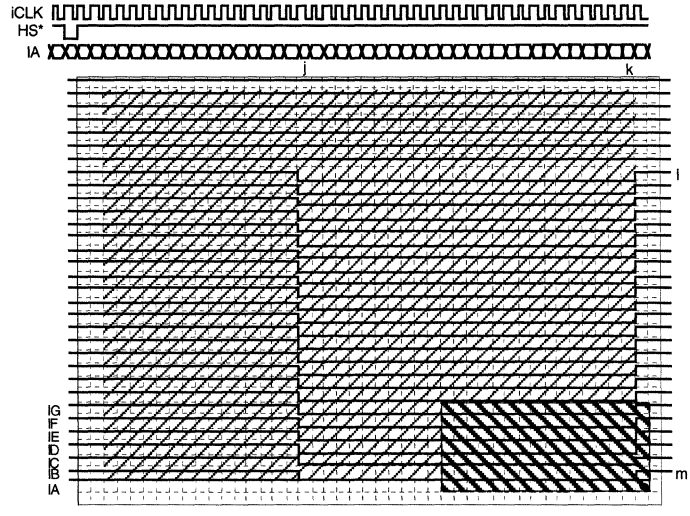
The image area is positioned on the row following the last row of the measurement area. The data simultaneously input through pins IB–IG still falls within the measurement area, but the entire image area involves areas not previously measured, so ACT* is de-asserted and remains high.



Sequence N

IA: (k+1, m+1). The image area is located on the row following the last row of the matching coordinate measurement area.

The image area is positioned on the row following the last row of the measurement area. The data simultaneously input through pins IB-IG still falls within the measurement area, but the entire image area involves areas not previously measured, so ACT* is de-asserted and remains high.



2.3 Operating Setup Procedures

2.3.1 One-Time Setup

0. Power ON.
   ↓
1. Hardware reset.
   Set RST* pin input low, then return to high and remain high.
   ↓
2. Set operating mode (note 1).
   ↓
3. Set template registers.
   - Template area enable register (xTMPEN) settings
   - Template data register (xTMP) settings

   ↓
4. Clear matching coordinate register (note 2).
   ↓
5. Write 1 to execute register execute flag, start execution of matching coordinate measurement.
   Load base coordinate into V coordinate counter Vcnt from VS*, and load base coordinate into H coordinate counter Hcnt from HS*.
   ↓
6. Input image data (while ACT* is low).
   Process image data input for template matching while ACT* is low.
   ↓
7. Verify the end of the first image data input.
   After the image is inputted, verify the internal processing delay cycle (or verify that BUSY* is high, indicating the end of processing).
   ↓
8. Write 0 to the execute register execute flag, and end execution of coordinate matching.
   ↓
9. Read the match coordinate register.
   ↓
10. End or repeat.
    To repeat processing, repeat from step 2 (or 3 if the mode is unchanged). If template data is unchanged, repeat from step 4.
    -> To step (2), (3) or (4)

Note 1: Operating mode settings:

- Soft reset from reset register
- Template selection register
- I/O control register
- Shifter connection register
- Sum-of-differences delay register
- Output compensation bias register
- Output compensation shift register
- Horizontal base coordinate register
- Vertical base coordinate register

Note 2: The matching coordinate clear register clears the matching coordinate register. (To the "cla" or "clb" flag, write 1 after clear, write 0 to cancel clear.) The previous match coordinates remain in memory until the clear procedure is executed, so results can be read any time. Always clear the register before making new measurements.

2.3.2 Setup for Continuous Processing with New Templates at Each Measurement

0. Power ON.
   ↓
1. Hardware reset.
   Set RST* pin input low, then return to high and remain high.
   ↓
2. Set operating mode (note 1).
   ↓
3. Set up template register set A.
   - Template area enable register set A (ATMPEN) settings
   - Template data register set A (ATMP) settings

   ↓
4. Clear matching coordinate register set A.
   The matching coordinate clear register clears the matching coordinate register.
   To the "cla" flag, write 1 after clear, write 0 to cancel clear.
   ↓
5. Execute.
   - Write 1 to execute register to start execution

   ↓
6. Input the first image data, run template matching processing.
   Process template register set A, only while ACT* is low.
   ↓
7. Set up template register set B during template matching processing with template register set A (note 2).
   - Template area enable register set B (BTMPEN) settings
   - Template data register set B (ATMP) settings

   ↓
8. Verify the end of the first image data input.
   After the image is inputted, verify the internal processing delay cycle (or verify that BUSY* is high, indicating that processing has ended).
   ↓
9. Switch template register sets.
   Using the template selection register or the TMPSEL pin, switch the template register set used for processing from set A to set B.
   ↓
10. Input the second image data, and run template matching processing (set B).
    Process template register set B, only while ACT* is low.
    ↓
11. Read matching coordinate register set A during processing with template register set B.
    ↓
12. Clear matching coordinate register set A.
    - The matching coordinate clear register clears the matching coordinate register.
      To the "cla" flag, write 1 after clear; to the "cla" flag, write 0 to cancel clear.

    ↓

13. Set up template register set A

    Set up template register set A, during template matching processing with template register set B.

    - Template area enable register set A (ATMPEN) settings
    - Template data register set A (ATMP) settings

    ↓

14. Verify the end of the second image data input.

    After the image is inputted, verify the internal processing delay cycle (or verify that the BUSY* pin signal is high, indicating that processing has ended).

    ↓

15. Switch template register sets.

    Using the template selection register or the TMPSEL pin, switch the template register set used for processing from set B to set A.

    ↓

16. To repeat, continue from step 17.

    → To end, go to step 24.

    ↓ To continue processing, go to step 17.

17. Input the third image, and run template matching processing.

    Process template register set A, only while ACT* is low.

    ↓

18. Read matching coordinate register set B (BMATCH) during template matching processing with template register set A.

    ↓

19. Clear matching coordinate register set B.

    - The matching coordinate clear register clears the matching coordinate register.

      To the "clb" flag, write 1 after clear, write 0 to cancel clear

    ↓

20. Set up template register set B during template matching processing with template register set A.

    - Template area enable register set B (BTMPEN) settings
    - Template data register set B (BTMP) settings

    ↓

21. Verify the end of the third image data input.

    After the image is inputted, verify the internal processing delay cycle (or verify that BUSY* is high, indicating the end of processing).

    ↓

22. Switch template register sets.

    Using the template selection register or the TMPSEL pin, switch the template register set used for processing from set A to set B.

    ↓

23. To repeat, return to step 10. To end, go to step 24.

    → To repeat, go to step 10.

    ↓ To end, go to step 24.

24. End processing.

    - Write 0 to the execute register to end processing.

    ↓

25. Read matching coordinate register set A.
    ↓
26. End Processing.
    For power-saving mode, write 1 to the appropriate flags in the I/O registers.

Note 1 Operating mode settings:

- Soft reset from reset register
- Template selection register
- I/O control register
- Shifter connection register
- Sum-of-differences delay register
- Output compensation bias register
- Output compensation shift register
- Horizontal base coordinate register
- Vertical base coordinate register

Note 2 Switching between template register sets A and B.

The IP90C08 provides two built-in template register sets (the registers on which template matching is based). Template matching can proceed while switching between these two register sets as needed. For further information on the registers, see Section 5, "Registers." Register sets A and B function equivalently, and provide an efficient way to frequently switch template data during template matching, as illustrated in the above procedure.

Register sets A and B can be accessed independently, and so offer independent address mapping for reading and writing. Similarly, the matching coordinate registers AMATCH0–AMATCH3 and BMATCH0–BMATCH3 have different addresses, and can be accessed independently.

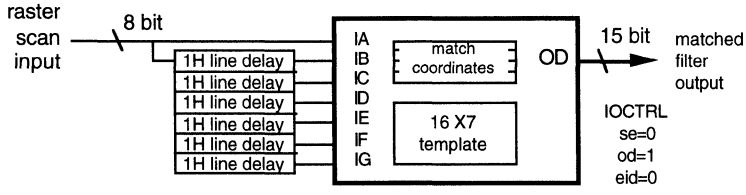
Switching between template registers sets is done through the contents of the template selection register (TMPSEL) and the TMPSEL pin. For information on the template selection register, see Section 5.12, "Template Select Register."

2.4 Sample Expanded System Configurations

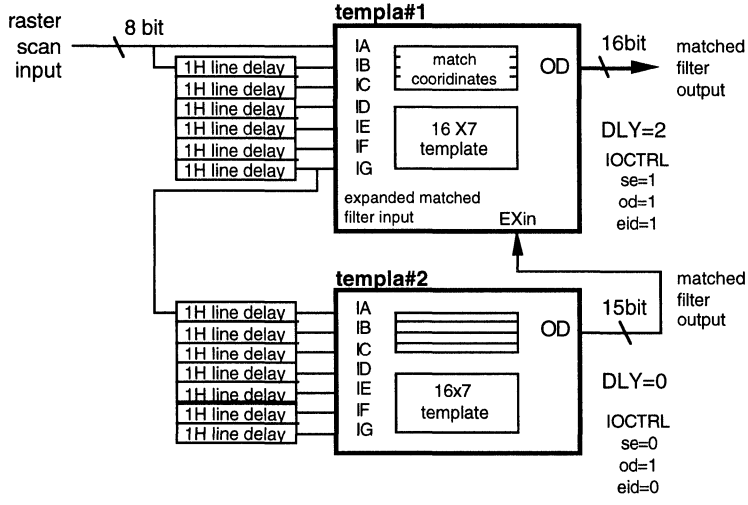
This section gives an example of using multiple IP90C08 chips for processing expanded image areas.

Note: The operation and functions of these sample systems are not warranted for actual use.

2.4.1 Template Matching and Match Filter Processing of a 16 x 7-Pixel Template



2.4.2 Template Matching and Match Filter Processing of an Expanded 16 x 14-Pixel Template

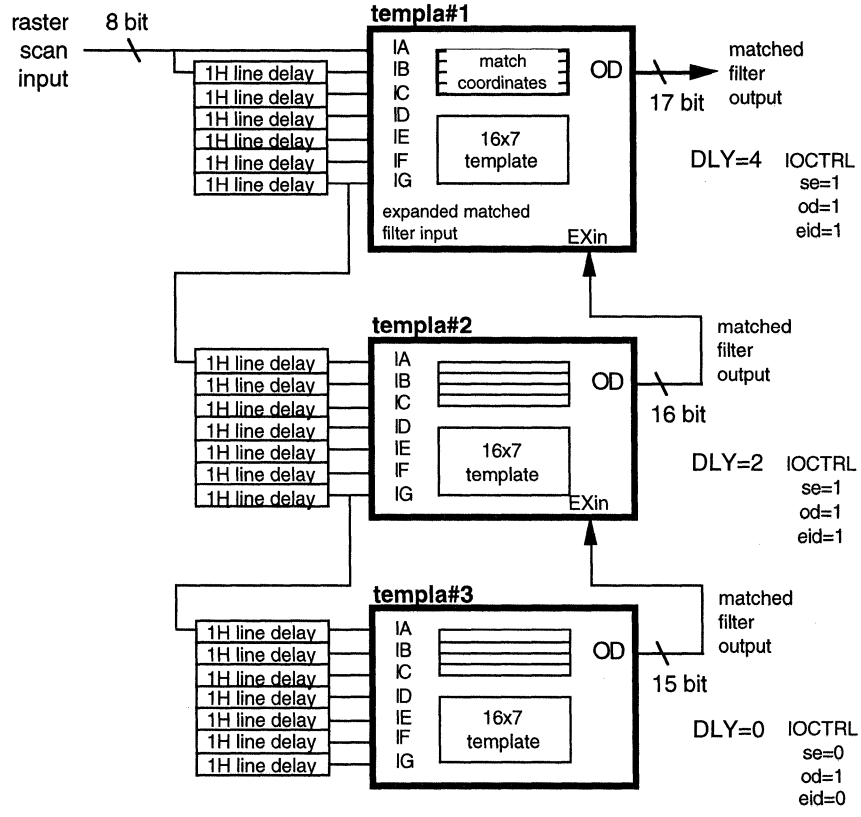


Matching coordinates are stored in the register of the templa#1 chip. In this example, the matching coordinate register of the templa#2 chip is ignored.

In the templa#2 chip, the output compensation circuit enable flag (se) in the I/O control register (IOCTRL) is set to 0. This reduces the output delay for sum-of-differences calculations in the templa#2 chip, and transfers the partial sum-of-differences value for its seven lines (16 x 7 pixels) to the templa#1 chip. In the templa#1 chip, this partial sum-of-differences value received from the templa#2 chip is added to the partial sum-of-differences value calculated by the templa#1 chip. The result is a total sum-of-differences value that is the result of template matching for the 16 x 14-pixel image area, and shows how closely the image matches the template.

The connection between the two IP90C08s delays the transfer of the partial sum-of-differences calculation from the templa#2 IP90C08 by two clock cycles, so the system must include a way to absorb this phase differential. This can be done by entering 2 in the sum-of-products delay register (DLY) to compensate for the delay in input from the expansion unit.

2.4.3 Template Matching and Match Filter Processing of an Expanded 16 x 21-Pixel Template



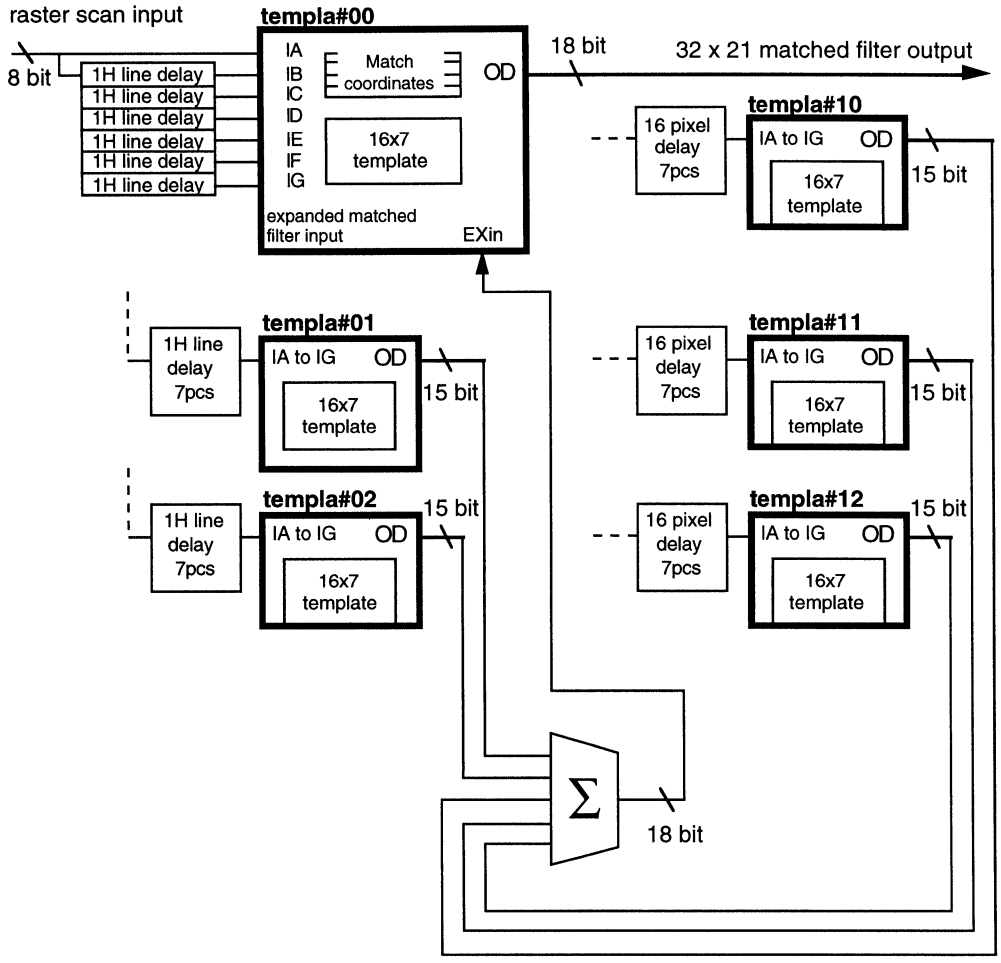
Matching coordinates are stored in the register of the templa#1 chip. In this example, the matching coordinate registers of the templa#2 and templa#3 chips are ignored.

The output compensation circuit enable flag (se) in the templa#3 chip's I/O control register (IOCTRL) is set to 0, reducing the output delay for sum-of-differences calculations in that chip, and transferring the partial sum-of-differences value for its seven lines (16 x 7 pixels) to the templa#2 chip. In the templa#2 chip, this partial sum-of-differences value is then added to the partial sum-of-differences value calculated by the templa#2 chip. This gives the result of template matching for a 16 x 14-pixel image area, which is then transferred to the templa#1 chip and added to the partial sum-of-differences value calculated by the templa#1 chip. The final value is the result of template matching for the 16 x 21-pixel image area, and indicates how closely the image matches the template.

The delay caused by using a cascade connection of multiple IP90C08s can be reduced by using the sum-of-products delay register (DLY). In the example above, the templa#3 chip's DLY would be set to 0, the templa#2 chip's DLY to 2, and the templa#1 chip's DLY to 4. This absorbs the phase differential between the respective sum-of-products calculations. Phase delay can also be reduced on the input side by using line memory or FIFO settings to change the length of the input line delay.

2.4.4 Template Matching and Match Filter Processing of an Expanded 32 x 21-Pixel Template

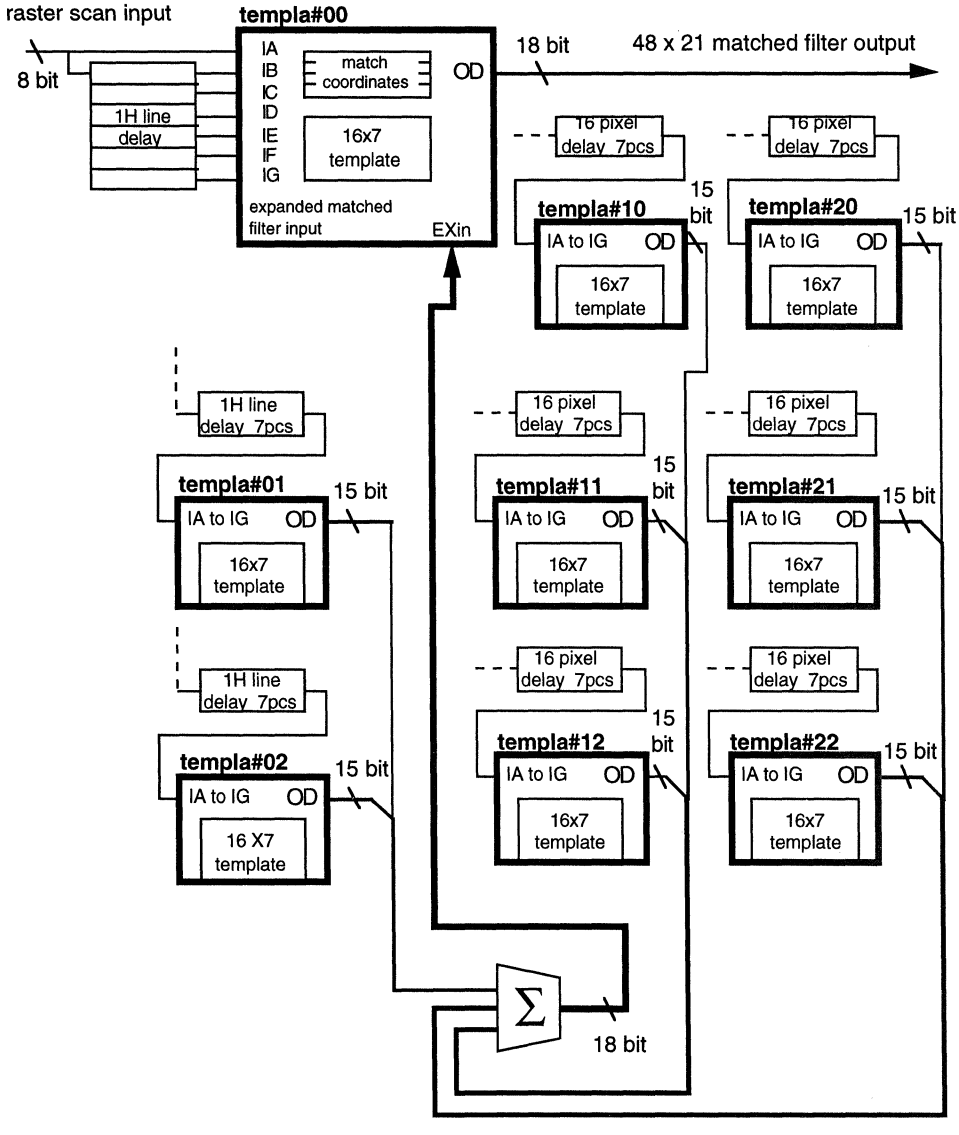
Matching coordinates are stored in the templa#00 chip.



Note: Image areas can be expanded horizontally by connecting the delay shift registers to the input pins (IA–IG) as shown above. However, the sum-of-differences output pins (OD) can also be connected to adjust for phase differential. The best method to use depends on the particular devices and applications.

2.4.5 Template Matching and Matching Filter Processing of an Expanded 48 x 21 Pixel Template

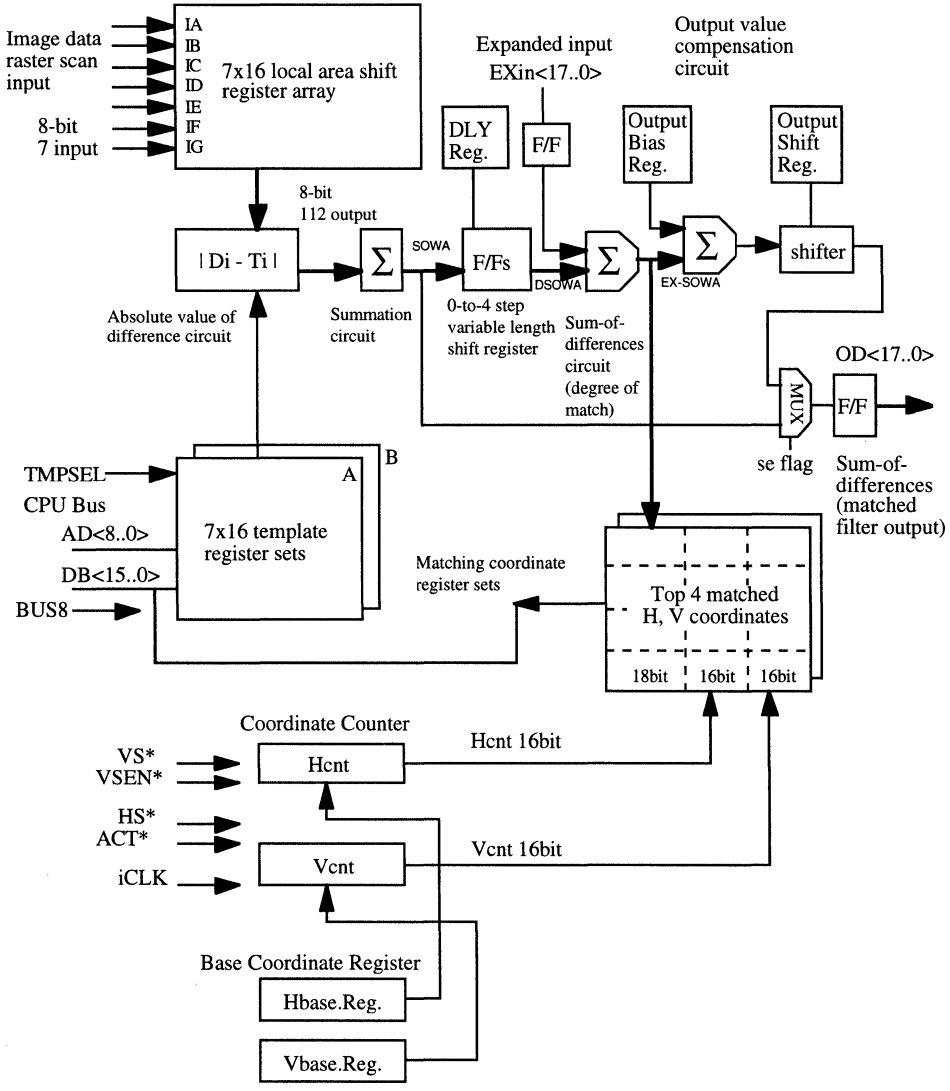
Matching coordinates are stored in the templa#00 chip.



Section 3. Internal Configuration

3.1 Internal Block Diagram

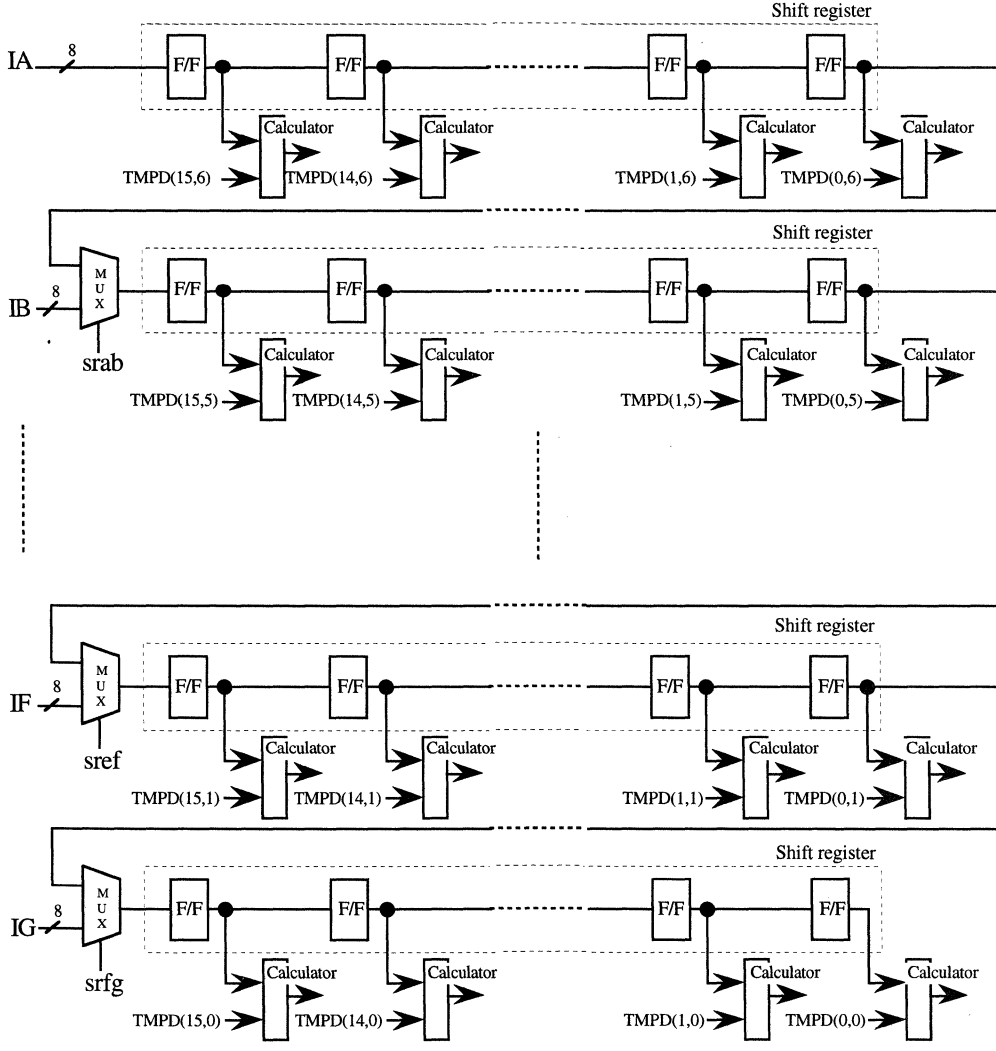
The following block diagram shows the configuration of the IP90C08.



Note: This block diagram does not show all functions of the IP90C08. For information on particular functions or other aspects of the IP90C08, see the appropriate sections of this manual.

3.2 Shift Registers and Template Data Registers

The illustration below shows the relationship between the shift register array, which holds the incoming image area data, and the template data registers.



Note: This block diagram does not show all functions of the IP90C08. For information on particular functions or other aspects of the IP90C08, see the appropriate sections of this manual.

Section 4. Pins and Functions

4.1 Pin Lists and Functional Descriptions

Signal group	Pin symbol	No.	I/O	Function	Description/Remarks
Image sync signals	iCLK	1	I	Pixel clock	
	HS*	1	I	Horizontal sync signal input	Starts every row
	VS*	1	I	Vertical sync signal input	Starts every field
	VSEN*	1	I	Vertical sync enable signal	Enables VS* input
	ACT*	1	I	Active area signal	L signal for processing
Image data input	IA<7..0>	8	I	8-bit input A, n^{th} line	Simultaneous input of 7 pixels of data, as one column of 7 rows.
	IB<7..0>	8	I	Input B, n+1 line	
	IC<7..0>	8	I	Input C, n+2 line	
	ID<7..0>	8	I	Input D, n+3 line	
	IE<7..0>	8	I	Input E, n+4 line	
	IF<7..0>	8	I	Input F, n+5 line	
	IG<7..0>	8	I	Input G, n+6 line	
Expanded sum-of-difference input	EXin<17..0>	18	I	Expanded sum-of-differences input	Expanded sum-of-differences input for template matching of up to 1024 total pixels. (note 1)
Sum-of-differences (matched filter) output	OD<17..0>	18	O	Sum-of-differences (matched filter) output	Sum-of-differences (matched filter) output based on sum-of-differences input
	ODEN*	1	I	OD output enable	Enables OD<17.0> output: -- If ODEN*=H: OD<17..0> is High-A (note 2) -- If ODEN*=L: OD<17..0> is drive (enable)
Template register select	TMPSEL	1	I	Selects template register set for processing	TMPSEL=L: (set A) TMPSEL=H: (set B) (note 3)
CPU bus	RD*	1	I	Read signal	
	WR*	1	I	Write signal	
	CS*	1	I	IP90C08 select	
	AD<8..0>	9	I	Address bus	(note 11)
	DB<15..0>	16	I/O	Data bus	
	BUS8	1	I	8-bit bus width	BUS8=H: (DB<15..8> disable) (note 10) BUS8=L:(DB<15..8>enable)
	RST*	1	I	Reset	Requires at least 4 clock cycles. Schmitt trigger input with pull-up resistance.
	BUSY*	1	O	Output busy signal	Operates with ACT* control, indicates internal processing in progress.
Test (note 4)	TEST0,1,2,3	4	O	Test signal	Test pin. Normally High-Z.
	TEST4*,5*	2	I	Test signal	Test pin. Schmitt trigger input with pull-up resistance. Normally always high
Power supply, GND (note 5)	Vdd	12	PW	5V	
	GND	12	PW	Ground	
Total pins		**160**			

Note 1: In LSI internal test mode, TEST4*=L may put the IP90C08 in output mode.

Note 2: When the template select register bit 1 (pr flag) is 0, this pin is enabled. When the pr flag is 1, selection of a template register during execution depends on the value of the template select register bit 0 (ab flag).

Note 3: The LSI test pins (TEST0, TEST1, TEST2, TEST3, TEST4*,and TEST5*) are only for LSI testing before shipment. In normal use, the input should be high and output should be open.

Note 4: To reduce noise, all power supply pins should be connected while the IP90C08 is in use.

Note 5: Pins with pull-up resistance are connected to Vdd through internal high-resistance transistors. Pay careful attention to their electrical characteristics.

Note 6: The RST*, TEST4*, and TEST5* pins have Schmitt trigger input with pull-up resistance.

Note 7: An asterisk (*) following a pin symbol indicates inverse logic.

Note 8: The package is 160-pin QFP (molded body 28-mm square, with 0.65-mm pin pitch).

Note 9: Unused input pins should be constantly set high or low.

Note 10: When BUS8=High to select 8-bit access mode, the upper 8 bits of the data bus (DB<15.0) should be pulled up. If input is allowed to float, there is a chance of excess current flow.

Note 11: When BUS8=Low to select 16-bit access mode, the value of the lowest bit AD0 of the address bus has no significance, but should be constantly set high or low.

4.2 Pin Assignments

Pin No.	Pin Symbol	(I/O)
1	GND	PW
2	IA0	I
3	IA1	"
4	IA2	"
5	IA3	"
6	IA4	"
7	IA5	"
8	IA6	"
9	IA7	"
10	IB0	"
11	IB1	"
12	IB2	"
13	IB3	"
14	IB4	"
15	IB5	"
16	IB6	"
17	IB7	"
18	IC0	"
19	IC1	"
20	GND	PW
21	Vdd	PW
22	IC2	I
23	IC3	"
24	IC4	"
25	IC5	"
26	IC6	"
27	IC7	"
28	ID0	"
29	ID1	"
30	ID2	"
31	ID3	"
32	ID4	"
33	ID5	"
34	ID6	"
35	ID7	"
36	IE0	"
37	IE1	"
38	IE2	"
39	IE3	"
40	Vdd	PW

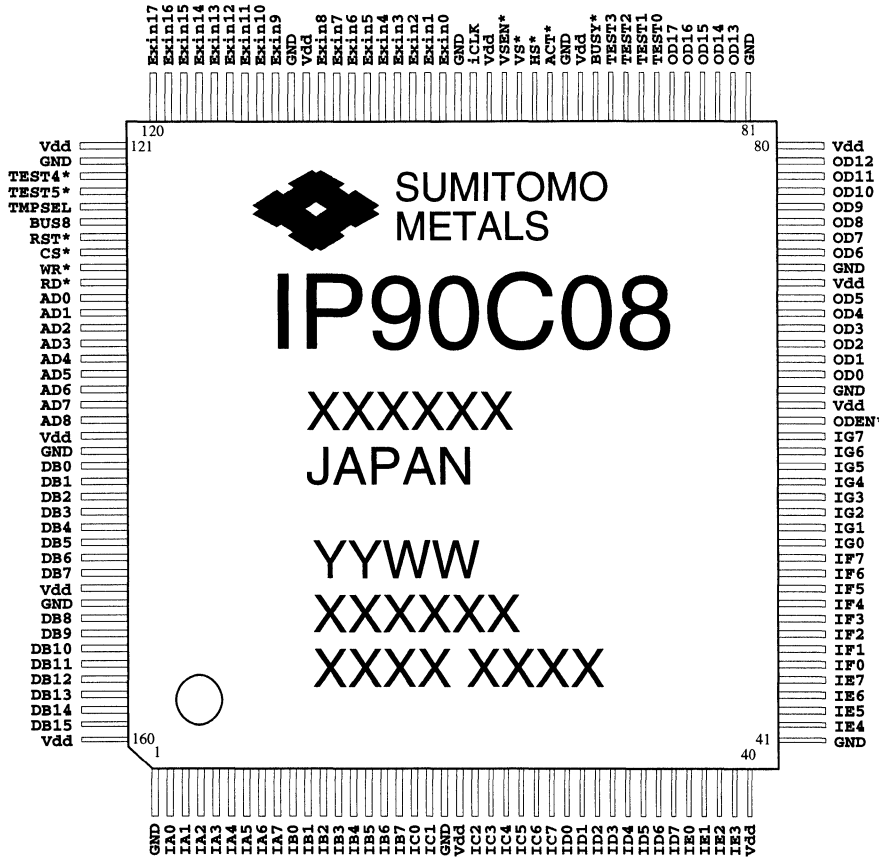
Pin No.	Pin Symbol	(I/O)
41	GND	PW
42	IE4	I
43	IE5	"
44	IE6	"
45	IE7	"
46	IF0	"
47	IF1	"
48	IF2	"
49	IF3	"
50	IF4	"
51	IF5	"
52	IF6	"
53	IF7	"
54	IG0	"
55	IG1	"
56	IG2	"
57	IG3	"
58	IG4	"
59	IG5	"
60	IG6	"
61	IG7	"
62	ODEN*	"
63	Vdd	PW
64	GND	PW
65	OD0	O
66	OD1	"
67	OD2	"
68	OD3	"
69	OD4	"
70	OD5	"
71	Vdd	PW
72	GND	PW
73	OD6	O
74	OD7	"
75	OD8	"
76	OD9	"
77	OD10	"
78	OD11	"
79	OD12	"
80	Vdd	PW

Pin No.	Pin Symbol	(I/O)
81	GND	PW
82	OD13	O
83	OD14	"
84	OD15	"
85	OD16	"
86	OD17	"
87	TEST0	"
88	TEST1	"
89	TEST2	"
90	TEST3	"
91	BUSY*	"
92	Vdd	PW
93	GND	PW
94	ACT*	I
95	HS*	"
96	VS*	"
97	VSEN*	"
98	Vdd	PW
99	iCLK	I
100	GND	PW
101	EXin0	I
102	EXin1	"
103	EXin2	"
104	EXin3	"
105	EXin4	"
106	EXin5	"
107	EXin6	"
108	EXin7	"
109	EXin8	"
110	Vdd	PW
111	GND	PW
112	EXin9	I
113	EXin10	"
114	EXin11	"
115	EXin12	"
116	EXin13	"
117	EXin14	"
118	EXin15	"
119	EXin16	"
120	EXin17	"

Pin No.	Pin Symbol	(I/O)
121	Vdd	PW
122	GND	PW
123	TEST4*	I
124	TEST5*	"
125	TMPSEL	"
126	BUS8	"
127	RST*	"
128	CS*	"
129	WR*	"
130	RD*	"
131	AD0	"
132	AD1	"
133	AD2	"
134	AD3	"
135	AD4	"
136	AD5	"
137	AD6	"
138	AD7	"
139	AD8	"
140	Vdd	PW
141	GND	PW
142	DB0	I/O
143	DB1	"
144	DB2	"
145	DB3	"
146	DB4	"
147	DB5	"
148	DB6	"
149	DB7	"
150	Vdd	PW
151	GND	PW
152	DB8	I/O
153	DB9	"
154	DB10	"
155	DB11	"
156	DB12	"
157	DB13	"
158	DB14	"
159	DB15	"
160	Vdd	PW

4.3 Pin Positions

This illustration shows the IP90C08 with its pin assignments, as viewed from above.



Section 5. Registers

5.1 List of Registers (Address Map)

Group	Address (hex)	Abbreviation	Width (bits)	Function
A Set	00–6F	ATMPD-(0,0)–(15,6)	8	Template data register A
	70–7D	ATMPEN-(0)–(6)	16	Template area enable flag register A
	7E–7F	(reserved)		Not used
	80–87	AMATCH0	16 or 18	Matching coordinate register set A, No. 0
	88–8F	AMATCH1	16 or 18	Matching coordinate register set A, No. 1
	90–97	AMATCH2	16 or 18	Matching coordinate register set A, No. 2
	98–9F	AMATCH3	16 or 18	Matching coordinate register set A, No. 3
	A0–FF	---		Equivalent to address 1A0-1FF
B Set	100–16F	BTMPD-(0,0)–(15,6)	8	Template data register B
	170–17D	BTMPEN-(0)–(6)	16	Template area enable flag register B
	17E–17F	(reserved)		Unused
	180–187	BMATCH0	16 or 18	Matching coordinate register set B, No. 0
	188–18F	BMATCH1	16 or 18	Matching coordinate register set B, No. 1
	190–197	BMATCH2	16 or 18	Matching coordinate register set B, No. 2
	198–19F	BMATCH3	16 or 18	Matching coordinate register set B, No. 3
	1A0–1EB	(reserved & LSI-TEST)		Unused, or LSI internal test register
SYSTEM	1EC–1EE	OutputBias	18	Output compensation bias register
	1EF	OutputShift	5	Output compensation shift register
	1F0	DLY	3	Sum-of-differences delay register
	1F1	SRCNCT	6	Shifter connection register
	1F2–1F3	Hbase	16	Horizontal base coordinate register
	1F4–1F5	Vbase	16	Vertical base coordinate register
	1F6	RST	1	Reset register
	1F8	CLRmatch	2	Matching coordinate clear register
	1FA	IOCNTRL	4	I/O control register
	1FC	TMPSEL	2	Template select register
	1FE	EXEC	2	Execute register

Note 1: The address map assumes byte access mode (8-bit units). In word access mode (16-bit units), the lower 8 bits are allocated to AD0=0 and the upper 8 bits to AD0=1, and address pins AD1 and AD8 are as shown.

Note 2: Access to template register sets A and B (a set consists of a template data register and template area enable flag register) and to matching coordinate register sets A and B is switched at address pin AD8 (AD8=0: set A; AD8=1: set B).

Note 3: AD8 is ignored with respect to addresses in the following registers:

- output bias
- output shift
- DLY
- SRCNCT
- Hbase
- Vbase
- RST
- CLRmatch
- IOCNTRL
- TMPSEL
- EXEC

Thus the values at pins AD0 to AD7 are valid regardless of the value of AD8.

Note 4: After a reset, all registers are set to 0, except for the sum-of-differences registers in the matching coordinate register sets, which are set to 3FFFFh.

Note 5: All registers are read/write enabled, except for the matching coordinate register sets, which are read-only.

5.2 Template Register Sets

A template register set is composed of a 16 x 7-pixel template data register TMPD(h,v) and the seven template area enable registers TMPEN(v). Use the template data register TMPD(h,v) to store pixel data corresponding to coordinates (h,v) within the template image.

The TMPEN(v) registers designate whether the absolute-value-of-difference data for any given pixel in the template register will be included in the template matching calculations. The TMPEN(v) registers consist of 1-bit flags corresponding to each pixel in the template data. Each flag has an enable/disable setting, and each template area enable register TMPEN(v) has a configuration of 16 bits ; TMPEN(v) contains values that enable or disable calculations for every pixel on the line corresponding to vertical coordinate v. After reset, all values in this register are set to 00h.

5.2.1 Template Data Register

The following shows the template data register coordinates.

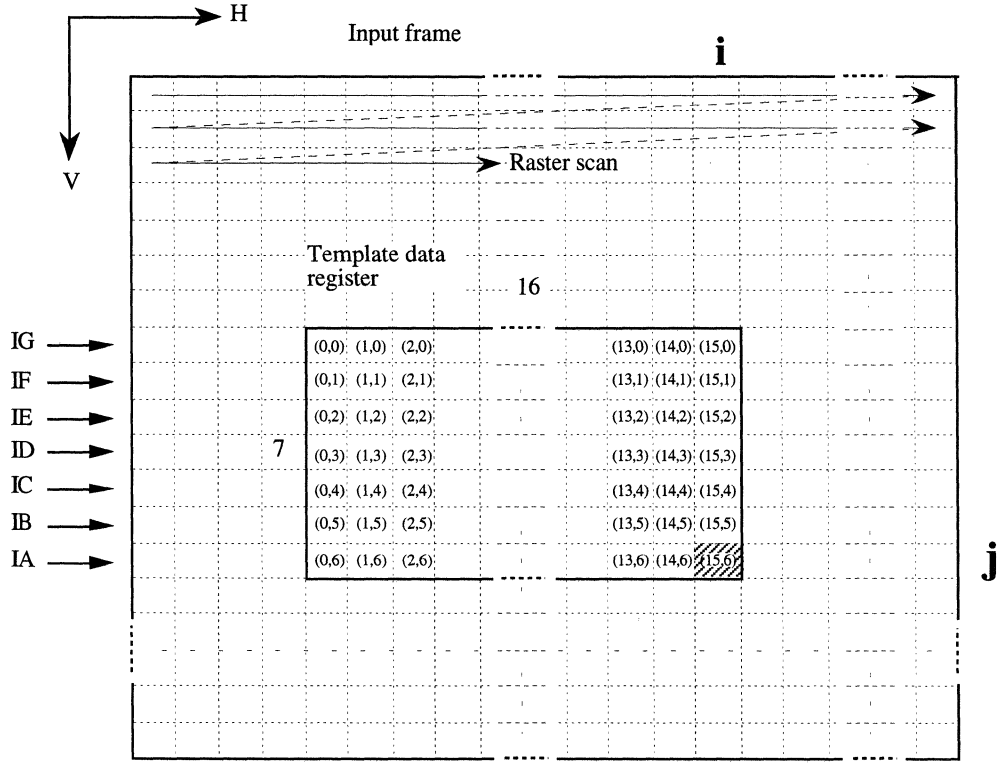
16

	(0,0)	(1,0)	(2,0)	....	(13,0)	(14,0)	(15,0)
	(0,1)	(1,1)	(2,1)	....	(13,1)	(14,1)	(15,1)
	(0,2)	(1,2)	(2,2)	....	(13,2)	(14,2)	(15,2)
7	(0,3)	(1,3)	(2,3)	....	(13,3)	(14,3)	(15,3)
	(0,4)	(1,4)	(2,4)	....	(13,4)	(14,4)	(15,4)
	(0,5)	(1,5)	(2,5)	....	(13,5)	(14,5)	(15,5)
	(0,6)	(1,6)	(2,6)	....	(13,6)	(14,6)	(15,6)

Address Map (hexadecimal notation)

Set A	Set B		
000	100	TMPD(0,0)	Template data register coordinates (0,0)
001	101	TMPD(1,0)	Template data register coordinates (1,0)
002	102	TMPD(2,0)	Template data register coordinates (2,0)
003	103	TMPD(3,0)	Template data register coordinates (3,0)
004	104	TMPD(4,0)	Template data register coordinates (4,0)
005	105	TMPD(5,0)	Template data register coordinates (5,0)
006	106	TMPD(6,0)	Template data register coordinates (6,0)
007	107	TMPD(7,0)	Template data register coordinates (7,0)
:	:	:	:
00F	10F	TMPD(15,0)	Template data register coordinates (15,0)
010	110	TMPD(0,1)	Template data register coordinates (0,1)
011	111	TMPD(1,1)	Template data register coordinates (1,1)
:	:	:	:
01F	11F	TMPD(15,1)	Template data register coordinates (15,1)
020	120	TMPD(0,2)	Template data register coordinates (0,2)
021	121	TMPD(1,2)	Template data register coordinates (1,2)
:	:	:	:
:	:	:	:
:	:	:	:
06E	16E	TMPD(14,6)	Template data register coordinates (14,6)
06F	16F	TMPD(15,6)	Template data register coordinates (15,6)

5.2.1.1 Template Data Register Coordinates and Relation to Input Image Frame



The above illustration shows how template data register coordinates are related to the input image frame. Given the above placement of template data register search positions within the input image frame, the coordinates in the matching coordinate register will be i (the H coordinate) and j (the V coordinate).

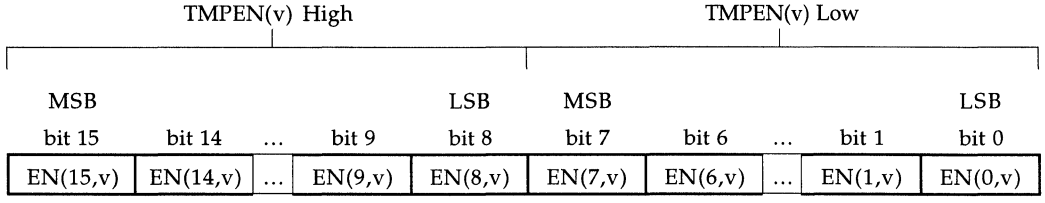
5.2.2 Template Area Enable Registers

A template area enable register TMPEN(v) contains values that enable or disable calculations for every pixel on the line corresponding to vertical coordinate v. Each bit in register TMPEN(v) is an enable/disable setting for calculations involving a pixel (h,v) located on the row with vertical coordinate v, and having a horizontal coordinate h.

After reset, all values in this register are set to 00h.

TMPEN(v)

TMPEN(v) is the template area enable register for row v.



where:

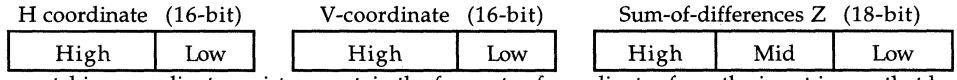
- EN(h,v): Template data enable flag for coordinate (h,v)
- h=0 to 15, v=0 to 6
- EN(h,v)=0: Values calculated with respect to template at coordinates (h,v) are not valid, and will not be included in sum-of-differences calculations.
- EN(h,v)=1: Values calculated with respect to template at coordinates (h,v) are valid, and will be included in sum-of-differences calculations.

Address Map (hexadecimal notation)

Set A	Set B			
070	170	TMPEN(0)	Low	Template data enable flag coordinates (h,0) h=0..7
071	171	TMPEN(0)	High	Template data enable flag coordinates (h,0) h=8..15
072	172	TMPEN(1)	Low	Template data enable flag coordinates (h,1) h=0..7
073	173	TMPEN(1)	High	Template data enable flag coordinates (h,1) h=8..15
074	174	TMPEN(2)	Low	Template data enable flag coordinates (h,2) h=0..7
075	175	TMPEN(2)	High	Template data enable flag coordinates (h,2) h=8..15
076	176	TMPEN(3)	Low	Template data enable flag coordinates (h,3) h=0..7
077	177	TMPEN(3)	High	Template data enable flag coordinates (h,3) h=8..15
078	178	TMPEN(4)	Low	Template data enable flag coordinates (h,4) h=0..7
079	179	TMPEN(4)	High	Template data enable flag coordinates (h,4) h=8..15
07A	17A	TMPEN(5)	Low	Template data enable flag coordinates (h,5) h=0..7
07B	17B	TMPEN(5)	High	Template data enable flag coordinates (h,5) h=8..15
07C	17C	TMPEN(6)	Low	Template data enable flag coordinates (h,6) h=0..7
07D	17D	TMPEN(6)	High	Template data enable flag coordinates (h,6) h=8..15

5.3 Matching Coordinate Register Sets

The matching coordinate registers are used to detect and store 16-bit coordinates and 18-bit sum-of-difference values.



The matching coordinate registers contain the four sets of coordinates from the input image that have the lowest sums of differences relative to the template data. Assuming that MINPOS(n) represents the coordinates from the input image that have the n^{th} lowest sum of differences, then four sets of coordinates MINPOS(0) to MINPOS(3) are calculated and stored in the register. The coordinates with the lowest sum of differences are recorded as MINPOS(0): these are horizontal coordinate H0, vertical coordinate V0, and sum-of-differences value Z0.

The matching coordinate registers are read-only. After a reset, all valid Z0-Z3 fields (which store 18-bit sum-of-products values) are set to 3FFFFh, and all bits in the H(n) and V(n) fields are set to 0000h.

Address Map (hexadecimal notation)

Set A	Set B				
080	180	H0	Low	Coordinate H0	MINPOS(0)
081	181	H0	High		
082	182	V0	Low	Coordinate V0	
083	183	V0	High		
084	184	Z0	Low	Sum-of-differences Z0	
085	185	Z0	Mid		
086	186	Z0	High		
087	187	–	–	(reserved address)	
088	188	H1	Low	Coordinate H1	MINPOS(1)
089	189	H1	High		
08A	18A	V1	Low	Coordinate V1	
08B	18B	V1	High		
08C	18C	Z1	Low	Sum-of-differences Z1	
08D	18D	Z1	Mid		
08E	18E	Z1	High		
08F	18F	–	–	(reserved address)	
090	190	H2	Low	Coordinate H2	MINPOS(2)
091	191	H2	High		
092	192	V2	Low	Coordinate V2	
093	193	V2	High		
094	194	Z2	Low	Sum-of-differences Z2	
095	195	Z2	Mid		
096	196	Z2	High		
097	197	–	–	(reserved address)	
098	198	H3	Low	Coordinate H3	MINPOS(3)
09B	19B	H3	High		
09A	19A	V3	Low	Coordinate V3	
09B	19B	V3	High		
09C	19C	Z3	Low	Sum-of-differences Z3	
09D	19D	Z3	Mid		
09E	19E	Z3	High		
09F	19F	–	–	(reserved address)	

5.4 Output Compensation Bias Register

A sum-of-differences output value can have up to 18 digits, as calculated between all respective pixels of the input image (IA–IG) and the selected template register (A or B), and including data from any expanded input areas (EXin). To enable accurate matched filter output, the IP90C08 provides 18-bit output capability for all results.

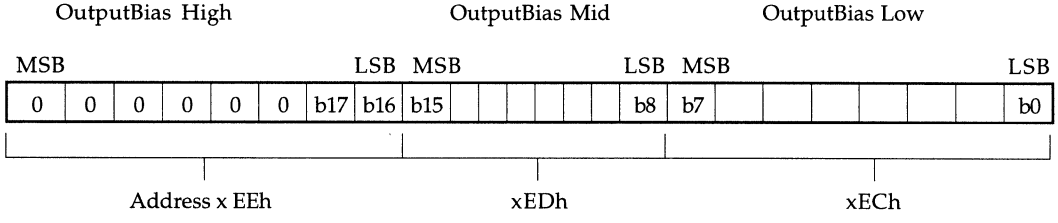
Sum-of-differences (matched filter) output values can be augmented by further addition or shift calculations. Any values to be added in the form of output compensation should be stored in the output compensation bias register. Shift values should be stored in the output compensation shift register described on the next page.

Output compensation values are added by an 18-bit full-scale adder. This minimizes the margin of error introduced by any subsequent shift processes. If no expanded-area input is used, or if the highest sum-of-differences results have 17 or fewer bits, the result can be considered a positive value and written in two's-complement notation with the highest bit (b17) used as a code bit. Be aware, however, that the output compensation shifter does not operate in two's-complement mode.

After reset, all values in this register are set to 00h.

OutputBias

The output compensation bias register has a width of 18 bits.



Address Map (hexadecimal notation)

0EC	1EC	OutputBias Low
0ED	1ED	OutputBias Mid
0EE	1EE	OutputBias High

Values are mapped for addresses 0ECh, 0EDh, and 0EEh, and for 1ECh, 1EDh, and 1EEh.

5.5 Output Compensation Shift Register

As mentioned earlier, output values can be augmented by applying further addition and shift operations to the sum-of-differences (matched filter) output (including expanded-area input, if any). This shift value is stored in the output compensation shift register.

A margin of error is smaller when shifting after addition of output compensation bias register than when adding output compensation bias register after shifting. The output compensation shifter applies a right-shift operation (division by 2 to the n^{th} power) to an 18-bit absolute-value integer in binary form. This shift places one or more zeros in the upper bits.

After reset, all values in this register are set to 00h. When no shift will be applied, a value of 00h should be entered in this register (representing a shift of 0 bits).

Output Shift

The diagram below illustrates the output compensation shift register.

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0					

The register address is either 0EFh or 1EFh. Values are mapped for either address.

5.6 Sum-of-Differences Delay Register

Before the results of the IP90C08's internal sum-of-differences calculations are added to expanded-area input (EXin) from other devices, a shift register is placed at the internal sum-of-differences calculator (see Section 3.1, "Internal Block Diagram"). This variable-length shift register acts as a delay circuit to adjust for phase differences between external input and internal calculations. The sum-of-differences delay register has a 4-level configuration, as shown below, and can produce a delay having 0 to 4 steps.

The register has an effective length of 3 bits, and the upper five bits should have the value 0. The value 04h causes the maximum delay of 4 clock cycles. After reset, all values in the register are set to 00h.

DLY

The diagram below illustrates the sum-of-products delay register.

MSB LSB

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0			

The register address is either 0F0h or 1F0h. Values are mapped for either address.

5.7 Shifter Connection Register

The one-column, 8-bit data entering from image data input pins IA-IG is latched at the rise of the clock signal and stored in internal image area shift register arrays. Each of these arrays has a 16-step x 7-row x 8-bit configuration. The IP90C08 can change the template matching area by changing how these image area shift registers are connected.

Each image area shift register array has a flag that designates whether its input end is to be connected to an external signal coming through an image data input pin (Srxy flag=0), or to the output end of the last step of the preceding shift register (Srxy flag=1). For example, the 'srab' flag controls switching between output from the last step of the shift register from the IA pin (srab=1) and input from the IB pin (srab=0).

After reset, all values in this register are set to 00h, and image areas are set to 16 x 7 pixels.

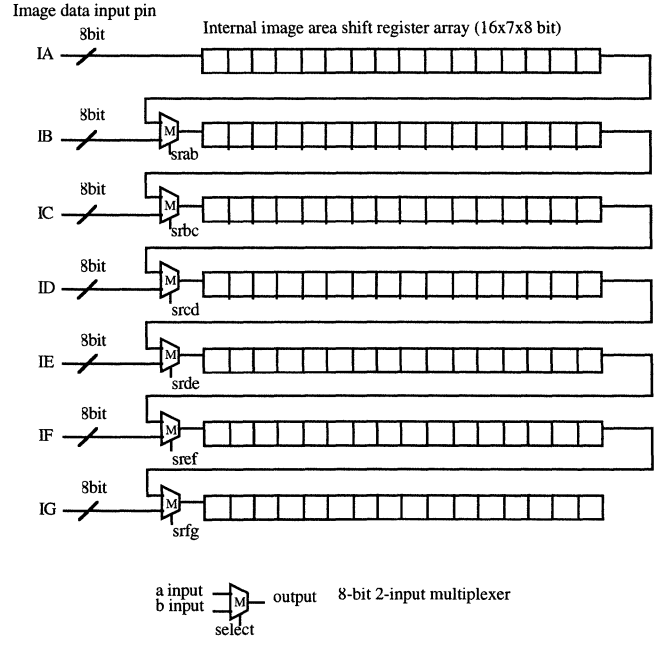
SRCNCT

Shifter connection register

MSB LSB

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	srfg	sref	srde	srcd	srbc	srab

The register address is either 0F1h or 1F1h. Values are mapped for either address.



Note 1: The above block diagram does not include all the IP90C08's functions. Refer to related sections of this manual for information on functions, timing, etc.

5.8 Base Coordinate Registers

The base coordinate registers contain the base coordinate values, to which the coordinate counter values are added. Conversely, coordinate counters begin counting from coordinate values loaded from these registers. The base coordinate registers consist of the horizontal base coordinate register Hbase (which contains the horizontal coordinate compensation value) and the vertical base coordinate register Vbase (which contains the vertical coordinate compensation value).

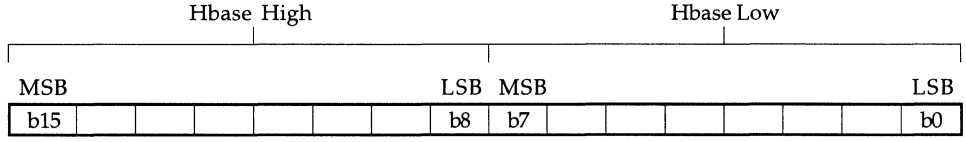
The horizontal coordinate counter (Hcnt) begins by loading the initial value from the horizontal base coordinate register Hbase when HS* is input, and then begins counting cycles of iCLK. The vertical coordinate counter (Vcnt) begins by loading the initial value from the vertical base coordinate register Vbase when VS* is input, and then begins counting upwards at each input of HS*. Hcnt and Vcnt are 16-bit up-count registers, and follow the maximum value of FFFFh by returning to 0000h. This configuration is best used by expressing coordinates in two's-complement format and placing the origin (0,0) at the center of the frame, then expressing the coordinates of template matching positions as complements oriented towards the center, lower left, etc. of the image area.

For example, if the center of a 640 x 480-pixel frame is the origin (0,0), the initial values of the counters can be assigned to Hbase=–320, Vbase=–240, and expressed in two's-complement notation. In this example, the measurement of the matching coordinates with respect to the center of the image area is Hbase "-8" and Vbase "-3."

After reset, all values in this register are set to 00h.

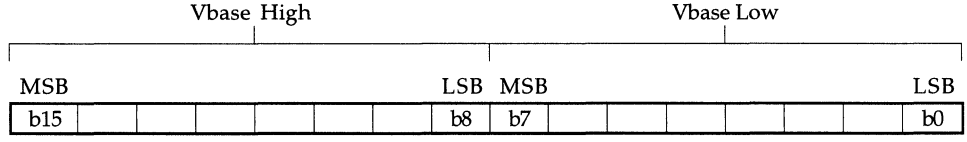
Hbase

The horizontal coordinate base register has a width of 16 bits.



Vbase

The vertical coordinate base register has a width of 16 bits.



Address Map (hexadecimal notation)

0F2	1F2	Hbase Low
0F3	1F3	Hbase High
0F4	1F4	Vbase Low
0F5	1F5	Vbase High

Values are mapped both for address 0F2h, 0F3h, 0F4h, and 0F5h, and for 1F2h, 1F3h, 1F4h, and 1F5h.

5.9 Reset Register

This register performs software resets by executing the logical sum of the hardware reset operations initiated from the reset signal pin (RST*). All internal registers are reset. The matching coordinate sum-of-differences registers in the matching coordinate register sets, in which all bits are set to 1, and all other registers are cleared to 0. The value 1 is written to this register to reset all registers, after which this bit is cleared to 0 when the software reset is completed. Software resets are completed within three cycles of iCLK.

After a hardware reset from the reset signal pin (RST*), this register is set to 00h.

RST

The diagram below illustrates the reset register.

MSB LSB

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	0	0	r

where r is the software reset flag:

r=1: reset

r=0: cancel reset

The register address is either 0F6h or 1F6h. Values are mapped for either address.

5.10 Matching Coordinate Clear Register

The matching coordinate clear register initiates clearing of registers in the matching coordinate register sets. This differs from software resets from the reset flag in the reset register, or hardware resets from the reset pin (RST*), in that the template registers and other registers are not affected.

After reset, all values in this register are set to 00h.

CLRmatch

The diagram below illustrates the matching coordinate clear register.

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	0	clb	cla

The cla and clb flags clear matching coordinate register set A (H coordinate register, V coordinate register, sum-of-differences register) and B, respectively. Clearing requires one clock cycle after writing to the flag. This differs from the software reset flag in that template register sets and other settings are not affected.

cla/clb=1: clear matching register set A/B

cla/clb=0: cancel clearing of matching register set A/B

The register address is either 0F8h or 1F8h. Values are mapped for either address.

5.11 Input/Output (I/O) Control Register

The I/O control register controls the flow of sum-of-differences output from output pin OD<17..0>, as well as data input from image data input pins IA–IG, and expanded sum-of-differences data input EXin<17..0>).

After reset, all values in this register are set to 00h.

IOCTRL

The diagram below illustrates the input/output control register.

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	eid	id	od	se

se: the output compensation circuit enable flag. This determines whether sum-of-differences output is sent directly from the sum-of-differences circuit or through the output compensation circuits (see Section 3.1, "Internal Block Diagram").

- se=1: enable output compensation circuits (enables expanded input, delay register, and output barrel shifter)
- se=0: disable output compensation circuits

od: the sum-of-differences output enable flag.

- od=1: output sum-of-differences output
- od=0: sum-of-differences output pin OD<17..0> signal set high

id: the input image data reset flag.

- id=1: set image input data signal to 0, regardless of pins IA-IG (note 1)
- id=0: process image input data from pins IA-IG

eid: the expanded input image data reset flag.

- eid=1: process expanded sum-of-differences input data from expanded sum-of-differences input pins EXin<17..0> (Note 1)
- eid=0: set expanded sum-of-differences input data signal to 0, regardless of expanded sum-of-differences input pins EXin<17..0>

Note 1: This option represents a power-saving mode, reducing chip power consumption.

The register address is either 0FAh or 1FAh. Values are mapped for either address.

5.12 Template Select Register

The template select register selects one of the template register sets (A or B) to be enabled for template matching processing.

When a template register is accessed directly from the CPU bus, it is mapped with a difference address, so that reading and writing are both enabled regardless of the settings in the template select register.

After reset, all values in this register are set to 00h.

TMPSEL

The diagram below illustrates the template select register.

MSB LSB

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	0	pr	ab

ab: This flag selects a template register set for template matching. This flag is enabled when pr=1. (When pr-0, the TMPSEL pin is enabled.)

- ab=0: enables template register set A
- ab=1: enables template register set B

pr: This flag determines the method of selecting the template register set for processing.

- pr=0: template register set is selected by the TMPSEL pin
- pr=1: template register set is selected by ab

The register address is either 0FCh or 1FCh. Values are mapped for either address.

5.13 Execute register

The execute register controls the execution of matching coordinate measurements in the template matching process. The register creates a logical product with ACT*. Although this register controls matching coordinate measurement, sum-of-differences calculation is performed constantly. To conserve power, use the input data rest flag id in the I/O control register (IOCTRL) to reset input data and enter power-saving mode.

When 1 is written to the execute flag (ex), ex enables the internal chip functions, which then become effective immediately after the input of the first valid VS* following a VSEN*. This initiates matching coordinate measurement.

In actual use, 1 is written to ex by a software procedure, after which VS* is latched low upon the rise of the clock signal, and an image field is thereby opened (by loading the vertical coordinate base register value as the initial value for the vertical coordinate counter Vcnt). Matching coordinate measurement is then performed on an area of the raster scan input, defined as an image area by the pixels that are input during the low phase of ACT* and that determine the lower right corner of the image area.

If 0 is written to ex, ex takes effect internally within two clock cycles, and matching coordinate measurement processing is terminated.

The field control flag (fl) is effective for the duration of one field only, specifically from the opening of a field by VS* until the next VS*, and executes matching coordinate measurement for that frame only. If fl is set to 1 when 1 is written to ex, matching coordinate measurement begins normally with the next VS*, but ex is cleared when the second VS* arrives. This feature enables a single field to be measured without needing an external means of holding ACT* high.

After reset, all values in this register are set to 00h.

EXEC

The diagram below illustrates the execute register.

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	0	fl	ex

ex: the start-execution flag.

- ex=1: start template matching
- ex=0: stop template matching

fl: the field-control flag.

- fl=1: enable field control circuit
- fl=0: disable field control circuit

The register address is either 0FEh or 1FEh. Values are mapped for either address.

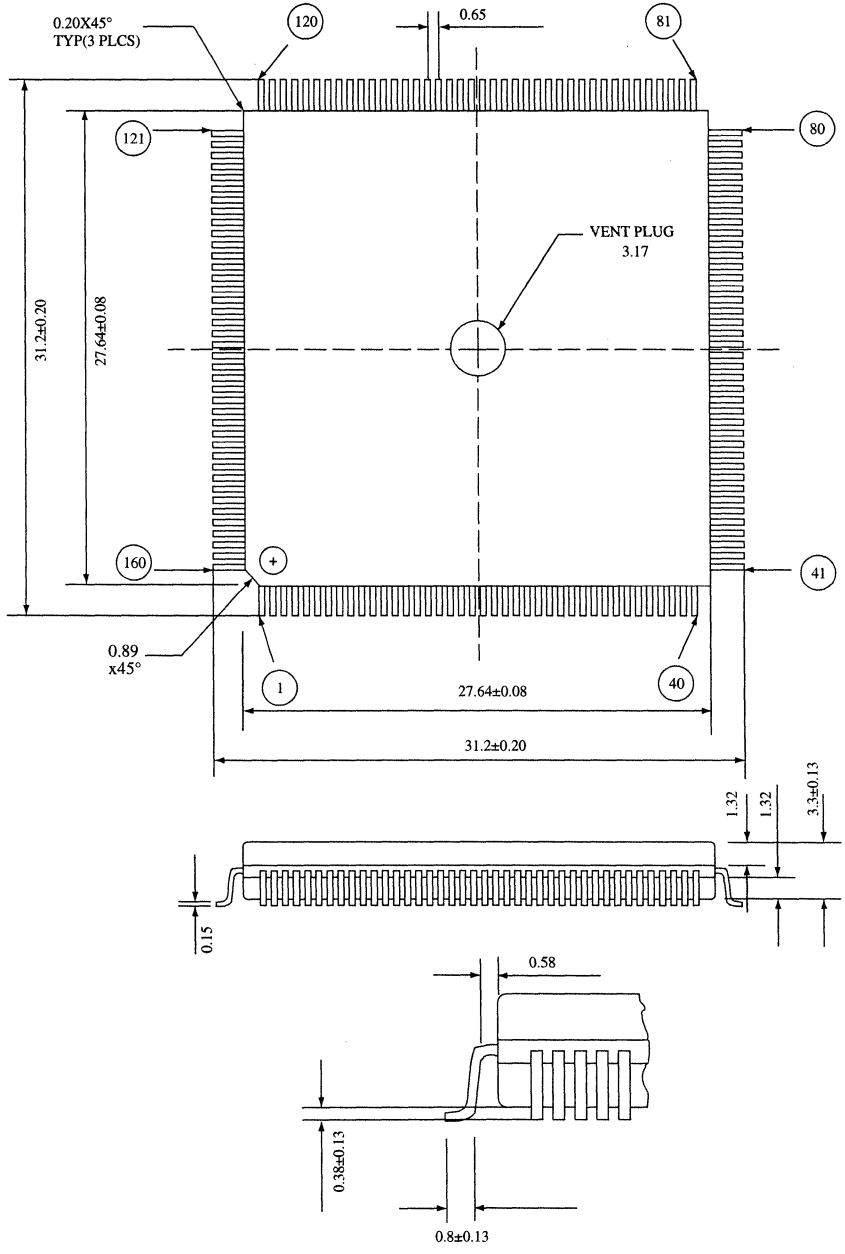
5.14 LSI Internal Test Register

This register is used for testing before shipment, and is not for other use. 0 should be written to all bits in the register, or the IP90C08 should be reset and used with all bits reading 00h. If any bit contains 1, the chip goes into shipment test mode, and normal operation cannot be assured.

The register address is 1A0h–1EBh.

Section 6. External Dimensions

6.1 Dimensions (160-Pin QFP)



Section 7. Electrical Characteristics

7.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Power supply voltage	Vdd	-0.3 to 6.5	V
Input voltage	Vi	-0.3 to Vdd+0.3	V
Input current	Ii	±10	mA
Output current	Io	10	mA
Operating temperature	Topt	0 to 70	°C
Storage temperature	Tstg	-10 to 80	°C

7.2 Recommended Operating Conditions (GND=0V, Ta=0° to 70°C)

Item	Symbol	Conditions	Min	Typ	Max	Unit
Power supply voltage	Vdd		4.75	5.0	5.25	V
High-level input voltage	Vih	TTL level	2.0		Vdd	V
Low-level input voltage	Vil	normal input	0		0.8	V
High-level input voltage	Vih	SCHMITT	2.25		Vdd	V
Low-level input voltage	Vil	input (note 1)	0		0.8	V
Input rise time	Tri	TTL level	0		100	ns
Input fall time	Tfi	normal input	0		100	ns
Input rise time	Trisch	SCHMITT	0		1000	ns
Input fall time	Tfisch	input (note 1)	0		1000	ns

Note 1: The three Schmitt trigger input pins are RST*, TEST4*, and TEST5*.

7.3 Input/Output Pin Capacity (Vdd=Vi=0V)

Item	Symbol	Conditions	Min	Typ	Max	Unit
Input pins	Cin	f=1 MHz		10		pF
Output pins	Cout	f=1 MHz		10		pF
I/O pins	Cin	f=1 MHz		10		pF

7.4 DC Characteristics (Vdd=5V±5%, GND=0V, Ta=0° to 70°C)

Item		Symbol	Conditions	Min	Typ	Max	Unit
Static current consumption (note 1)		Il	Vi=VDD or GND			200	μA
Output short current (note 2)		Ios	VDD=Max, Vo=VDD	15	50	130	mA
(all output and I/O pins)			VDD=Max, Vo=0	-5	-25	-100	
Low-level input leak current							
	Normal I/O pins	Iil	Vi=GND	-10	±1	10	μA
	Pins with pull-up resistance (note 3)	Iipl	Vi=GND	-35	-115	-350	μA
High-level input leak current (all input and I/O pins)		Iih	Vi=VDD	-10	±1	10	μA
Low-level output voltage		Vol	Iol=4 mA		0.2	0.4	V
High-level output voltage		Voh	Ioh=-0.5 mA	4.5		VDD	V
(note 4)		Voh	Iol=-4 mA	2.4		VDD	V
SCHMITT hysteresis voltage		Vsch	Vil to Vih	0.4	0.8		V

Note 1: Excluding static current consumption to pull-up resistors.

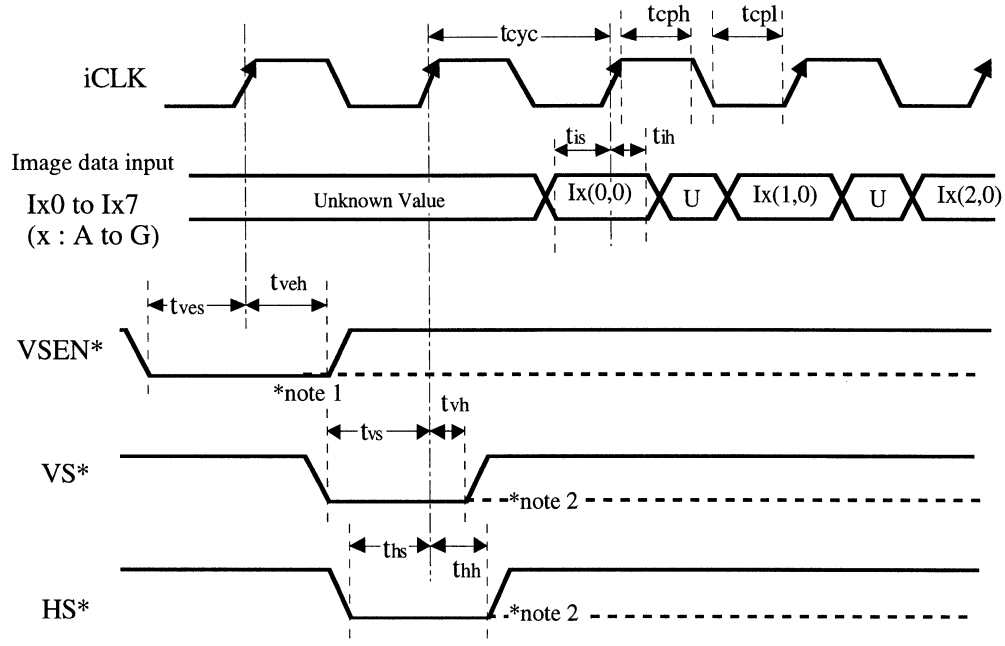
Note 2: Output short current for one second or less, at one LSI pin.

Note 3: The three pins pull-up resistors are RST*, TEST4*, and TEST5*.

Note 4: Output is CMOS level (TTL level).

7.5 AC Characteristics (Pin load capacitance 30pf)

a) Data Timing at Frame Start



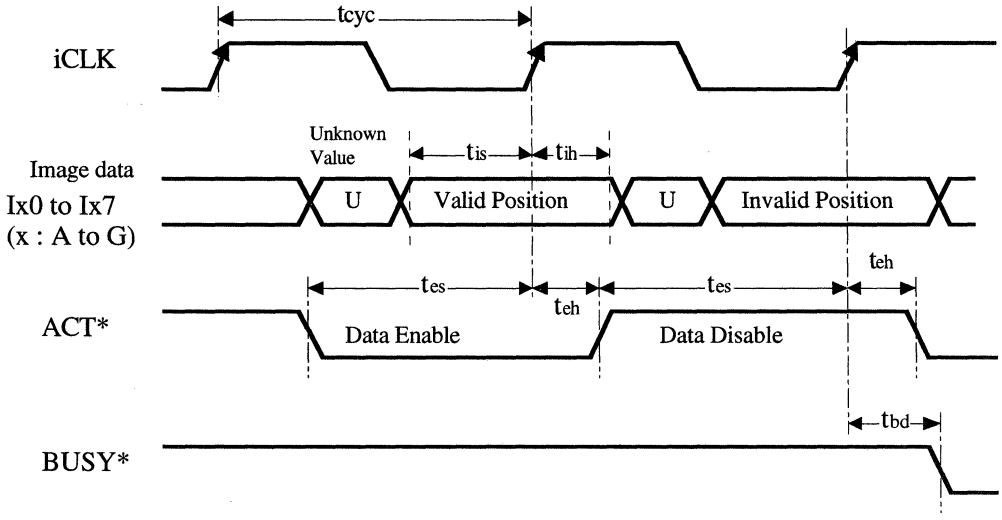
Note 1: VS* is enabled when VSEN* of the previous clock cycle is low.

Note 2: VS* and HS* need not be pulses, but can extend as shown by the dotted lines. However, a high-level input signal is required at least two clock cycles before the next sync signal is input.

Unit: ns

Item	Symbol	Min	Typ	Max
iCLK cycle period	tcyc	33.0		
iCLK high level period	tcph	13.0		
iCLK low level period	tcpl	13.0		
Image data IA-IG setup time	tis	8.0		
Image data IA-IG hold time	tih	3.0		
VSEN* signal setup time	tves	8.0		
VSEN* signal hold time	tveh	3.0		
VS* signal setup time	tvs	8.0		
VS* signal hold time	tvh	3.0		
HS* signal setup time	ths	8.0		
HS* signal hold time	thh	3.0		

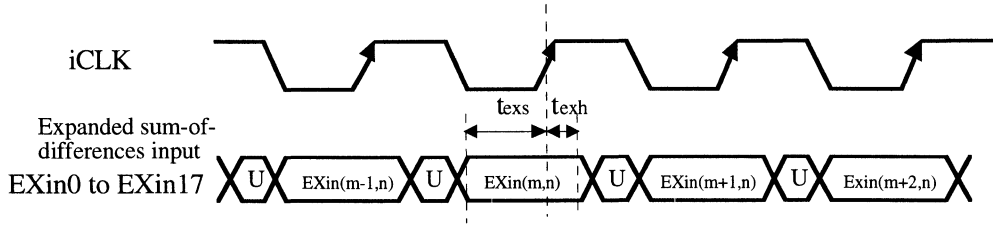
b) Image Data Input Control Timing



Unit: ns

Item	Symbol	Min	Typ	Max
iCLK cycle length	tcyc	33.0		
Image data IA-IG setup time	tis	8.0		
Image data IA-IG hold time	tih	3.0		
ACT* signal setup time	tes	8.0		
ACT* signal hold time	teh	3.0		
BUSY* delay time	tbd	2.0		20.0

c) Expanded Sum-of-Differences Input Timing

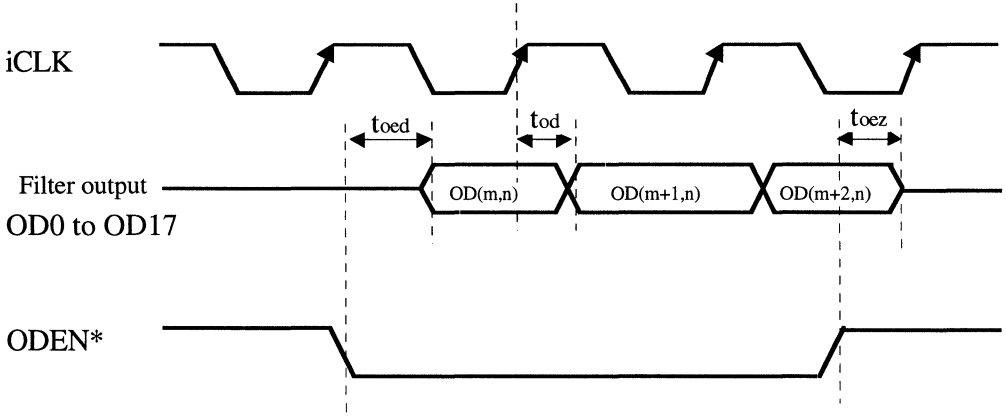


Unit: ns

Item	Symbol	Min	Typ	Max
EXin0 to EXin17 setup time	texs	8.0		
EXin0 to EXin17 hold time	texh	3.0		

d) Matched Filter Output Control Timing

Matched Filter Output Enable

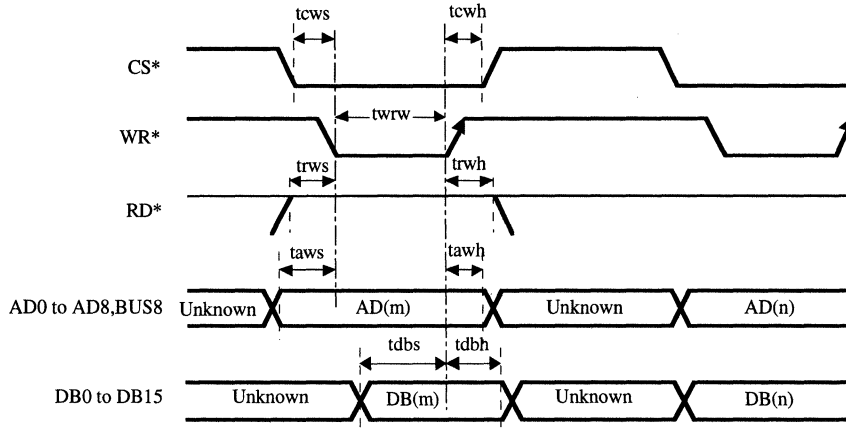


Unit: ns

Item	Symbol	Min	Typ	Max
Delay time from ODEN* signal fall to filter output OD enable	toed			15.0
Delay time from ODEN* signal fall to filter output OD disable	toez			15.0
Filter output OD delay time	tod	2.0		20.0

e) CPU Interface Timing

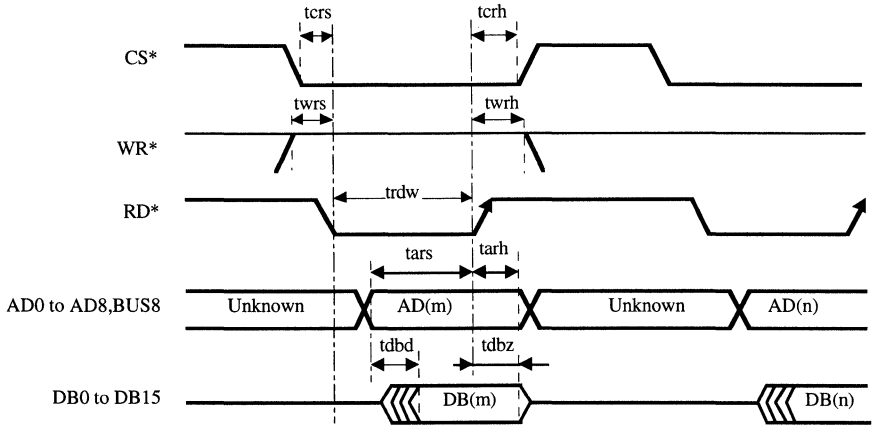
Write Cycle



Unit: ns

Item	Symbol	Min	Typ	Max
CS* signal setup time (from WR* fall)	tcws	3.0		
CS* signal hold time (from WR* rise)	tcwh	3.0		
RD* signal setup time (from WR* fall)	trws	3.0		
RD* signal hold time (from WR* rise)	trwh	3.0		
WR* low pulse width	twrw	20.0		
AD0 to AD8, BUS8 setup time (from WR* fall)	taws	8.0		
AD0 to AD8, BUS8 hold time (from WR* rise)	tawh	3.0		
DB0 to DB15 setup time (from WR* fall)	tdbs	15.0		
DB0 to DB15 hold time (from WR* fall)	tdbh	3.0		

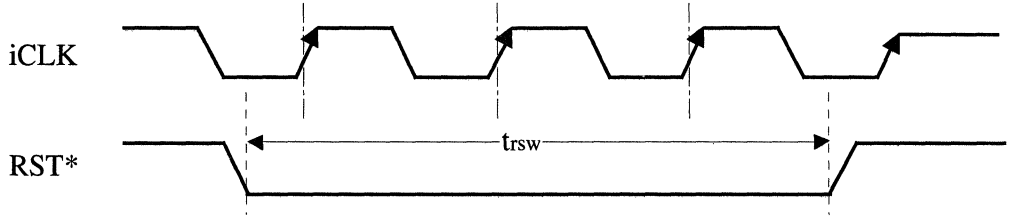
Read Cycle



Unit: ns

Item	Symbol	Min	Typ	Max
CS* signal setup time (from RD* fall)	tcrs	3.0		
CS* signal hold time (from RD* rise)	tcrh	3.0		
WR* signal setup time (from RD* fall)	twrs	3.0		
WR* signal hold time (from RD* rise)	twrh	3.0		
RD* low pulse width	trdw	20.0		
AD0 to AD8, BUS8 setup time (from RD* fall)	tars	15.0		
AD0 to AD8, BUS8 hold time (from RD* rise)	tarh	3.0		
Delay time until DB0-DB15 disable (from AD0 to AD8 verification)	tdbd			30.0
Delay time until DB0-DB15 disable (from RD* rise)	tdbz			15.0

f) Reset Timing

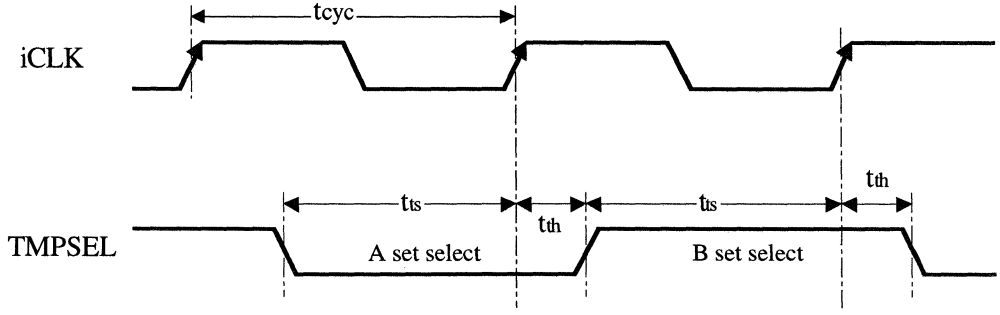


The minimum reset signal input period should be three clock cycles. An internal reset is executed when a low RST* is detected on two consecutive clock rises. RST* rise is detected by clock rise, reset will be released after three clocks.

Unit: clock cycles

Item	Symbol	Min	Typ	Max
Low pulse duration of RST*	trsw	3.0		

g) Template Register Switching Control



Unit: ns

Item	Symbol	Min	Typ	Max
iCLK cycle	tcyc	33.0		
TMPSEL setup time	tts	8.0		
TMPSEL hold time	tth	3.0		

Section 8. Sample Applications and Reference Information

8.1 16 x 7 Averaging Filter Processing

As described in Section 1.2, "Algorithms," template matching operates on image areas of 16 x 7 pixels by calculating the differences between the template and pixel data from the 16 x 7-pixel area, then summing these differences. One application of this process is the most general type of spatial filter, known as an averaging filter. This filter calculates the average value within the image area and outputs that value as the median pixel value. This is used in noise reduction and similar applications.

The IP90C08 uses a template in which all data values are 00h, sets the template area enable register (TMPEN) for any desired image area, and selects an output value range using the output shift register and output compensation bias register.

For example, for averaging filter processing of a 7 x 7-pixel area, all pixels in the template register set should remain at 00h, and template enable register values should all be E000h.

Because the maximum value will be the sum of differences of 7 x 7 = 49 pixels, or

$$(2^8-1) \times 49 = 12{,}495 < 2^{13},$$

a 5-bit shift is needed to express the output as an 8-bit value. The output shift register should therefore be set to 5.

Note: This example is intended only as a conceptual sketch of the use of the IP90C08, and requires further refinement before it can actually be used.

Template Data in a 7 x 7-Pixel Averaging Filter

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	0	0									
1	0	0	0	0	0	0	0									
2	0	0	0	0	0	0	0									
3	0	0	0	0	0	0	0									
4	0	0	0	0	0	0	0									
5	0	0	0	0	0	0	0									
6	0	0	0	0	0	0	0									

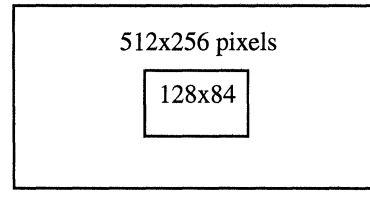
Valid Not valid

8.2 Sample System Configuration for Measurement of Movement Vectors in a Grayscale Image Area

Note: This example is intended only as a conceptual sketch of the use of the IP90C08, and requires further refinement before it can actually be used.

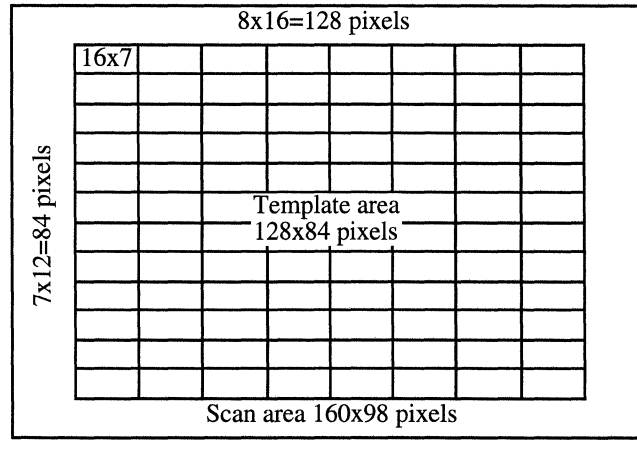
8.2.1 Input Image Area

The total input image area is 512 x 256 pixels, of which movement vectors within an area of 128 x 84 pixels are to be identified in units of 16 x 7-pixel image areas.

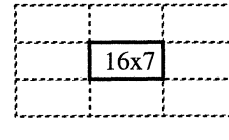


8.2.2 Movement Vector Measurement Area

Movement vectors are to be measured in 8 x 14 = 112 areas.



8.2.3 Range of Movement Vector Identification in Each Image Area



The scan area is 48 x 21 pixels.

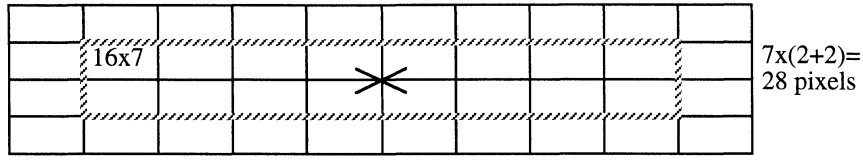
Each 16 x 7 image area is taken as the center of a 48 x 21-pixel area, and scanned after movement has taken place. The coordinates with the best closeness-of-fit on a pixel-to-pixel basis are identified, and the distance of those coordinates from the base point is taken as the movement vector.

8.2.4 Image Data Scanned and Transferred Using 16 IP90C08 Chips

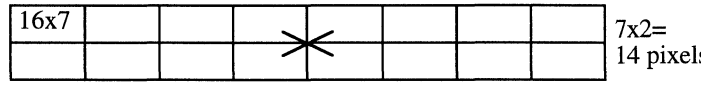
Input image data (image after movement).

Data transferred at one scan: 160 x 28 = 4480 pixels.

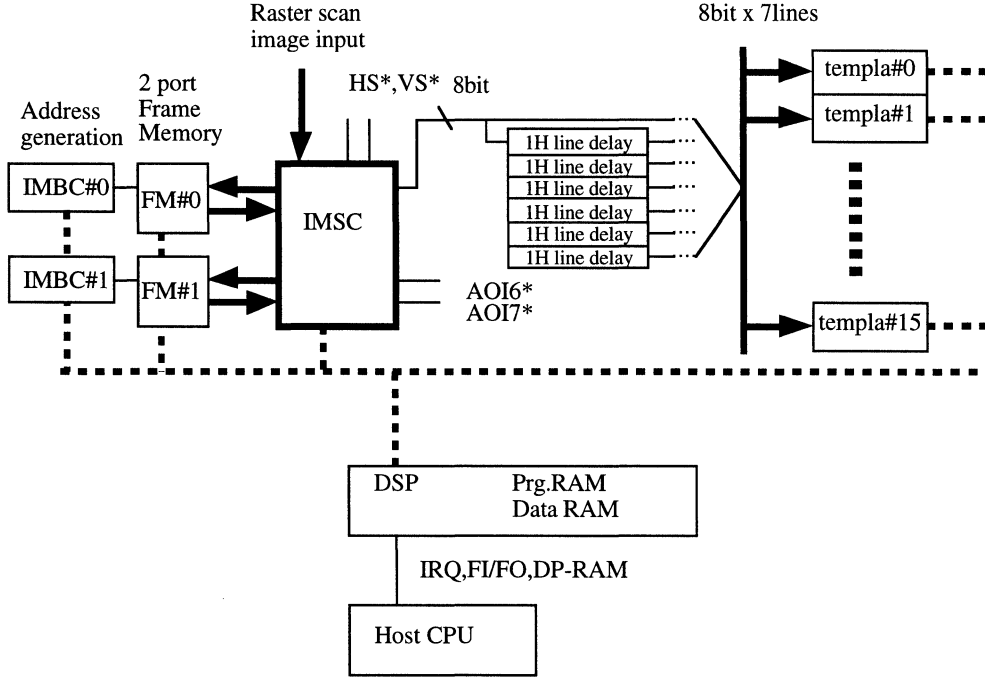
16 x (8+2) = 160 pixels



Template image (base image) 16 x 8 = 128 pixels



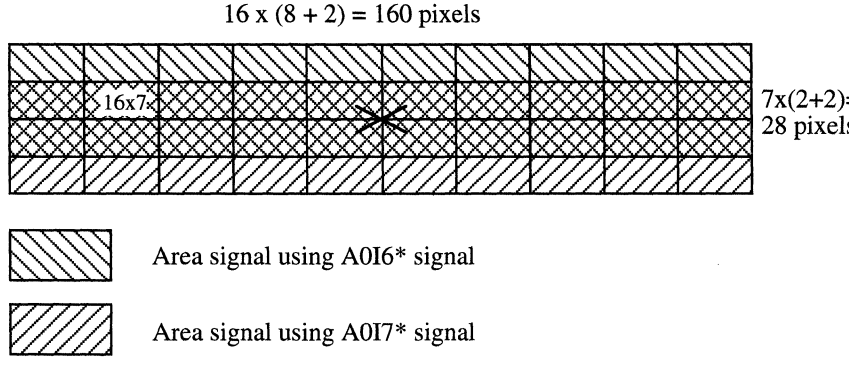
8.2.5 System Configuration for Processing of a Continuous Area



8.2.6 Area Signal Generation in Each templa Chip

Because the row setting for template matching must be moved with every scan, and the column coordinates for the start of the template matching area must be able to be changed as needed, an IMSC or similar system with AOI function for programmable coordinate selection is advised. Because the column setting for the area is fixed in terms of widths from the starting point, it is possible to make this setting by decoding counter values.

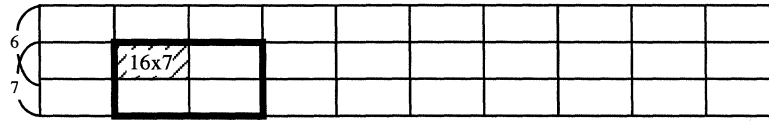
1) Example of Horizontal Area Selection and Vertical Starting Coordinate (Image Area Coordinate) Selection



2) Column Area Designation

Using an 8-bit binary counter, select the column setting for the image area by decoding the counter value.

3) Example of Active Area Signal (ACT*) Setting



When looking for movement in each of the image areas in 8 directions surrounding the shaded area, assert ACT* low for the area enclosed in the heavy line.

8.3 Template Data Notation Format

The following charts show the format used for template notation. Refer to them when preparing template data manually.

Address

Upper 5 bits Lower 4 bits

Aset		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Template data	00	000	001	002	003	004	005	006	007	008	009	00A	00B	00C	00D	00E	00F
	01	010	011	012	013	014	015	016	017	018	019	01A	01B	01C	01D	01E	01F
	02	020	021	022	023	024	025	026	027	028	029	02A	02B	02C	02D	02E	02F
	03	030	031	032	033	034	035	036	037	038	039	03A	03B	03C	03D	03E	03F
	04	040	041	042	043	044	045	046	047	048	049	04A	04B	04C	04D	04E	04F
	05	050	051	052	053	054	055	056	057	058	059	05A	05B	05C	05D	05E	05F
	06	060	061	062	063	064	065	066	067	068	069	06A	06B	06C	06D	06E	06F
/	07	070	071	072	073	074	075	076	077	078	079	07A	07B	07C	07D		

Valid

Address

Upper 5 bits Lower 4 bits

Bset		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Template data	10	100	101	102	103	104	105	106	107	108	109	10A	10B	10C	10D	10E	10F
	11	110	111	112	113	114	115	116	117	118	119	11A	11B	11C	11D	11E	11F
	12	120	121	122	123	124	125	126	127	128	129	12A	12B	12C	12D	12E	12F
	13	130	131	132	133	134	135	136	137	138	139	13A	13B	13C	13D	13E	13F
	14	140	141	142	143	144	145	146	147	148	149	14A	14B	14C	14D	14E	14F
	15	150	151	152	153	154	155	156	157	158	159	15A	15B	15C	15D	15E	15F
	16	160	161	162	163	164	165	166	167	168	169	16A	16B	16C	16D	16E	16F
/	17	170	171	172	173	174	175	176	177	178	179	17A	17B	17C	17D		

Valid

When writing in 16-bit bus width, the lower values at the LSB end are placed in the lower addresses, and the higher values at the MSB end are placed in the higher addresses.

SMI ASSP

Image Processing LSI Series

IP90C10

Labeling Accelerator (LABop)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.4

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Overview

1.1 Product Overview

The IP90C10 (LABop) is a single-chip, image-link component labeling processor that uses a proprietary algorithm and high-speed circuit technology. It can process up to 4,094 temporary labels using either a 4-connectedness or an 8-connectedness labeling operation. Labeling can be processed in real time using three chips in parallel.

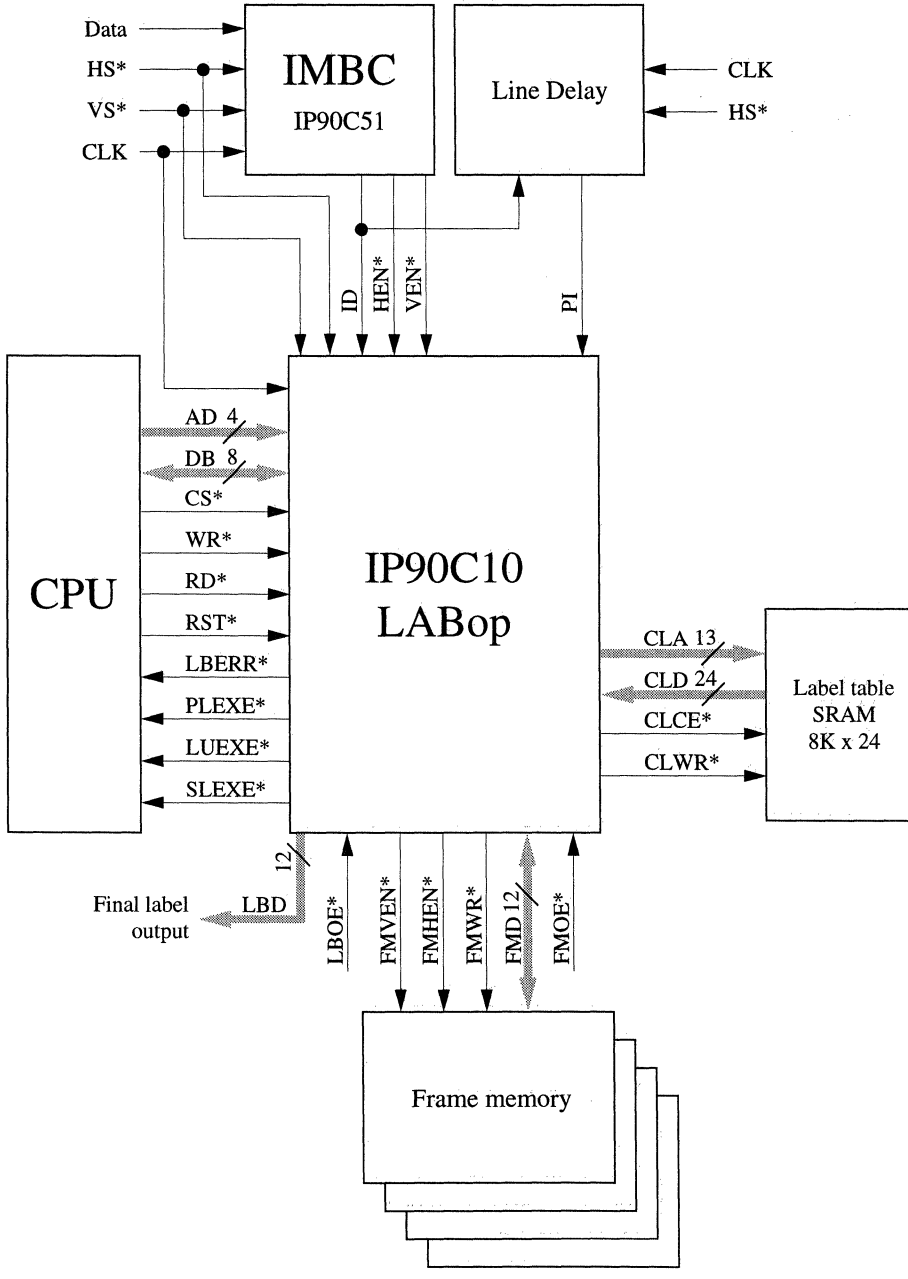
1.2 Features

- Labeling processing functions
  - Labeling processing of non-interlaced raster-scanned binary input data
  - Maximum number of labels:

    Maximum number of temporary labels: 4,094

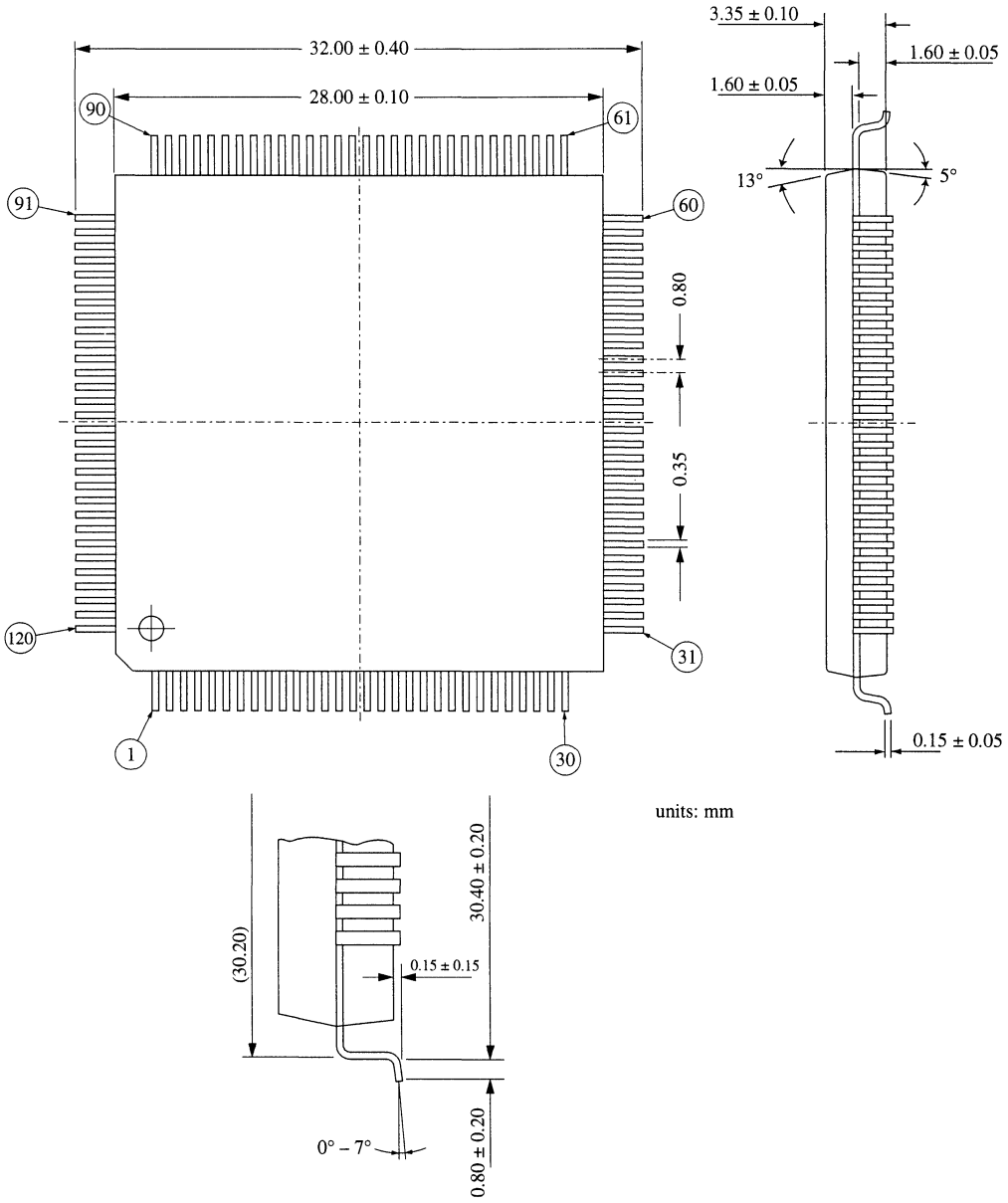
    Maximum number of label linkage information entries: 4,095
  - 4-connectedness/8-connectedness labeling
  - Wide range of image input. The IP90C10 processes up to 512 run labels in a horizontal line. (This does not represent the maximum number of pixels in the horizontal direction. See Section 4.4, "Algorithm," for details.)
  - Practical real-time labeling. Labeling can be processed in real time using three chips in parallel.
  - Automatic labeling function without CPU assistance. The IP90C10 executes a series of processes automatically:
    1. Primary labeling
    2. Linked data combination
    3. Secondary labeling
- External interface
  - Labeling information read out:
    - — Number of temporary labels
    - — Number of label linkage information entries
    - — Number of final labels
  - Built-in timer counter for linked data combination processing
  - Status output to CPU:
    - — Execution status for primary labeling, linked data combination, and secondary labeling
    - — Interrupt generation for overflow (e.g., number of temporary labels) or time-out in each stage of processing (interrupts can be masked as necessary)
  - General-purpose interface using 8-bit data bus
- Maximum operation frequency: f_{max} = 40 MHz
- Power supply: +5V single power supply
- Input/output: TTL-level compatible
- Package: 120-pin QFP

1.3 System Block Diagram



Section 2: Pin Descriptions

2.1 Package Dimensions



120 Pin QFP

2.2 Pin Descriptions

Pin group	Symbol	No. of Pins	Type	Description
Image data input	ID	1	I	Image data input
	HEN*	1	I	Horizontal input image enable signal
	VEN*	1	I	Vertical input enable signal
	HS*	1	I	Horizontal synchronization signal
	VS*	1	I	Vertical synchronization signal
	PI	1	I	1-line delayed image data input
	CLK	1	I	Clock
Label table interface	CLD23–CLDO	24	I/O	Label table data bus
	CLA12–CLA0	13	O	Label table address bus
	CLCE*	1	O	Label table chip enable
	CLWR	1	O	Label table write enable
Frame memory interface	FMD11–FMD0	12	I/O	Temporary label data bus
	FMOE*	1	I	Temporary label output enable
	FMWR*	1	O	Frame memory write enable
	FMVEN*	1	O	Frame memory vertical enable signal
	FMHEN*	1	O	Frame memory horizontal enable signal
Final label output	LBD11–LBD0	12	O	Final label data bus
	LBOE*	1	I	Final label output enable
Status	LBERR*	1	O	Labeling errors
	PLEXE*	1	O	Primary labeling execution frame status
	LUEXE*	1	O	Linked data combination processing execution status
	SLEXE*	1	O	Secondary labeling execution frame status
CPU interface	AD3–AD0	4	I	Address bus
	DB7–DB0	8	I/O	Data bus
	CS*	1	I	Chip select
	WR*	1	I	Write enable
	RD*	1	I	Read enable
	RST*	1	I	System reset (Note 1)
Test	TSTEN0	1	I	Test signal, normally low (Note 2)
	TSTEN1	1	I	Test signal, normally low (Note 2)
Power supply	Vdd	8	PW	5V
	GND	15	PW	Ground
Total pin count		120		

Note 1: RESET must stay low for at least three clock cycles. This pin is provided with a pull-up resistor.

Note 2: The test pins TSTEN0 and TSTEN1 test the LSI's internal logic, and must normally be held low. Note that these pins are provided with pull-down resistors.

Note 3: An asterisk (*) after a pin name indicates negative logic.

2.3 Pin Configuration

Pin No.	Name	Type
1	GND	PW
2	AD0	I
3	AD1	I
4	AD2	I
5	AD3	I
6	GND	PW
7	DB0	I/O
8	DB1	I/O
9	DB2	I/O
10	DB3	I/O
11	DB4	I/O
12	DB5	I/O
13	DB6	I/O
14	DB7	I/O
15	Vdd	PW
16	GND	PW
17	LBERR*	O
18	PLEXE*	O
19	LUEXE*	O
20	SLEXE*	O
21	LBD0	O
22	LBD1	O
23	GND	PW
24	LBD2	O
25	LBD3	O
26	LBD4	O
27	LBD5	O
28	LBD6	O
29	LBD7	O
30	Vdd	PW

Pin No.	Name	Type
31	LBOE*	I
32	GND	PW
33	LBD8	O
34	LBD9	O
35	LBD10	O
36	LBD11	O
37	FMHEN*	O
38	FMVEN*	O
39	FMWR*	O
40	GND	PW
41	FMD0	I/O
42	FMD1	I/O
43	FMD2	I/O
44	FMD3	I/O
45	GND	PW
46	FMOE*	I
47	Vdd	PW
48	FMD4	I/O
49	FMD5	I/O
50	FMD6	I/O
51	FMD7	I/O
52	FME8	I/O
53	FMD9	I/O
54	FMD10	I/O
55	FMD11	I/O
56	GND	PW
57	CLD0	I/O
58	CLD1	I/O
59	CLD2	I/O
60	Vdd	PW

Pin No.	Name	Type
61	GND	PW
62	CLD3	I/O
63	CLD4	I/O
64	CLD5	I/O
65	CLD6	I/O
66	CLD7	I/O
67	CLD8	I/O
68	GND	PW
69	CLD9	I/O
70	CLD10	I/O
71	CLD11	I/O
72	CLD12	I/O
73	CLD13	I/O
74	CLD14	I/O
75	GND	PW
76	Vdd	PW
77	CLD15	I/O
78	CLD16	I/O
79	CLD17	I/O
80	CLD18	I/O
81	CLD19	I/O
82	GND	PW
83	CLD20	I/O
84	CLD21	I/O
85	CLD22	I/O
86	CLD23	I/O
87	GND	I/O
88	CLCE*	O
89	CLWR*	O
90	Vdd	PW

Pin No.	Name	Type
91	CLA0	O
92	CLA1	O
93	CLA2	O
94	CLA3	O
95	CLA4	O
96	CLA5	O
97	GND	PW
98	CLA6	O
99	CLA7	O
100	CLA8	O
101	CLA9	O
102	CLA10	O
103	CLA11	O
104	CLA12	O
105	Vdd	PW
106	CLK	I
107	GND	PW
108	ID	I
109	PE	I
110	HEN*	I
111	VEN*	I
112	HS*	I
113	VS*	I
114	RD*	I
115	RST*	I
116	WR*	I
117	CS*	I
118	TSTEN0	I
119	TSTEN1	I
120	Vdd	PW

Section 3: Registers

3.1 Register List

Address (hex)	Symbol		Read/Write	Description
0	CMD		Read/Write	Control command register
1	STAT		Read only	Execution status register
2	ERRM		Read/Write	Error interrupt mask register
3	ERRS		Read/Write	Error status register
4	TIME	Low	Read/Write	Linked data combination processing timer-count setup register
5		High		
6	Reserved		—	Reserved for system use; do not use
7	Reserved		—	Reserved for system use; do not use
8	TLBN	Low	Read only	Temporary label count register
9		High		
A	RLBN	Low	Read only	Final count register
B		High		
C	ERLN	Low	Read only	Error line register
D		High		
E	CMBN	Low	Read only	Linked data combination information count register
F		High		

3.2 Register Configuration

Control Command Register CMD

Address 0h — Read/Write
Hardware reset: — 00h
Software reset: — 0 except the RESET bit

This register provides the commands necessary to control the LABop.

7	6	5	4	3	2	1	0
4/8D	RESET	LEVEL	IRM	EXE4	EXE3	EXE2	EXE1

4/8D: Selects 4-connectedness (when set to 1) or 8-connectedness (when set to 0) processing.

RESET: Performs a reset. When set to 1, the LABop resets the entire register to 0, except the RESET bit. To clear the reset, write 0 to this bit. When setting the control command register after a reset from the RESET bit, first clear the RESET bit to 0, then write the desired value.

LEVEL: Sets a frame-memory control signal. When set to 0, the LABop selects one clock-pulsed output of the FMVEN* and FMHEN* signals, which indicate the frame-memory-enable period. When pulsed output is selected, the output signal is accompanied by a single pulse immediately preceding and another immediately following the enable period.

When set to 1, the LABop selects level output of FMVEN* and FMHEN*, which holds output held low during the enable period.

(For details, see Section 5, "System Timing." Pulsed signal waveforms are indicated by dotted lines.)

IRM: Disables interrupts. When set to 0, the LABop prevents LBERR* from going low regardless of which type of error occurs.

EXE4: Sets continuous execution of secondary labeling. When the EXE4 and EXE3 bits are set to 1, and EXE2, EXE1, and the STAT register's STAT2 bit are also 0, the LABop starts secondary labeling when the next VS* signal is asserted. The LABop continues executing secondary labeling until EXE4 is cleared to 0.

EXE3: Sets the execution of secondary labeling. When set to 1, and EXE2, EXE1, and the STAT register's STAT2 bit are 0, the LABop starts secondary labeling when the next VS* is asserted. EXE3 is reset to 0 immediately after secondary labeling starts.

EXE2: Sets the execution of linked data combination processing. When set to 1 while primary labeling is not being executed, the LABop immediately starts linked data combination processing. EXE2 is reset to 0 immediately after linked data combination processing starts.

EXE1: Sets the execution primary labeling. When set to 1, the LABop starts primary labeling when the next VS* is asserted. EXE1 is reset to 0 immediately after primary labeling starts.

Execution Status Register STAT

Address 1h Read-only

Reset: XXXXX000h

This register reads the LABop status.

7	6	5	4	3	2	1	0
X	X	X	X	X	STAT3	STAT2	STAT1

STAT3: Indicates secondary labeling status. When set to 1, the frame is a secondary labeling execution frame.

STAT2: Indicates linked data combination processing status. When set to 1, the LABop is executing linked data combination processing.

STAT1: Indicates primary labeling execution status. When set to 1, the frame is a primary labeling execution frame.

X: Reserved. (This bit is undefined when read by the CPU.)

Error Interrupt Mask Register ERRM

Address 2h Read/Write

Reset: 00h

This register masks interrupt requests caused by errors. Setting a bit to 0 prevents the LBERR* pin from going low even when an error occurs. Interrupts can also be disabled by setting the CMD register's IRM bit to 0. This prevents the LBERR* pin from going low regardless of what type of error occurs.

7	6	5	4	3	2	1	0
RLOFM	TMOVM	CMOFM3	CMOFM2	CMOFM1	TLOFM3	TLOFM2	TLOFM1

RLOFM: Masks a run-label overflow error. The LABop memory stores up to 512 run labels per line (see Section 4.4, "Algorithm"). If the RLOFM bit is set to 0, LBERR* cannot go low even when the number of run labels per line exceeds 512.

TMOVM: Masks label linkage information processing overtime error. If set to 0, LBERR* cannot go low even when the label linkage information process exceeds the preset time (see the TIME register).

CMOFM3: Masks a label linkage information overflow error 3. If set to 0, LBERR* cannot go low even when the number of label linkage information entries exceeds 4,095 (register CMBN value = FFFh).

CMOFM2: Masks label linkage information overflow error 2. If set to 0, LBERR* cannot go low even when the number of label linkage information entries exceeds 1,023 (register CMBN value = 3FFh).

CMOFM1: Masks label linkage information overflow error 1. If set to 0, LBERR* cannot go low even when the number of label linkage information entries exceeds 255 (register CMBN value = 0FFh).

TLOFM3: Masks temporary label overflow error 3. If set to 0, LBERR* cannot go low even when the number of temporary labels exceeds 4,094 (register TLBN value = FFFh).

TLOFM2: Masks temporary labels overflow error 2. If set to 0, LBERR* cannot go low even when the number of temporary labels exceeds 1,022 (register TLBN value = 3FFh).

TLOFM1: Masks temporary labels overflow error 1. If set to 0, LBERR* cannot go low even when the number of temporary labels exceeds 254 (register TLBN value = 0FFh).

Error Status Register ERRS

Address 3h Read/Write

Reset: 00h

This register reads the status of errors occurring during labeling. If an error occurs during labeling execution, the bit corresponding to the error is set to 1, and remains at 1 unless the CPU writes 0 to it. When a bit in the ERRS register and the CMD register's IRM bit are set to 1, and the ERRM register bit corresponding to that error is 1, LBERR* is pulled low, and the LABop stops processing. (See Section 4.3, "Error Handling Sequence," for details.)

7	6	5	4	3	2	1	0
RLOFS	TMOVS	CMOFS3	CMOFS2	CMOFS1	TLOFS3	TLOFS2	TLOFS1

RLOFS: Indicates run-label overflow status. Set to 1 when the number of run labels per line exceeds 512.

TMOVS: Indicates label linkage information processing overtime status. Set to 1 when the time required for linked data combination processing exceeds the preset time (see the TIME register).

CMOFS3: Indicates label linkage information overflow status 3. Set to 1 when the number of label linkage information entries exceeds 4,095 (register CMBN value = FFFh).

CMOFS2: Indicates label linkage information overflow status 2. Set to 1 when the number of label linkage information entries exceeds 1,023 (register CMBN value = 3FFh).

CMOFS1: Indicates label linkage information overflow status 1. Set to 1 when the number of label linkage information entries exceeds 255 (register CMBN value = 0FFh).

TLOFS3: Indicates temporary labels overflow status 3. Set to 1 when the number of temporary labels exceeds 4,094 (register TLBN value = FFFh).

TLOFS2: Indicates temporary labels overflow status 2. Set to 1 when the number of temporary labels exceeds 1,022 (register TLBN value = FFFh).

TLOFS1: Indicates temporary labels overflow status 1. Set to 1 when the number of temporary labels exceeds 254 (register TLBN value = FFFh).

Linked Data Combination Processing Timer-Count Setup Register TIME

Address 4h/5h Read/Write
Reset: 00h

The time required for linked data combination processing is proportional to the number of temporary labels or label linkage information entries involved. If the system processes labeling in real time, and noise or a similar problem causes the number of temporary labels or label linkage information entries to be excessively large, processing may need to be stopped. This register sets a time limit (in clock cycles) after which label linkage information processing is stopped.

After label linkage information processing starts, the time-out counter begins counting clock cycles. When the count exceeds the value set by the TIME register multiplied by 256, a time-out error is assumed and the ERRS register's TMOVS bit is set to 1.

	7	6	5	4	3	2	1	0
5h	TIME15	TIME14	TIME13	TIME12	TIME11	TIME10	TIME9	TIME8
4h	TIME7	TIME6	TIME5	TIME4	TIME3	TIME2	TIME1	TIME0

TIME0–15: Sets the time to discontinue label linkage information processing.

Temporary Label Count Register TLBN

Address 8h/9h Read-only
Reset: X000h

This register reads the number of temporary labels. (The value in this register exceeds the generated temporary labels by one.) This register resets each time the LABop starts primary labeling, then counts the temporary labels.

	7	6	5	4	3	2	1	0
9h	X	X	X	X	TLBN11	TLBN10	TLBN9	TLBN8
8h	TLBN7	TLBN6	TLBN5	TLBN4	TLBN3	TBN2	TLBN1	TLBN0

TLBN0–11: Indicates the number of temporary labels.
X: Reserved. (This bit is undefined when read by the CPU.)

Final Label Count Register RLBN

Address Ah/Bh Read-only

Reset: X000h

This register reads the number of final labels. (The value in this register exceeds the generated final labels by one.) This register resets each time the LABop starts primary labeling, then counts the final labels.

	7	6	5	4	3	2	1	0
Bh	X	X	X	X	RLBN11	RLBN10	RLBN9	RLBN8
Ah	RLBN7	RLBN6	RLBN5	RLBN4	RLBN3	RLBN2	RLBN1	RLBN0

RLBN0–11: Indicates the number of final labels.

X: Reserved. (This bit is undefined when read by the CPU.)

Error Line Register ERLN

Address Ch/Dh Read-only

Reset: X000h

This register reads the number of horizontal lines when an error occurs. This register resets each time the LABop starts primary labeling, then counts the horizontal lines.

	7	6	5	4	3	2	1	0
Dh	X	X	X	X	ERLN11	ERLN10	ERLN9	ERLN8
Ch	ERLN7	ERLN6	ERLN5	ERLN4	ERNL3	ERLN2	ERLN1	ERLN0

ERLN0–11: Indicates the value of the vertical address at the time an error occurs.

X: Reserved. (This bit is undefined when read by the CPU.)

Linked Data Combination Count Register CMBN

Address Eh/Fh Read-only

Reset: X000h

This register reads the number of label linkage information entries. This register is reset each time the LABop starts primary labeling, then counts the label linkage information entries.

	7	6	5	4	3	2	1	0
Fh	X	X	X	X	CMBN11	CMBN10	CMBN9	CMBN8
Eh	CMBN7	CMBN6	CMBN5	CMBN4	CMBN3	CMBN2	CMBN1	CMBN0

CMBN0–11: Indicates the number of label linkage information entries.

X: Reserved. (This bit is undefined when read by the CPU.)

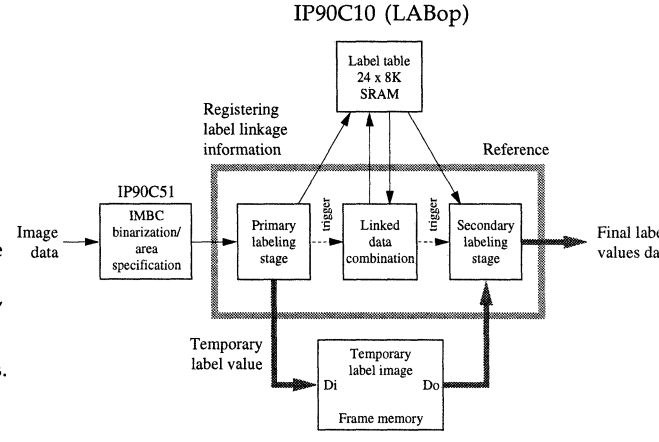
Section 4: Functional Description

4.1 Description

The LABop labels raster-scanned binary image data, and includes these three functions in a single chip:

1. Primary labeling
2. Linked data combination
3. Secondary labeling

The LABop provides individual command and status bits for these three functions. This allows each function to execute independently, and to run automatically and continuously by simultaneously applying the executing commands.



Internal Processing Block Diagram

4.2 Device Operation

Primary Labeling Processing

The LABop performs primary labeling on raster-scanned binary image data. When the CPU sets the CMD register EXE1 bit to 1, the LABop enters primary labeling mode when the next VS* signal is asserted.

When in primary labeling mode, the LABop receives image data for the effective area defined by the VEN* and HEN* signals, and simultaneously outputs temporary labels to the frame memory and label linkage information to the label table. After starting processing, the LABop terminates primary labeling mode when the next VS* is asserted. The effective area must contain at least two lines (i.e., there must be two HEN* periods during the VEN* period). Also, an additional HEN* period is required to process the last line of the effective area. The LABop operation is not affected by disabling VEN*. (For details, see Section 5, "System Timing.")

Note that the LABop minimizes the number of temporary labels required, by using a proprietary algorithm for handling large images. (For details, see Section 4.4, "Algorithm.")

The LABop outputs the value 0 (background value) for all input data that falls outside the effective area, regardless of the input data value.

Linked Data Combination Processing

The LABop enters label linkage information processing mode when the CMD register's EXE2 bit is set to 1. If this occurs when the EXE1 bit is 1 or primary labeling is under way, the LABop enters label linkage information processing mode only after primary labeling processing of the effective area is done. Otherwise, the LABop enters the mode immediately.

In this mode, the LABop processes the label linkage information in the label table. This creates a conversion table (look-up table) used to convert between final and temporary labels.

The LABop ends label linkage information processing mode after all processing is done.

Secondary Labeling Processing

The LABop enters secondary labeling processing mode when the CMD register's EXE3 bit is set to 1. If this occurs when label linkage information processing is underway, secondary labeling processing mode is entered only after the current processing is completed and the next VS* is asserted. Otherwise, the mode is entered immediately after the next VS* is asserted.

In this mode, the LABop executes secondary labeling for the temporary label data in the frame memory by referring to the look-up table in the label table.

The LABop ends this mode when the next VS* signal is asserted. Note that if the EXE4 bit is set to 1, the LABop continues processing even if VS* is asserted. If the EXE4 bit is set to 0, the LABop ends secondary labeling processing mode when the next VS* is asserted.

Error Handling

If an error occurs while the CMD register's IRM bit and the error mask bit corresponding to the error are both set to 1, the LABop asserts the LBERR* signal and stops operation. Refer to the ERRS register to find the cause of the error. Restart the LABop by applying a system reset.

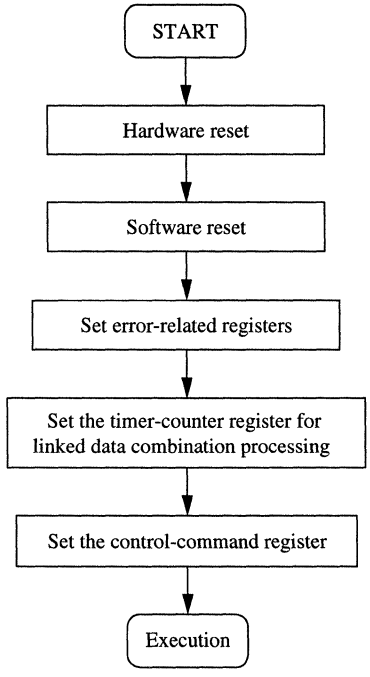
Reset

This operation initializes the LABop's registers and internal status. Bi-directional pins are placed in high-impedance state.

The final label output LBDs are set to output (FFFFh) when the LBOE* signal is low, and to high-impedance state when LBOE* is high. Status outputs are deasserted. During a reset, all inputs except RST* are invalid.

4.3 Control Sequence

Internal Register Write Sequence



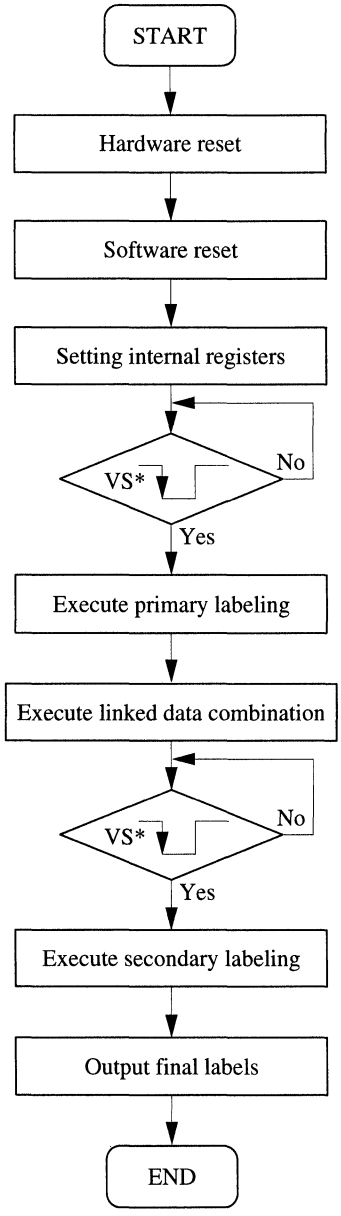
Set the ERRM and ERRS registers to their designated states.

Set the TIME register to the designated value.

Set the CMD register to its designated state.

If the CMD register is set to 07h, for example, a series of operations from primary labeling to secondary labeling is automatically executed using 8-connectedness labeling operation.

Labeling Execution Sequence



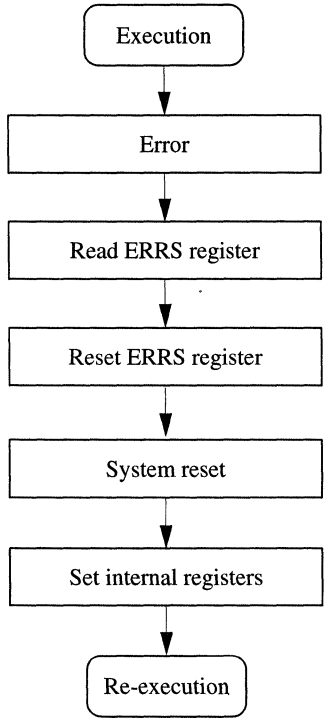
The LABop begins primary labeling when VS* is asserted, and outputs temporary label data to the frame memory and label linkage information to the label table.

Immediately after primary labeling finishes, the LABop begins linked data combination to create a look-up table (in the label table) that shows the relationship between temporary and final labels.

When linked data combination is complete, the LABop begins secondary labeling when the next VS* is asserted, and converts temporary labels to final labels.

Final labels are output.

Error-Handling Sequence



If an error occurs, the LABop asserts status output signal LBERR* by pulling it low, then stops processing.

Reading the ERRS register shows the cause of the error.

Clear the error by writing 0 to the ERRS register.

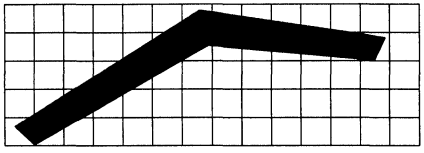
Reset the system using a hardware or software reset.

4.4 Algorithm (Primary Labeling Processing)

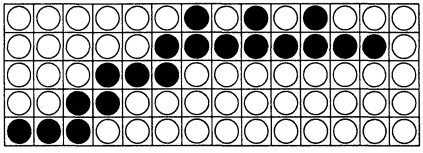
Assigning Temporary Labels

The IP90C10 minimizes the number of temporary labels required. For example, if the target image shown in (a) below is read into the system from a camera, the pixels in the data do not always form a smooth-edged line (due to jags or noise) as shown in (b). In such a case, a conventional algorithm generates varying temporary labels, as shown in (c). However, the LABop applies the single label shown in (d) by recognizing that the pixels of the object to be read from a camera are connected.

(a)
Target image



(b)
Entered data
(binary image)



(c)
Example of temporary labels by conventional simple raster-scan operation

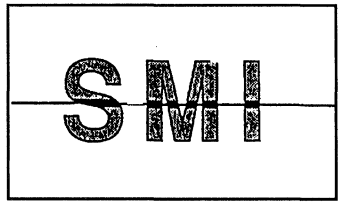
						1		2		3			
					1	1	1	1	1	1	1		
			4	1	1								
		4	4										
5	4	4											

(d)
Temporary labels by the LABop

						1		1		1			
					1	1	1	1	1	1	1		
			1	1	1								
		1	1										
1	1	1											

Run Label

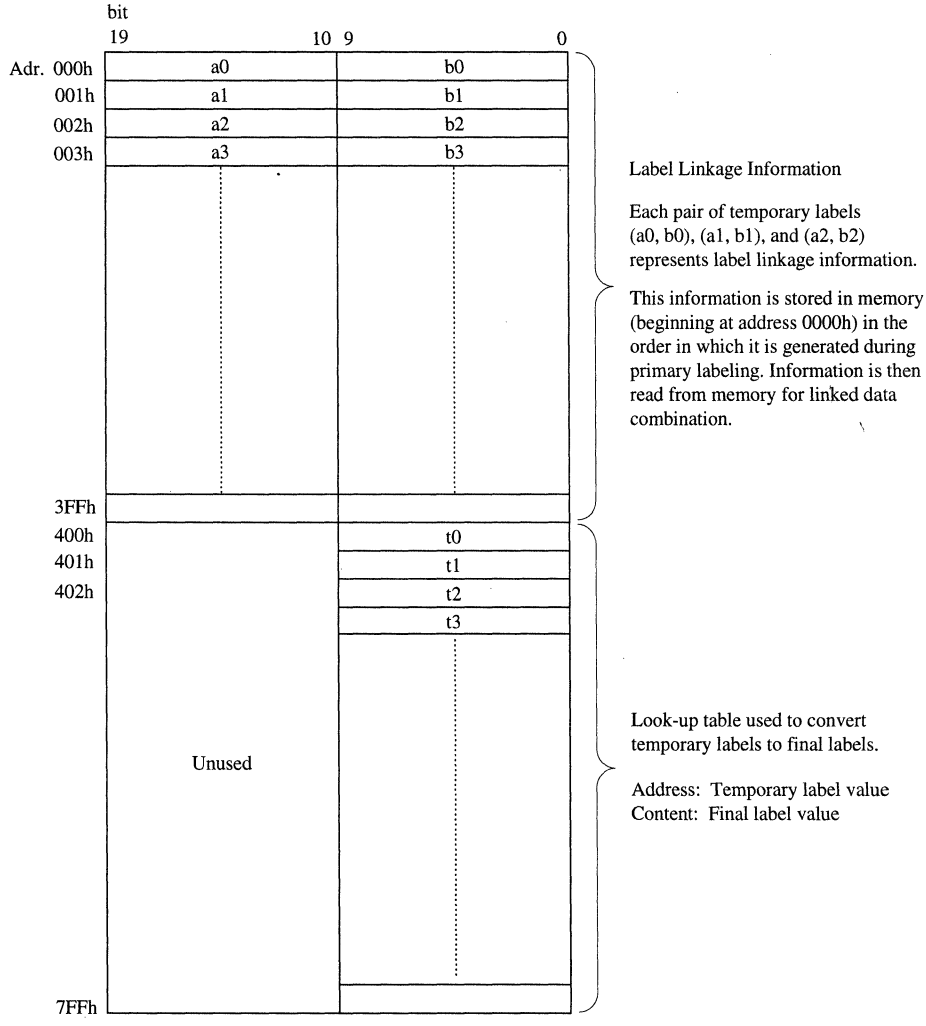
The LABop uses a "run label" to minimize the temporary labels during primary processing. A row of consecutive logical 1s of input data is called a "run," and the preliminary label information assigned to each run is called a "run label." The LABop incorporates 512 run label buffers, and so can process up to 512 runs per line.



The thick line portions of the line in the figure to the left indicate runs. The line contains six runs.

Therefore, in practical application, the maximum number of processed pixels in the horizontal directions is much greater than the worst case for alternating logical 1s and 0s, which is 512 x 2 = 1024 pixels.

4.5 Label Table Format

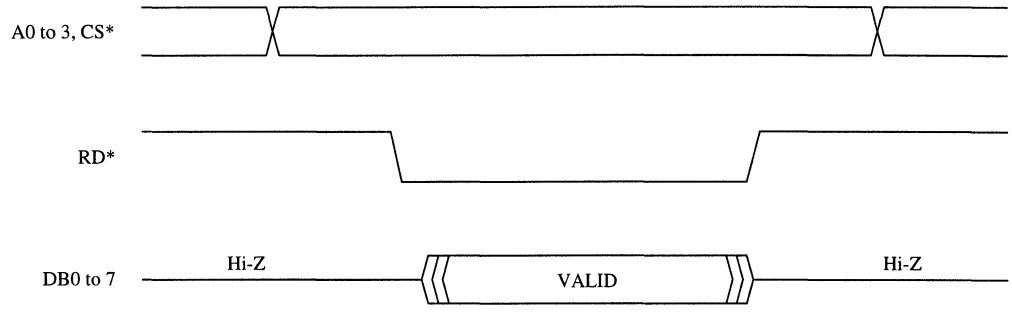


To store label linkage information, memory space equivalent to the number of label linkage information entries is allocated, beginning with address 0000h. Memory space equivalent to the number of temporary labels is allocated in the look-up table, beginning with address 1000h. The contents of the unused addresses in the label linkage information and look-up-table memory spaces are undefined.

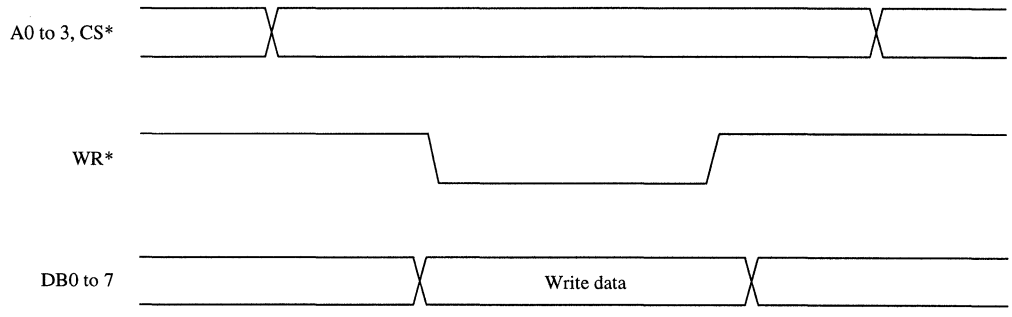
Section 5: System Timing

5.1 CPU Interface

Read Cycle Timing

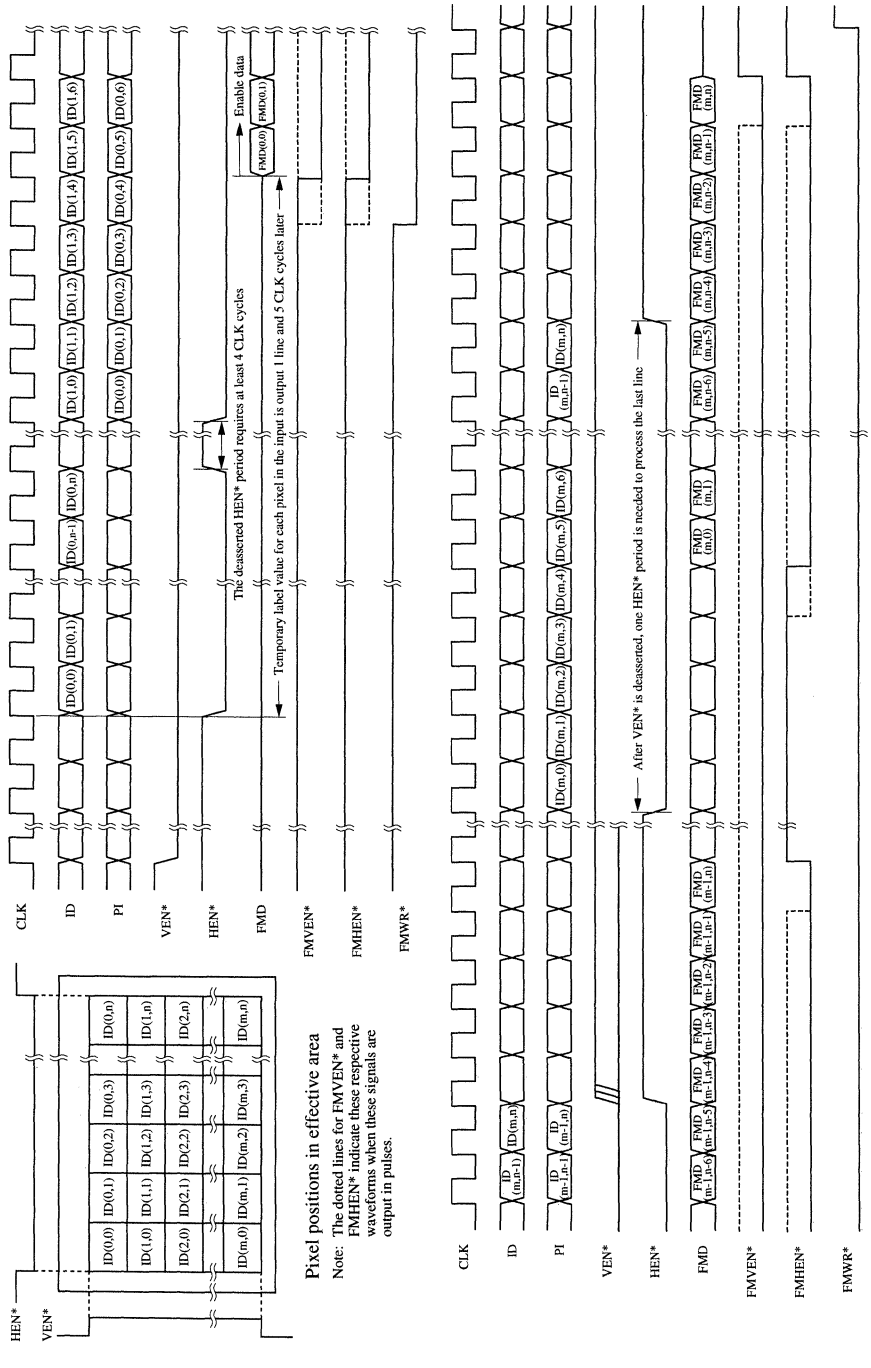


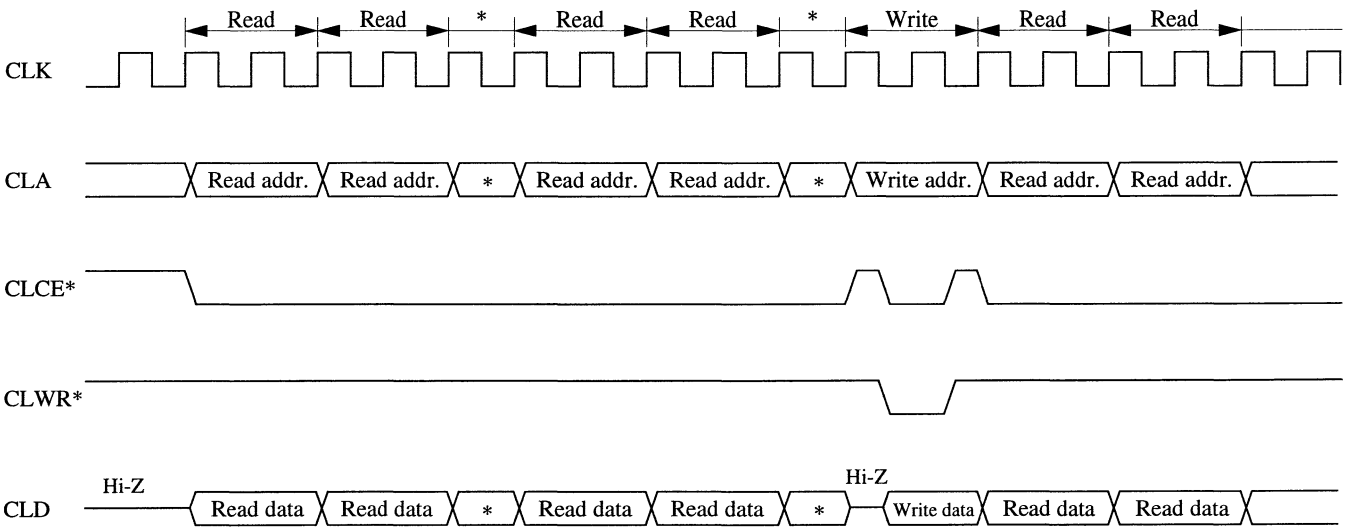
Write Cycle Timing



Note: The CPU interface signals are not synchronized with the image clock.

5.2 Primary Labeling Processing

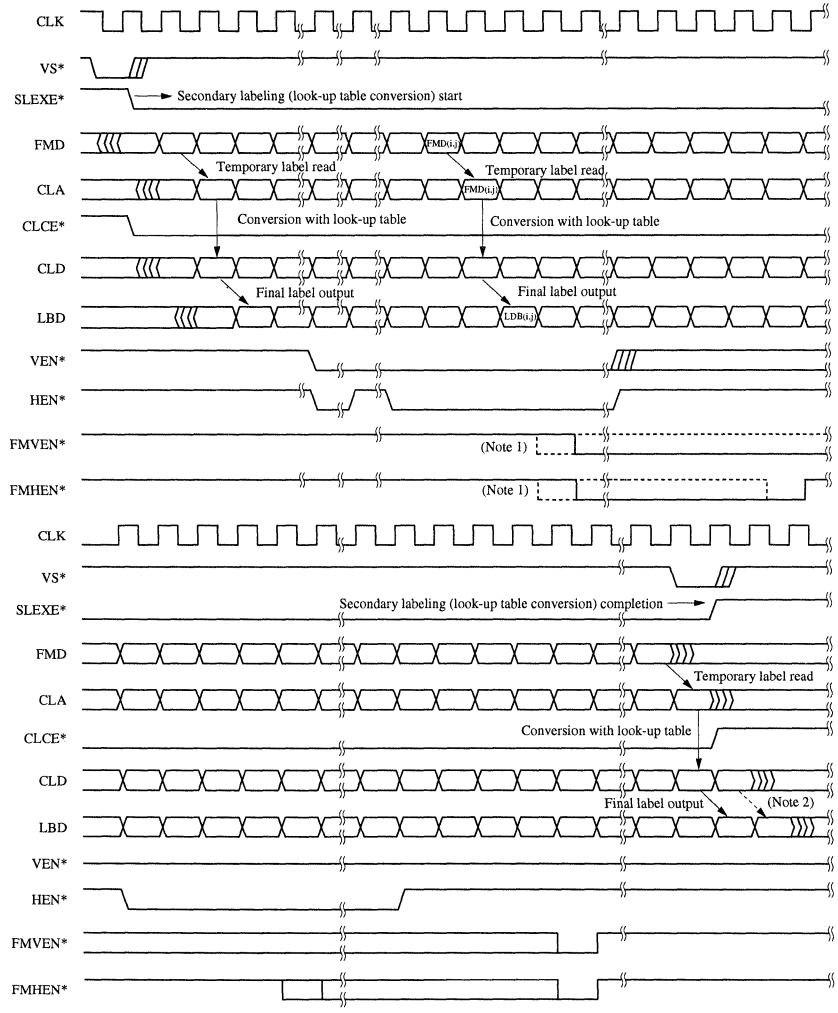




Note: Linked data combination requires two clock cycles each for the read and and the write cycles. A single dummy cycle, marked by *, can be inserted during a read operation. CLA and CLD are indeterminate during this dummy cycle.

5.4 Secondary Labeling Processing

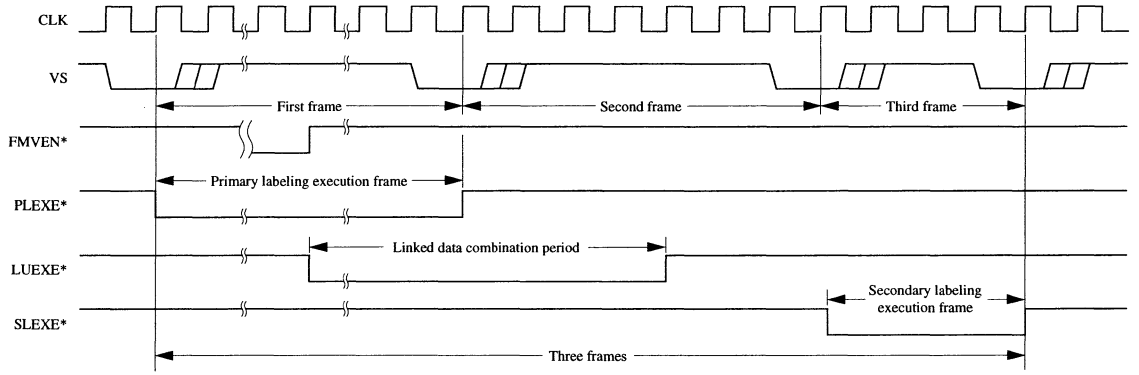
Temporary labels can be converted to final labels using the look-up table during the VS*-to-VS* interval. The LABop always latches the FMD value as long as SLEXE* is held low, and outputs to CLA. At the same time, the LABop latches the CLD value and outputs it to LBD. While SLEXE* is held low, temporary labels can be entered into FMD, so system timing can be set as required. The FMVEN* and FMHEN* timing relative to VEN* and HEN* are the same as in primary labeling.



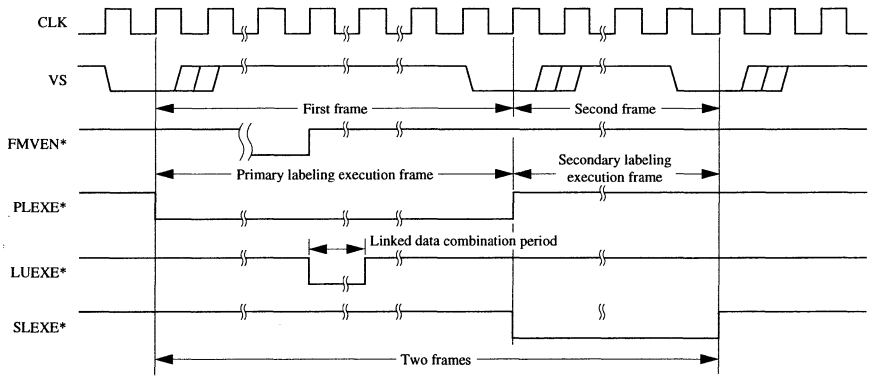
Note 1: The dotted lines for FMVEN* and FMHEN* indicate the respective waveforms when these signals are output in pulses.

Note 2: For high-speed systems that may require two clock cycles for converting temporary labels to final labels using the look-up table (with an external latch provided for CLA to extend SRAM access time), an additional operating cycle to latch CLD and output it to LBD is provided here.

a) Linked data combination processing is not completed within the primary labeling execution frame interval.



b) Linked data combination processing is completed within the primary labeling execution frame interval.



Description of status signals

PLEXE* is asserted and held low during the primary labeling execution frame interval.

LUEXE* is asserted immediately after primary labeling of the effective area is completed, and is de-asserted when linked data combination is complete.

SLEXE* is asserted and held low during the secondary labeling execution frame interval.

Section 6: Electrical Characteristics

6.1 Absolute Maximum Ratings

(Referenced to GND, Ta = 25°C unless otherwise specified)

Parameter	Symbol	Rating	Unit
DC supply voltage	Vdd	-0.3 to 6.5	V
Input/output voltage	VIN	-0.3 to Vdd + 0.3	V
Operating temperature	TOPT	0 to 70	°C
Storage temperature	TSTG	-10 to 80	°C

6.2 Recommended Operating Conditions

(Ta = 0 to 70°C)

Parameter		Symbol	Condition	Min.	Typ.	Max.	Unit
Power-supply voltage		Vdd		4.75	5.0	5.25	V
Input voltage	Range	VI	TTL level normal input	0		Vdd	V
	High level	VIH		2.2		Vdd	V
	Low level	VIL		0		0.8	V
Input rising time		TRI	TTL level normal input	0		200	ns
Input falling time		TFI		0		200	ns

6.3 Input/Output Capacitance

Parameter	Symbol	Condition	Typ.	Unit
Input capacitance	CIN	Any input (see Note)	10	pF
Output capacitance	COUT	Any output	10	pF

Note 1: Does not apply to bi-directional buffers.

6.4 DC Characteristics

Vdd = 5 V ± 5%
Ta = 0 to 70°C
GND = 0 V

Symbol	Parameter		Condition	Min.	Typ.	Max.	Unit
VIL	Low level input voltage					0.8	V
VIH	High level voltage			2.0			V
IIN	Input current	Without pull-up resistor Without pull-down resistor	VIN = Vdd or GND	-10	±1	10	µA
		With pull-down resistor	VIN = Vdd	35	110	335	µA
		With pull-up resistor	VIN = GND	-35	-115	-350	µA
VOH	High level output voltage (Note 1)		IOH = -4 mA	2.4	4.5		V
			IOH = -6 mA	2.4	4.5		V
VOL	Low level output voltage (Note 1)		IOL = 4 mA		0.2	0.4	V
			IOL = 6 mA		0.2	0.4	V
IOZ	Off-state leakage current		VOH = Vdd or GND	-10	±1	10	µA
IOS	Output short circuit current (Note 2)		Vdd = Max, Vo = Vdd	15	50	130	mA
			Vdd = Max, Vo = 0 V	-5	-25	-100	mA

Note 1: IOL = 6 mA and IOH = -6 mA for the FMD11–0 and LBD11–0 pins; 4 mA for other pins.

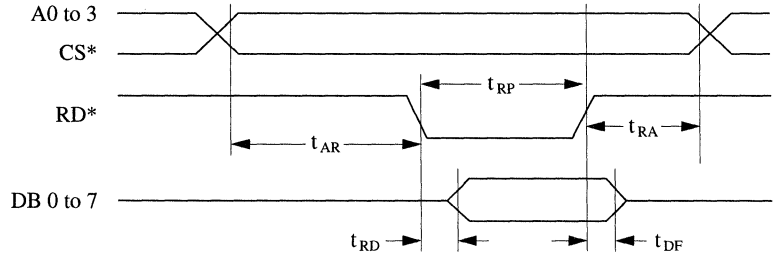
Note 2: When referring to the 4-mA buffers in Note 1, this parameter defines the current that flows when a high output is shorted to GND. Make all efforts to ensure that two or more outputs are never shorted to GND simultaneously. If this nevertheless occurs, the shorting time must be no more than 1 second.

6.5 AC Characteristics

(Note: The load capacitance of all output pins is assumed to be 30 pF.)

CPU Interface

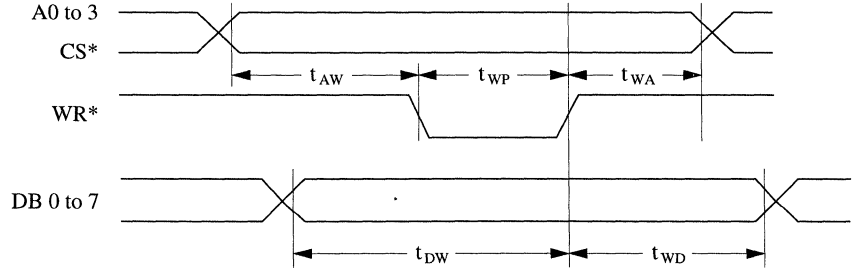
Read Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
A and CS* setup times to RD* falling	t_{AR}	5	—	—
A and CS* hold times to RD* rising	t_{RA}	0	—	—
RD* pulse width	t_{RP}	30	—	—
Delay time from RD* falling to DB enable	t_{RD}	5	—	20
Delay time from RD* rising to DB placed in Hi-Z	t_{DF}	3	—	15

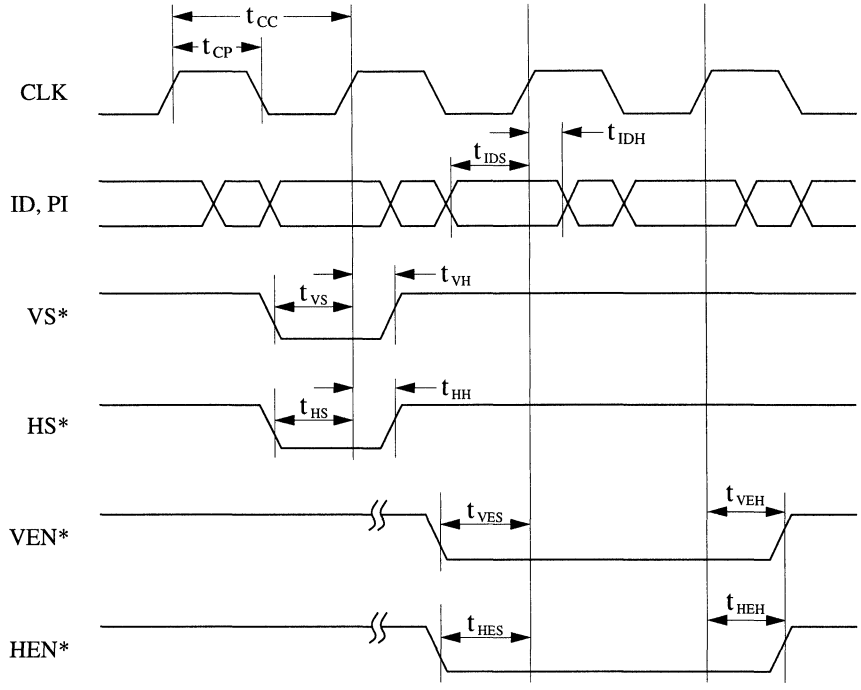
Write Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
A and CS* setup times to WR* falling	t_{AW}	5	—	—
A and CS* hold times to WR* rising	t_{WA}	0	—	—
WR pulse width	t_{WP}	30	—	—
DB setup time to WR* rising	t_{DW}	15	—	—
DB hold time to WR* rising	t_{WD}	0	—	—

Image Data Input

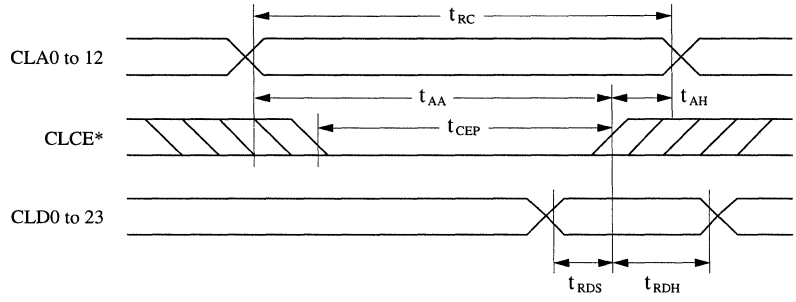


unit: ns

Parameter	Symbol	Min.	Typ.	Max.
CLK cycle	t_{CC}	25	—	—
High-level period of CLK	t_{CP}	10	tcc/2	tcc-10
ID and PI setup times to CLK rising	t_{IDS}	8	—	—
ID and PI hold times to CLK rising	t_{IDH}	5	—	—
VS* falling setup time to CLK rising	t_{VS}	8	—	—
VS* rising hold time to CLK rising	t_{VH}	5	—	—
HS* falling setup time to CLK rising	t_{HS}	8	—	—
HS* rising hold time to CLK rising	t_{HH}	5	—	—
VEN* falling setup time to CLK rising	t_{VES}	8	—	—
VEN* rising hold time to CLK rising	t_{VEH}	5	—	—
HEN* falling setup time to CLK rising	t_{HES}	8	—	—
HEN* rising hold time to CLK rising	t_{HEH}	5	—	—

Label Table Interface

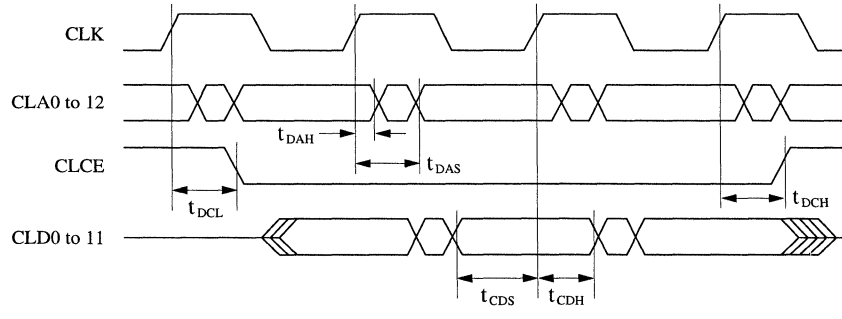
Read Cycle 1 (label linkage information processing)



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Read cycle time	tRC	2tcc-5	2tcc	—
CLA access time to CLCE* rising	tAA	30	—	—
CLA hold time to CLCE* rising	tAH	5	—	—
CLCE* pulse width	tCEP	2tcc-10	2tcc	—
CLD setup time to CLCE* rising	tRDS	4	—	—
CLD hold time to CLCE* rising	tRDH	5	—	—

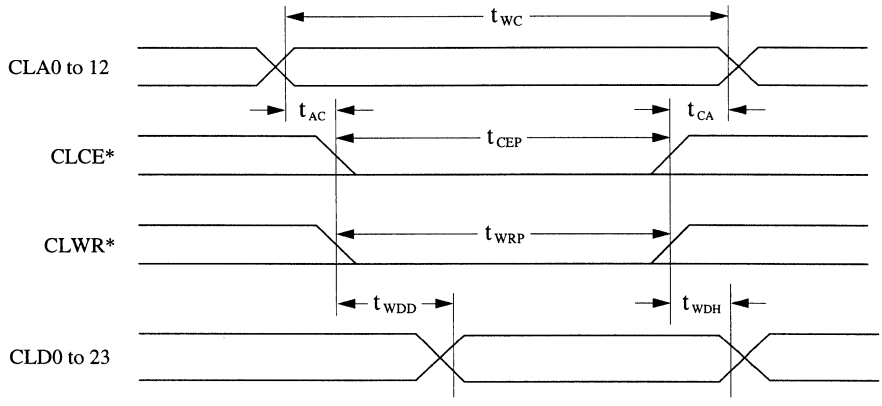
Read Cycle 2 (secondary labeling)



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
CLA hold time to CLK rising	tDAH	5	—	—
Delay time from CLK rising to CLA enable	tDAS	—	—	20
Delay time from CLK rising to CLCE* falling	tDCL	—	—	20
Delay time from CLK rising to CLCE* rising	tDCH	—	—	20
CLD setup time to CLK rising	tCDS	4	—	—
CLD hold time to CLK rising	tCDH	5	—	—

Write Cycle

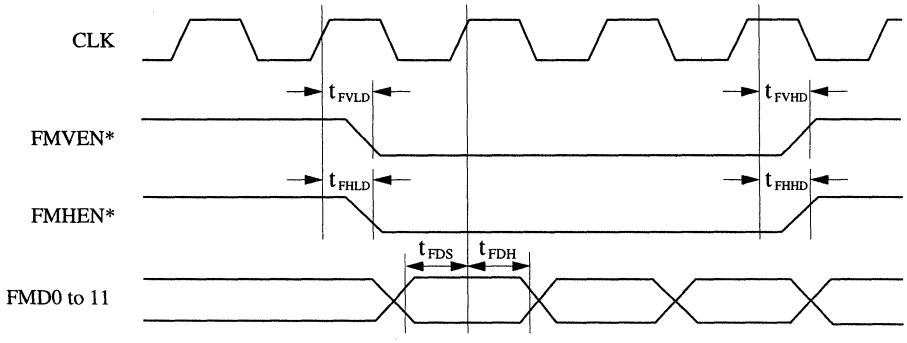


unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Write cycle time	t_{WC}	2tcc-5	2tcc	—
CLA setup time to CLCE* and CLWR* falling	t_{AC}	5	—	—
CLCE* pulse width	t_{CEP}	tcc-5	tcc	—
CLA hold time to CLCE* and CLWR* rising	t_{CA}	5	—	—
CLWR* pulse width	t_{WRP}	tcc-5	tcc	—
Delay time from CLCE* and CLWR* falling to CLD	t_{WDD}	—	—	17
CLD hold time to CLCE* and CLWR* rising	t_{WDH}	5	—	—

Frame Memory Interface

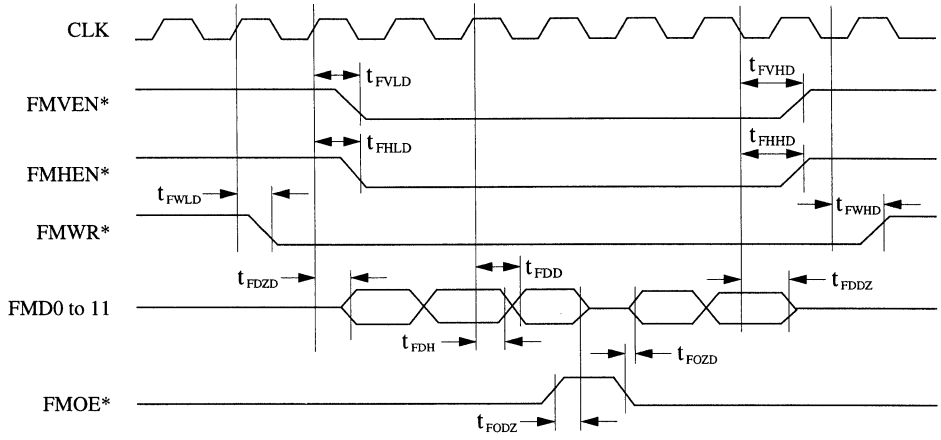
Read Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Delay time from CLK rising to FMVEN* falling	t_{FVLD}	—	—	20
Delay time from CLK rising to FMVEN* rising	t_{FVHD}	—	—	20
Delay time from CLK rising to FMHEN* falling	t_{FHLD}	—	—	20
Delay time from CLK rising to FMHEN* rising	t_{FHHD}	—	—	20
FMD setup time to CLK rising	t_{FDS}	4	—	—
FMD hold time to CLK rising	t_{FDH}	5	—	—

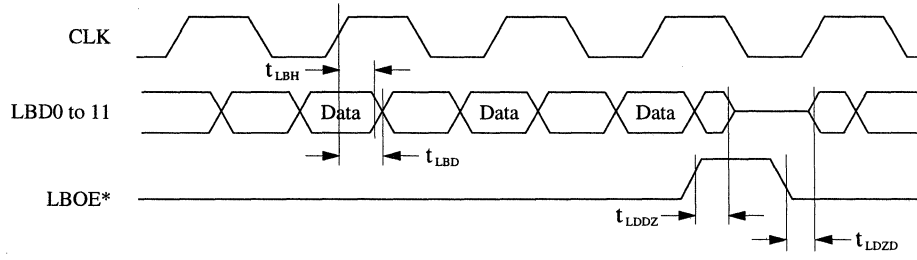
Write Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Delay time from CLK rising to FMVEN* falling	t_{FVLD}	—	—	20
Delay time from CLK rising to FMVEN* rising	t_{FVHD}	—	—	20
Delay time from CLK rising to FMHEN* falling	t_{FHLD}	—	—	20
Delay time from CLK rising to FMHEN* rising	t_{FHHD}	—	—	20
Delay time from CLK rising to FMWR* falling	t_{FWLD}	—	—	20
Delay time from CLK rising to FMWR* rising	t_{FWHD}	—	—	20
Delay time from CLK rising to FMD* enable	t_{FDZD}	—	—	20
Delay time from CLK rising to FMD* placed in Hi-Z	t_{FDDZ}	—	—	20
FMD hold time to CLK rising	t_{FDH}	4	—	—
Delay time from CLK rising to valid FMD	t_{FDD}	—	—	20
Delay time from FMOE* rising to FMD placed in Hi-Z	t_{FODZ}	—	—	20
Delay time from FMOE* falling to valid FMD	t_{FOZD}	—	—	20

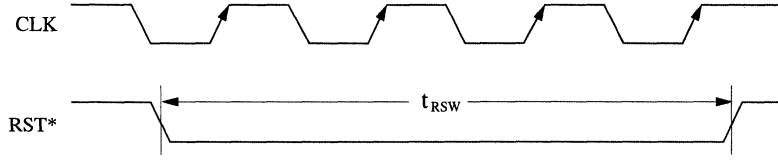
Secondary Labeling



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
LBD hold time to CLK rising	tLBH	4	—	—
Delay time from CLK rising to valid LBD	tLBD	—	—	15
Delay time from LBOE* rising to LBD* placed in Hi-Z	tLDDZ	—	—	15
Delay time from LBOE* falling to valid LBD	tLDZD	—	—	15

Reset Timing



The reset signal must be held active for at least three clock cycles.

unit: clock cycles

Parameter	Symbol	Min.	Typ.	Max.
RST* pulse width (Low period)	tRSW	3	—	—

Section 7: Sample Application

7.1 Sample Labeling Result

The LABop executes primary and secondary labeling in real time. However, the time required for label linkage information processing depends on the input image data and the clock rate. The table below lists the clock cycles required for label linkage information processing, as well as the time required (in ms) for the LABop to execute label linkage information processing on the sample images in Photos 1 through 3.

Example of Labeling Results

Image	No. of temp. labels	Number of final labels	Number of label linkage information entries	Linked data combination			
				Required clock cycles	Required time (ms)		
					15 MHz	20 MHz	40 MHz
Photo 1	3234	2227	1285	39.5k	2.63	1.98	0.99
Photo 2	650	366	297	8.5k	0.57	0.43	0.21
Photo 3	800	50	750	15k	1.03	0.77	0.38

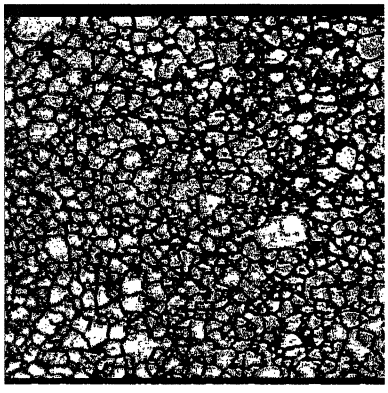


Photo 1: Crystal surface

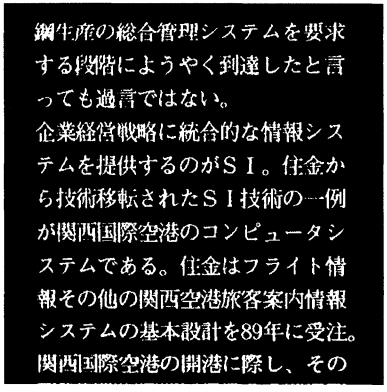


Photo 2: Text data

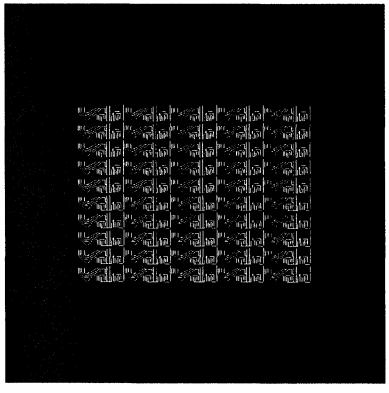
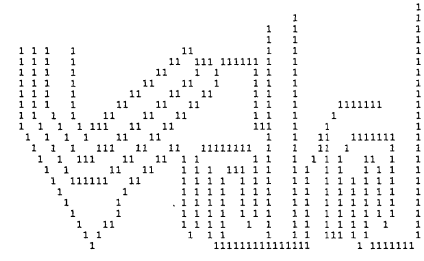
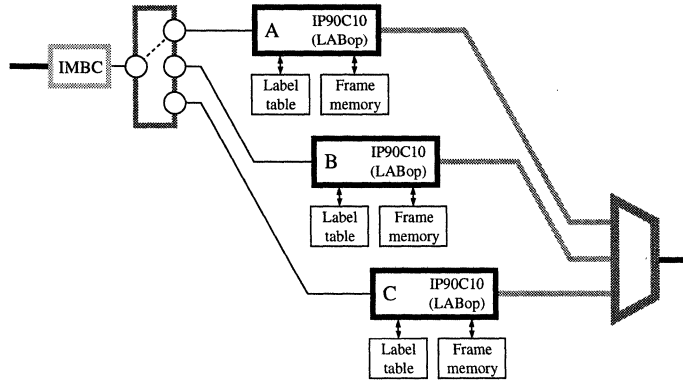


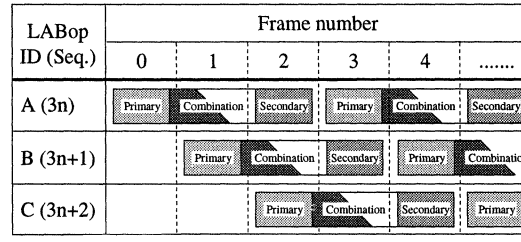
Photo 3: Hattori Pattern, in a 5 x 10 Array (Note)

Note: The diagram below is from T. Hattori, "New Regional Labeling Algorithm for Pipeline Image Processors and Proof of Correctedness", in proc. IECON '91, pages 2005–2010, 1991.

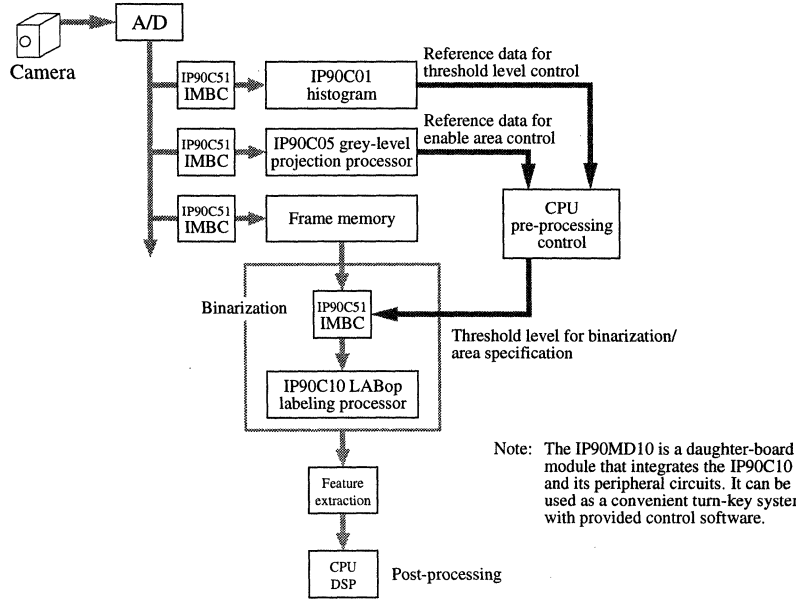


7.2 Real-Time Parallel Processing



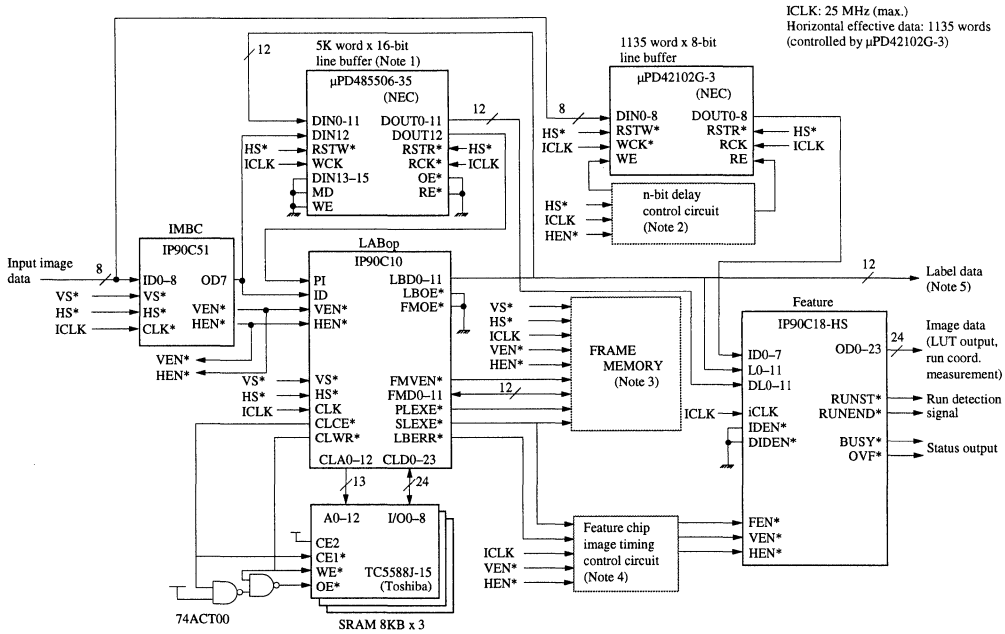


7.3 Sample Application Using an Image-Processing Device



Note: The IP90MD10 is a daughter-board module that integrates the IP90C10 and its peripheral circuits. It can be used as a convenient turn-key system with provided control software.

7.4 Reference Circuits (Labeling Processing + Feature Extraction Processing) IP90C51 + IP90C10 + IP90C18



Note 1: This line buffer is used as a 1H line delay for the IP90C10 and IP90C18.

Note 2: The IP90C10 requires a frame memory design that generates a horizontal pixel delay factor. This line buffer is required to absorb that pixel delay factor. The delay value (n) depends on the frame memory design.

Note 3: This is the frame memory required by the IP90C10.

Note 4: This determines the processing area for the IP90C18. Use IP90C51s as necessary. In certain cases where the IP90C10 executes secondary labeling only, the circuit can be designed to function as if only the IP90C18 were operating.

Note 5: Frame memory should be controlled so that the input image data and label data appear on the same horizontal line (with no 1-line delay differential).

Appendix A: Functional Description and Algorithms for LABop

New Approach to and Implementation of an LSI for High-Speed Image Labeling

Nobuo Hayashi, Hiroshi Nittaya, Masahiro Kohno, and Masahiro Kato

Sumitomo Metal Industries, Ltd.
1–8 Fuso-cho Amagasaki Hyogo 660 Japan

Abstract: Image labeling plays an important role in image analysis and pattern recognition. This paper presents the use of a single accelerator LSI chip for image labeling, and proposes a new primary labeling algorithm suitable for hardware implementation. The algorithm determines primary labels by using a simple 2 x 2 matrix window. The LSI performs all labeling processes, including primary labeling, labeling unification, and secondary labeling. It runs at 40 MHz, and can generate up to 4094 primary labels for use in 4- or 8-connectivity.

A.1 Introduction

High-speed labeling of connected components in a binary image is one of the most fundamental problems in image analysis and pattern recognition.

Many algorithms and hardware designs have attempted to achieve high-speed labeling ([1]–[4]). Labeling connected components based on raster scanning consists of three stages: primary labeling, labeling unification, and secondary labeling. In primary labeling, a two-dimensional binary image is scanned from upper left to lower right to generate temporary labels. When the primary labeling process generates different temporary labels on a connected component and later recognizes them to be linked or part of the same connected pattern, the process then generates label linkage information that represents two temporary labels to be linked. In the linked data combination stage, a look-up table is generated by unifying label linkage information, which represents the relationship between temporary labels and final labels. Finally, secondary labeling generates final labels by referring to the look-up table and relabeling temporary labels accordingly.

Conventional algorithms for primary labeling concentrate mainly on achieving high-speed label unification by reducing the number of temporary labels and using relatively large matrix windows ([1], [2]). Previous papers have focused on the label unification process, though they dealt mainly with processing using software, which requires a large two-dimensional memory or complicated memory access to obtain a look-up table ([3], [4]).

This appendix presents a method of using a single LSI chip for high-speed image labeling, and proposes a new algorithm for primary labeling. Section A.2.1, "Primary Labeling," discusses this new algorithm, which determines temporary labels by using a simple 2 x 2 matrix window to simplify the hardware and achieve high-speed processing, while still using the same number of temporary labels as conventional methods. Section A.2.2, "Linked Data Combination," then presents the linked data combination algorithm, which requires one-dimensional memory and is suitable for use in hardware. Finally, Section A.3, "Hardware Architecture," describes the hardware and specifications of the labeling LSI.

A.2 Algorithms

A.2.1 Primary Labeling

In this discussion, two-dimensional image input is represented by a binary number in which pixels marked 1 denote the target, and the 0-pixels denote the background. In primary labeling, a matrix window scans a binary image from upper left to lower right to generate temporary labels and label linkage unification. The primary labeling algorithm uses a simple 2 x 2 matrix window (see Figure A-1) to recognize the target image and its connectivity, and then determines temporary labels according to the 16 conditions of the matrix window shown in Figure A-2.

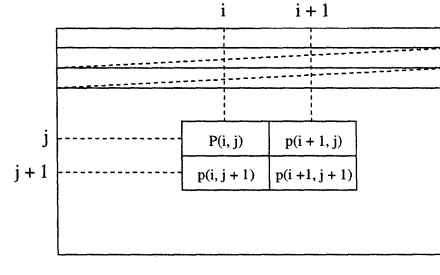


Figure A-1: Raster Scanning a 2 x 2 Matrix Window

The operations "R<-f(RLB)" and "R<- g(R, RLB)" determine temporary labels (the notation "<--" represents a substitution):

R <-- f(RLB):
  X <-- RLB
  if X = 0 then R <-- LC and LC <-- LC + 1
    else R <-- X

R <-- g(R, RLB):
  X <-- RLB
  (1) if X = 0, R = 0, then R <-- LC and LC <-- LC+1
  (2) if X ≠ 0, R = 0, then R <-- X
  (3) if X ≠ 0, R ≠ 0, then do nothing
  (4) if X ≠ 0, R ≠ 0, X = R, then do nothing
  (5) if X ≠ 0, R ≠ 0, X ≠ R, then LIB <-- X, R and R <-- min(X, R)

where:

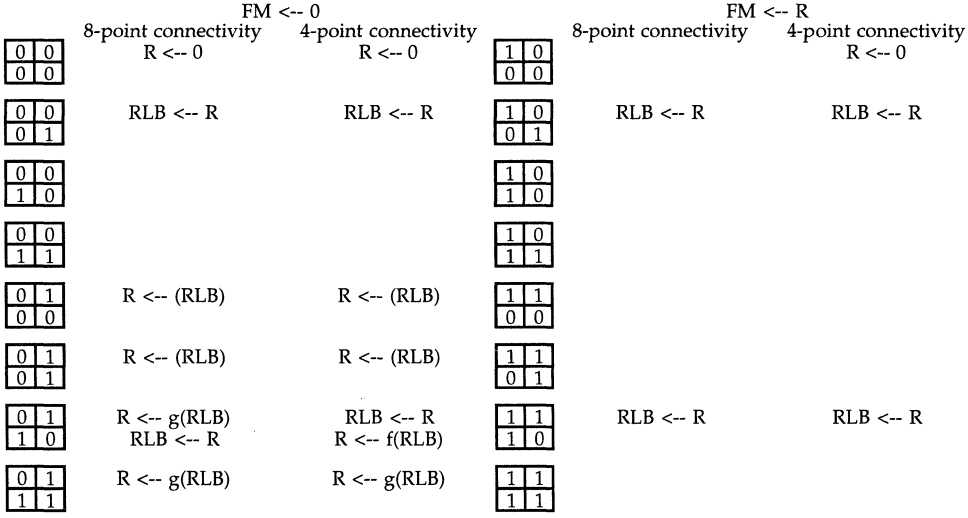
R:	Temporary label register value.
X:	Run label register value.
RLB:	Run label buffer value. This buffer stores run labels and works as FIFO memory.
LIB:	Linkage information buffer value. This buffer stores label linkage information.
LC:	Label counter value that represents the number of temporary labels.

The temporary label register value R is terminated according to the procedures shown in Figure A-2. When the value in the upper-left of the matrix window is 0 (background), 0 is stored in frame memory; when the value in the upper-left of the matrix window is 1 (1 indicates the target image), R is stored in frame memory as a temporary label. Label linkage information is also stored in the linkage information buffer when it is generated.

This algorithm recognizes connectivity, and gives a temporary label for the image that has a depression 1-pixel depth or diagonal pattern, thereby reducing the number of temporary labels generated. The algorithm corresponds to both 4- and 8-point connectivity by slightly modifying the procedure shown in Figure A-2.

A.2.2 Linked Data Combination

The linked data combination process unifies the label linkage information in the linkage information buffer, and generates a look-up table that determines how temporary and final labels correspond. The linked data combination algorithm generates a chain label information table that represents the connectivity chain of temporary labels based on the label linkage information before the look-up table is generated.



Case 1: When the upper-left value of the window is 0, 0 is written to the frame memory (FM).

Case 2: When the upper-left value of the window is 1, R is written to the frame memory (FM).

Figure A-2: Definition of Primary Labeling Operation

The algorithm consists of the three steps shown in Figure A-3: clear, chain, and table generation. Assume the label table is a one-dimensional memory. In the clear and chain steps, the label table works as a chain label information table; in the table generation step, it works as a look-up table. In the chain label information table the notation "(m) = n" means that n represents the contents of the label table denoted by address m, in which m must be greater than n (every temporary label m has chain information with a smaller-numbered temporary label n). When n = 0, temporary label m has no chain information with smaller temporary labels.

The clear step initializes the label chain table by clearing its contents (i.e., any temporary label has no chain information with smaller-numbered temporary labels).

The chain step then generates a chain label information table in which each temporary label has connectivity with only one smaller-numbered temporary label. This table is generated by reading label linkage information (label A_0 connects with label B_0) from the label linkage buffer, and searching for connectivity for each A_0 and B_0 in the chain label information table in the direction of smaller-numbered temporary labels. Let A_i and B_j be the smallest number of temporary labels that connect with A_0 and B_0, respectively. $(A_i) = B_j$ is written in the chain label information table when $A_i > B_j$, and $(B_i) = A_j$ is written in the chain label information table when $A_i < B_j$. As a result, temporary labels A_0 and B_0 are connected with each other on the chain label information table through A_i and B_j connectivity. By applying these operations to all label linkage information in the linkage information buffer, every temporary label either has connectivity with only one smaller-numbered temporary label, or has no connectivity with smaller-numbered temporary labels.

The table generation step generates the look-up table. In this step, the contents of the chain label information table are modified and arranged sequentially from the smallest address number. When the contents of the chain label information table are 0 (temporary label has no connectivity with a smaller-numbered temporary label), a new label is assigned. When the contents of the chain label information table are non-zero (temporary label has connectivity with a smaller-numbered temporary label—already modified to be a final label), this label is assigned. By applying these operations to the contents of the chain label information table sequentially from the smallest address number, the look-up table is generated from the label table.

Since the algorithm requires one dimensional memory and consists of simple memory access and data comparison, this algorithm is suitable for hardware implementation.

(Step 1) Clear
(1) k <-- 0
(2) (k) <-- 0
(3) k <-- k+1
(4) if k < LC then go to (2)
(5) end

where (k) represents contents of label table denoted by address k, and LC is the final number of temporary label.

(Step 2) Chain
(1) k <-- 0
(2) m, n <-- LIB(k)
(3) if (m) ≠ 0 then m <-- (m) and go to (3)
(4) if (n) ≠ 0 then n <-- (n) and go to (4)
(5) if m < n then (n) <-- m
else if n < m then (m) <-- n
(6) k <-- k + 1
(7) if k<LIC then go to (2)
(8) end

where LIB(k) represents the two labels linkage information stored in the linkage information buffer addressed by k, and LIC is the number of the label linkage information.

(Step 3) Table generation
(1) c <-- 1 and k <-- 1
(2) if (k) = 0 then (k) <-- c and c <-- c + 1
else (k) <-- (k)
(3) k <-- k+1
(4) if k < LC then go to (2)
(5) end

Figure A-3: Labeling Unification Algorithm

A.3 Hardware Architecture

The labeling system consists of a labeling LSI, one-line delay, frame memory, linkage information buffer, and label table, as shown in Figure A-4. The labeling LSI consists of three main modules (primary labeling, labeling unification, and secondary labeling), and executes these processes sequentially. First, the primary labeling module generates temporary labels and label linkage information from binary image data input and one-line delayed data input. The generated temporary labels and label linkage information are stored in the frame memory and the linkage information buffer, respectively. Next, the linked data combination module generates the chain label information table from the label table based on the label linkage information in the linkage information buffer. After the chain label information table is generated, it is modified to obtain a look-up table. Finally, the secondary labeling module generates final labels by referring to the look-up table and relabeling the temporary labels.

The primary labeling algorithm is implemented using hardware shown in the block diagram in Figure A-5. The labeling LSI executes primary labeling with four-stage pipeline processing and contains the run label buffer, thus providing high-speed hardware performance. The linked data combination algorithm is also implemented in the hardware, so efficient and high-speed linked data combination can be achieved by designing for managing memory access and data comparison.

The main features of the labeling LSI are described below:

- All labeling processes are executed in a single chip.
- The maximum input image data rate is 40 MHz, and the entire labeling process is executed in three frames.
- The maximum number of generated temporary labels is 4094.
- The maximum number of generated label linkage information entries is 4095.
- 4-point and 8-point connectivity are available.

By using the three labeling systems in parallel, real-time video rate labeling can be achieved.

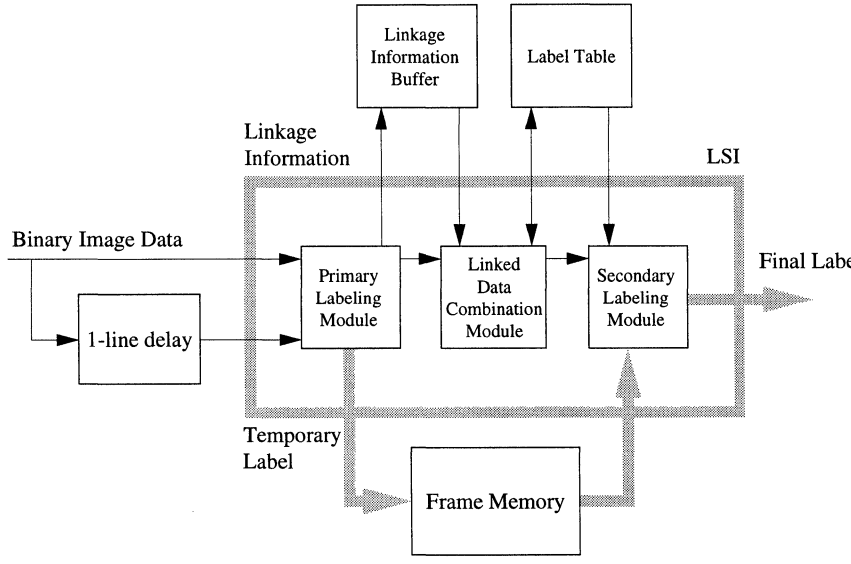
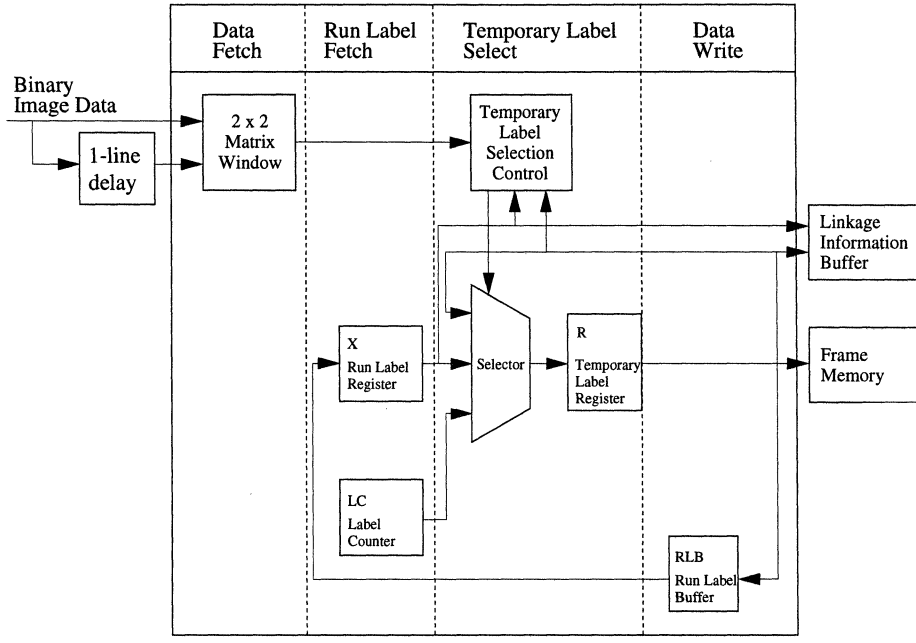
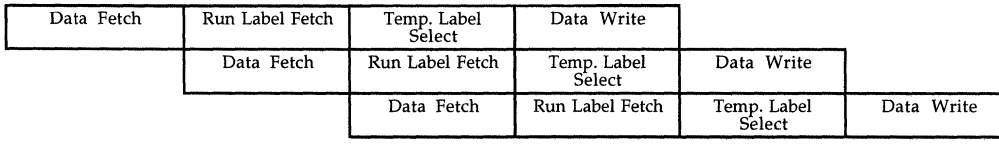


Figure A-4: Block Diagram of the LSI for High-Speed Image Labeling



(a) Block Diagram



(b) Pipeline Processing

Figure A-5: Architecture for Primary Labeling

A.4 Conclusion

Sumitomo has developed an LSI for high-speed image labeling, and a new primary labeling algorithm suitable for hardware implementation. The labeling LSI contains the entire labeling operation on a single chip that runs at 40 MHz. It has many applications, particularly in real-time video rate labeling processing.

Acknowledgment

The authors would like to thank Prof. T Hattori of Kagawa University for his helpful suggestions, and J. Watanabe and H. Tsuge for their encouragement.

References

[1] T. Hattori, "New Regional Labeling Algorithm of Pipeline Image Processor and Proof for the Correctness." in proc. IECON'91, pp. 2005–2030, 1991.

[2] H. Fujiwara, F. Isogsi. "A Method for High-performance labeling processor,"(in Japanese) Computer Vision, No, 56, pp. 1–7, 1988.

[3] Y. Shirai, K. Sugihara, "Algorithms and Hardwares for Labeling Connected Regions," (in Japanese) IECE Tech. Report IE78-9, pp. 29-33, 1978.

[4] T. Goto. Y. Ohta, M. Yoshida and Y. Shirai, "High speed labeling for component labeling," (in Japanese) Tran IEICE D-IT vol. 372-D-II, No. 2, pp. 247-255, 1989.

SMI ASSP

Image Processing LSI Series

IP90C11

Labeling Accelerator (LABop1K)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.5

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Overview

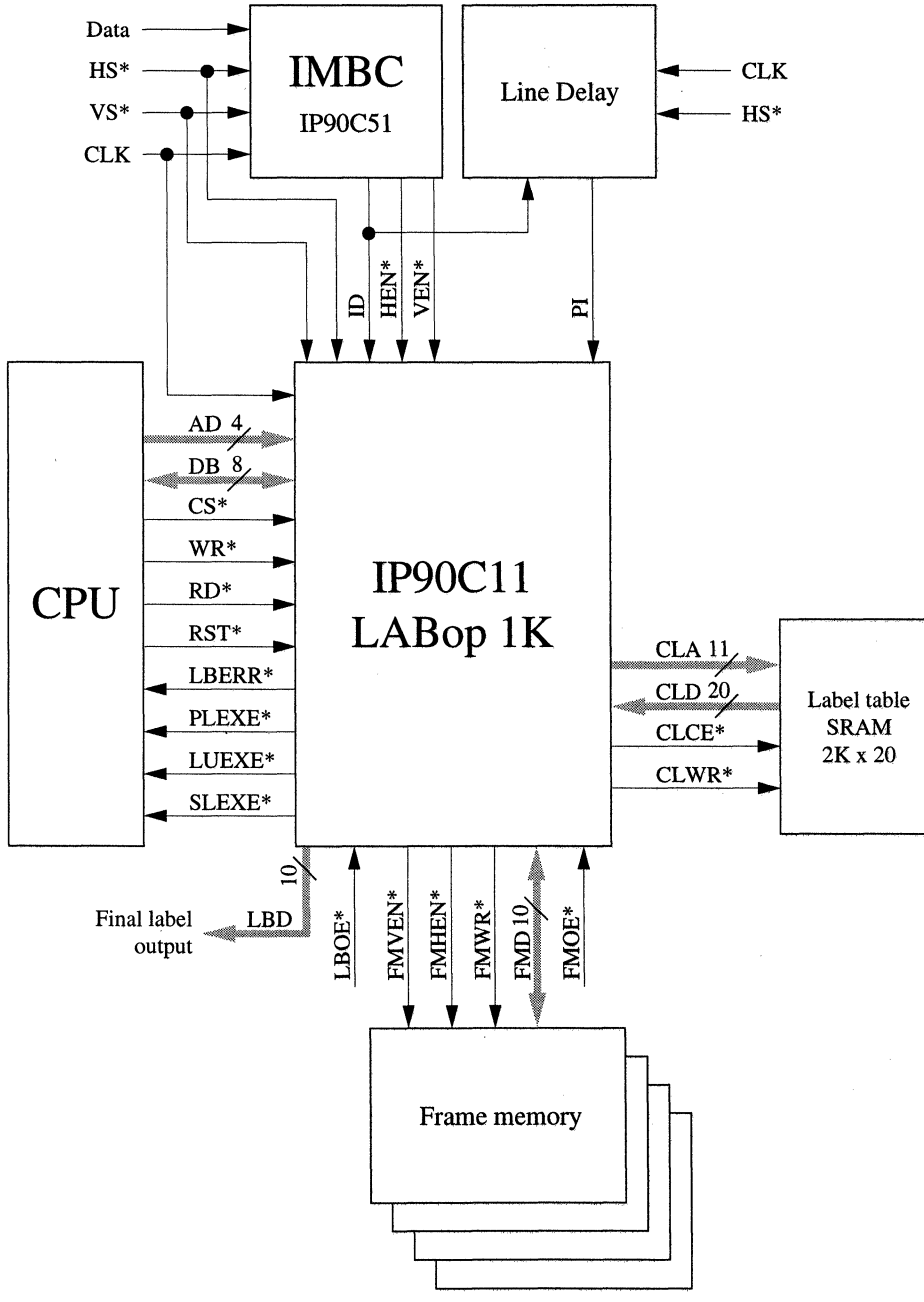
1.1 Product Overview

The IP90C11 (LABop1K) is a single-chip, image-link component labeling processor that uses a proprietary algorithm and high-speed circuit technology. It can process up to 1,022 temporary labels using either a 4-connectedness or an 8-connectedness labeling operation. Labeling can be processed in real-time using three chips in parallel.

1.2 Features

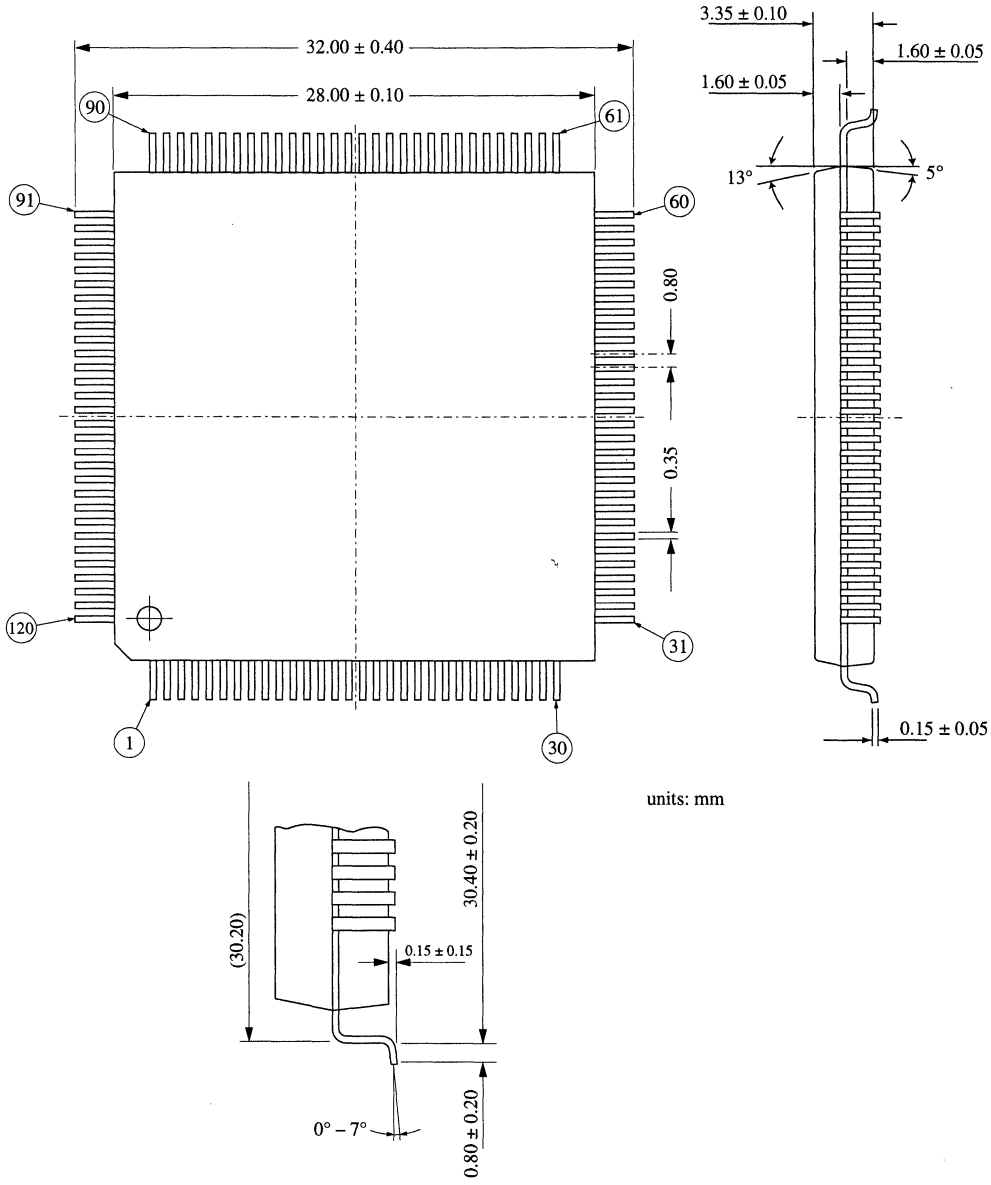
- Labeling processing functions
  - Labeling processing of non-interlaced raster-scanned binary input data
  - Maximum number of labels:
    - — Maximum number of temporary labels: 1,022
    - — Maximum number of label linkage information entries: 1,023
  - 4-connectedness/8-connectedness labeling
  - Wide range of image input. The IP90C11 processes up to 512 run labels in a horizontal line. (This does not represent the maximum number of pixels in the horizontal direction. See Section 4.4,"Algorithm," for details.)
  - Practical real-time labeling. Labeling can be processed in real time using three chips in parallel.
  - Automatic labeling function without CPU assistance. The IP90C11 executes a series of processes automatically:
    1. Primary labeling
    2. Linked data combination
    3. Secondary labeling
- External interface
  - Labeling information read out:
    - — Number of temporary labels
    - — Number of label linkage information entries
    - — Number of final labels
  - Built-in timer counter for label linkage information processing
  - Status output to CPU:
    - — Execution status for primary labeling, linked data combination, and secondary labeling
    - — Interrupt generation for overflow (e.g., number of temporary labels) or time-out in each stage of processing (interrupts can be masked as necessary)
  - General-purpose interface using 8-bit data bus
- Maximum operation frequency: f_{max} = 20 MHz
- Power supply: +5V single power supply
- Input/output: TTL-level compatible
- Package: 120-pin QFP

1.3 Block Diagram



Section 2: Pin Descriptions

2.1 Package Dimensions



2.2 Pin Descriptions

Pin group	Symbol	No. of Pins	I/O	Description
Image data input	ID	1	I	Image data input
	HEN*	1	I	Horizontal input image enable signal
	VEN*	1	I	Vertical input enable signal
	HS*	1	I	Horizontal synchronization signal
	VS*	1	I	Vertical synchronization signal
	PI	1	I	1-line delayed image data input
	CLK	1	I	Clock
Label table interface	CLD19–CLDO	20	I/O	Label table data bus
	CLA10–CLA0	11	O	Label table address bus
	CLCE*	1	O	Label table chip enable
	CLWR	1	O	Label table write enable
Frame memory interface	FMD9–FMD0	10	I/O	Temporary label data bus
	FMOE*	1	I	Temporary label output enable
	FMWR*	1	O	Frame memory write enable
	FMVEN*	1	O	Frame memory vertical enable signal
	FMHEN*	1	O	Frame memory horizontal enable signal
Final label output	LBD9–LBD0	10	O	Final label data bus
	LBOE*	1	I	Final label output enable
Status	LBERR*	1	O	Labeling errors
	PLEXE*	1	O	Primary labeling execution frame status
	LUEXE*	1	O	Linked data combination processing execution status
	SLEXE*	1	O	Secondary labeling execution frame status
CPU interface	AD3–AD0	4	I	Address bus
	DB7–DB0	8	I/O	Data bus
	CS*	1	I	Chip select
	WR*	1	I	Write enable
	RD*	1	I	Read enable
	RST*	1	I	System reset (Note 1)
Test	TSTEN0	1	I	Test signal, normally low (Note 2)
	TSTEN1	1	I	Test signal, normally low (Note 2)
Open		4		No connect (Note 3)
Pull down		6		(Note 4)
Power supply	Vdd	8	PW	5V
	GND	15	PW	Ground
Total pin count		120		

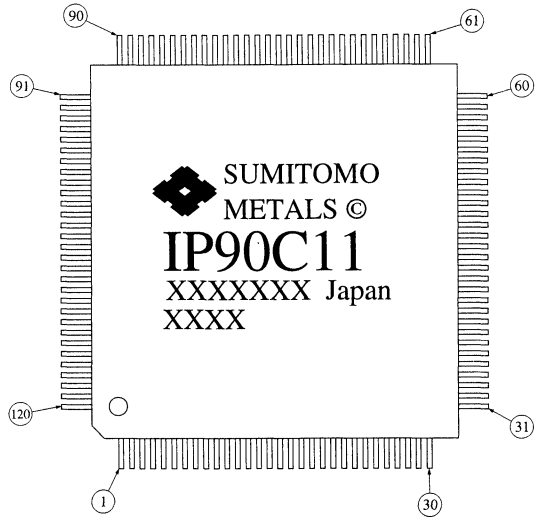
Note 1: RESET must stay low for at least three clock cycles. This pin is provided with a pull-up resistor.

Note 2: The test pins TSTEN0 and TSTEN1 are used to test the LSI's internal logic, and must normally be held low. Note that these pins are provided with pull-down resistors.

Note 3: These pins must not be connected to any signal or power supply.

Note 4: These pins must be connected to GND through a 10-KΩ pull-down resistor.

2.3 Pin Configuration



Pin No.	Name	Type
1	GND	PW
2	AD0	I
3	AD1	I
4	AD2	I
5	AD3	I
6	GND	PW
7	DB0	I/O
8	DB1	I/O
9	DB2	I/O
10	DB3	I/O
11	DB4	I/O
12	DB5	I/O
13	DB6	I/O
14	DB7	I/O
15	Vdd	PW
16	GND	PW
17	LBERR*	O
18	PLEXE*	O
19	LUEXE*	O
20	SLEXE*	O
21	LBD0	O
22	LBD1	O
23	GND	PW
24	LBD2	O
25	LBD3	O
26	LBD4	O
27	LBD5	O
28	LBD6	O
29	LBD7	O
30	Vdd	PW

Pin No.	Name	Type
31	LBOE*	I
32	GND	PW
33	LBD8	O
34	LBD9	O
35	open	Note 1
36	open	Note 1
37	FMHEN*	O
38	FMVEN*	O
39	FMWR*	O
40	GND	PW
41	FMD0	I/O
42	FMD1	I/O
43	FMD2	I/O
44	FMD3	I/O
45	GND	PW
46	FMOE*	I
47	Vdd	PW
48	FMD4	I/O
49	FMD5	I/O
50	FMD6	I/O
51	FMD7	I/O
52	FME8	I/O
53	FMD9	I/O
54	pull down	Note 2
55	pull down	Note 2
56	GND	PW
57	CLD0	I/O
58	CLD1	I/O
59	CLD2	I/O
60	Vdd	PW

Pin No.	Name	Type
61	GND	PW
62	CLD3	I/O
63	CLD4	I/O
64	CLD5	I/O
65	CLD6	I/O
66	CLD7	I/O
67	CLD8	I/O
68	GND	PW
69	CLD9	I/O
70	pull down	Note 2
71	pull down	Note 2
72	CLD10	I/O
73	CLD11	I/O
74	CLD12	I/O
75	GND	PW
76	Vdd	PW
77	CLD13	I/O
78	CLD14	I/O
79	CLD15	I/O
80	CLD16	I/O
81	CLD17	I/O
82	GND	PW
83	CLD18	I/O
84	CLD19	I/O
85	pull down	Note 2
86	pull down	Note 2
87	GND	I/O
88	CLCE*	O
89	CLWR*	O
90	Vdd	PW

Pin No.	Name	Type
91	CLA0	O
92	CLA1	O
93	CLA2	O
94	CLA3	O
95	CLA4	O
96	CLA5	O
97	GND	PW
98	CLA6	O
99	CLA7	O
100	CLA8	O
101	CLA9	O
102	open	Note 1
103	open	Note 1
104	CLA10	O
105	Vdd	PW
106	CLK	I
107	GND	PW
108	ID	I
109	PE	I
110	HEN*	I
111	VEN*	I
112	HS*	I
113	VS*	I
114	RD*	I
115	RST*	I
116	WR*	I
117	CS*	I
118	TSTEN0	I
119	TSTEN1	I
120	Vdd	PW

Note 1: No connect. These pins must not be connected to any signal or power supply.

Note 2: These pins must be connected to GND through a 10-KΩ pull-down resistor.

Section 3: Registers

3.1 Register List

Address (hex)	Symbol		Read/Write	Description
0	CMD		Read/Write	Control command register
1	STAT		Read only	Execution status register
2	ERRM		Read/Write	Error interrupt mask register
3	ERRS		Read/Write	Error status register
4	TIME	Low	Read/Write	Linked data combination processing timer-count setup register
5		High		
6	Reserved		—	Reserved for system use; do not use
7	Reserved		—	Reserved for system use; do not use
8	TLBN	Low	Read only	Temporary label count register
9		High		
A	RLBN	Low	Read only	Final count register
B		High		
C	ERLN	Low	Read only	Error line register
D		High		
E	CMBN	Low	Read only	Linked data combination information count register
F		High		

3.2 Register Configuration

Control Command Register CMD

Address 0h — Read/Write
Hardware reset: — 00h
Software reset: — 0 except the RESET bit

This register provides the commands necessary to control the LABop1K.

7	6	5	4	3	2	1	0
4/8D	RESET	LEVEL	IRM	EXE4	EXE3	EXE2	EXE1

4/8D: Selects 4-connectedness (when set to 1) or 8-connectedness (when set to 0) processing.

RESET: Performs a reset. When set to 1, the LABop1K resets the entire register to 0, except the RESET bit. To clear the reset, write 0 to this bit. When setting the control command register after a reset from the RESET bit, first clear the RESET bit to 0, then write the desired value.

LEVEL: Sets a frame-memory control signal. When set to 0, the LABop1K selects one clock-pulsed output of the FMVEN* and FMHEN* signals, which indicate the frame-memory-enable period. When pulsed output is selected, the output signal is accompanied by a single pulse immediately preceding and another immediately following the enable period.

When set to 1, the LABop1K selects level output of FMVEN* and FMHEN*, which holds output low during the enable period.

(For details, see Section 5, "System Timing." Pulsed signal waveforms are indicated by dotted lines.)

IRM: Disables interrupts. When set to 0, the LABop1K prevents LBERR* from going low regardless of which type of error occurs.

EXE4: Sets continuous execution of secondary labeling. When the EXE4 and EXE3 bits are set to 1, and EXE2, EXE1, and the STAT register's STAT2 bit are also 0, the LABop1K starts secondary labeling when the next VS* signal is asserted. The LABop1K continues executing secondary labeling until EXE4 is cleared to 0.

EXE3: Sets the execution of secondary labeling. When set to 1, and EXE2, EXE1, and the STAT register's STAT2 bit are 0, the LABop1K starts secondary labeling when the next VS* is asserted. EXE3 is reset to 0 immediately after secondary labeling starts.

EXE2: Sets the execution of label linkage information processing. When set to 1 while primary labeling is not being executed, the LABop1K immediately starts label linkage information processing. EXE2 is reset to 0 immediately after label linkage information processing starts.

EXE1: Sets the execution primary labeling. When set to 1, the LABop1K starts primary labeling when the next VS* is asserted. EXE1 is reset to 0 immediately after primary labeling starts.

Execution Status Register STAT

Address 1h Read-only

Reset: XXXXX000h

This register reads the LABop1K status.

7	6	5	4	3	2	1	0
X	X	X	X	X	STAT3	STAT2	STAT1

STAT3: Indicates secondary labeling status. When set to 1, the frame is a secondary labeling execution frame.

STAT2: Indicates label linkage information processing status. When set to 1, the LABop1K is executing label linkage information processing.

STAT1: Indicates primary labeling execution status. When set to 1, the frame is a primary labeling execution frame.

X: Reserved. (This bit is undefined when read by the CPU.)

Error Interrupt Mask Register ERRM

Address 2h Read/Write

Reset: 00h

This register masks interrupt requests caused by errors. Setting a bit to 0 prevents the LBERR* pin from going low even when an error occurs. Interrupts can also be disabled by setting the CMD register's IRM bit to 0. This prevents the LBERR* pin from going low regardless of what type of error occurs.

7	6	5	4	3	2	1	0
RLOFM	TMOVM	0	CMOFM2	CMOFM1	0	TLOFM2	TLOFM1

RLOFM: Masks a run-label overflow error. The LABop1K memory stores up to 512 run labels per line (see Section 4.4, "Algorithm"). If set to 0, LBERR* is prevented from going low even when the number of run labels per line exceeds 512.

TMOVM: Masks a linked data combination processing overtime error. If set to 0, LBERR* cannot go low even when the label linkage information process exceeds the preset time (see the TIME register).

CMOFM2: Masks label linkage information overflow error 2. If set to 0, LBERR* cannot go low even when the number of label linkage information entries exceeds 1,023 (register CMBN value = 3FFh).

CMOFM1: Masks label linkage information overflow error 1. If set to 0, LBERR* cannot go low even when the number of label linkage information entries exceeds 255 (register CMBN value = 0FFh).

TLOFM2: Masks temporary labels overflow error 2. If set to 0, LBERR* cannot go low even when the number of temporary labels exceeds 1,022 (register TLBN value = 3FFh).

TLOFM1: Masks temporary labels overflow error 1. If set to 0, LBERR* cannot go low even when the number of temporary labels exceeds 254 (register TLBN value = 0FFh).

NOTE: Bit 5 and Bit 2 of the ERRM must be set to 0.

Error Status Register ERRS

Address 3h Read/Write

Reset: 00h

This register reads the status of errors occurring during labeling. If an error occurs during labeling, the bit corresponding to the error is set to 1, and remains at 1 unless the CPU writes 0 to it. When a bit in the ERRS register and the CMD register's IRM bit are set to 1, and the ERRM register bit corresponding to that error is 1, LBERR* is pulled low, and the LABop1K stops processing. (See Section 4.3, "Error Handling Sequence," for details.)

7	6	5	4	3	2	1	0
RLOFS	TMOVS	X	CMOFS2	CMOFS1	X	TLOFS2	TLOFS1

RLOFS: Indicates run-label overflow status. Set to 1 when the number of run labels per line exceeds 512.

TMOVS: Indicates linked data combination processing overtime status. Set to 1 when the time required for linked data combination processing exceeds the preset time (see the TIME register).

CMOFS2: Indicates label linkage information overflow status 2. Set to 1 when the number of label linkage information entries exceeds 1, 023 (register CMBN value = 3FFH).

CMOFS1: Indicates label linkage information overflow status 1. Set to 1 when the number of label linkage information entries exceeds 255 (register CMBN value = 0FFH).

TLOFS2: Indicates temporary labels overflow status 2. Set to 1 when the number of temporary labels exceeds 1,022 (register TLBN value = 3FFh).

TLOFS1: Indicates temporary labels overflow status 1. Set to 1 when the number of temporary labels exceeds 254 (register TLBN value = 0FFh).

X: Reserved. (This bit is undefined when read by the CPU.)

Linked Data Combination Processing Timer-count Setup Register TIME

Address 4h/5h Read/Write

Reset: 00h

The time required for label linkage information processing is proportional to the number of temporary labels or label linkage information entries involved. If the system processes labeling in real time, and noise or a similar problem causes the number of temporary labels or label linkage information entries to be excessively large, processing may need to be stopped. This register sets a time limit (in clock cycles) after which label linkage information processing is stopped.

After label linkage information processing starts, the time-out counter begins counting clock cycles. When the count exceeds the value set by the TIME register multiplied by 256, a time-out error is assumed and the ERRS register's TMOVS bit is set to 1.

	7	6	5	4	3	2	1	0
5h	TIME15	TIME14	TIME13	TIME12	TIME11	TIME10	TIME9	TIME8
4h	TIME7	TIME6	TIME5	TIME4	TIME3	TIME2	TIME1	TIME0

TIME0–15: Sets the time to discontinue label linkage information processing.

Temporary Label Count Register TLBN

Address 8h/9h Read-only

Reset: X000h

This register reads the number of temporary labels. (The value in this register exceeds the generated temporary labels by one.) This register resets each time the LABop1K starts primary labeling, then counts the temporary labels.

	7	6	5	4	3	2	1	0
9h	X	X	X	X	X	X	TLBN9	TLBN8
8h	TLBN7	TLBN6	TLBN5	TLBN4	TLBN3	TBN2	TLBN1	TLBN0

TLBN0-9: Indicates the number of temporary labels.

X: Reserved. (This bit is undefined when read by the CPU.)

Final Label Count Register RLBN

Address Ah/Bh — Read-only

Reset: — bit 0 to bit 9 = 0

This register reads the number of final labels. (The value in this register exceeds the generated final labels by one.) This register resets each time the LABop1K starts primary labeling, then counts the final labels.

	7	6	5	4	3	2	1	0
Bh	X	X	X	X	X	X	RLBN9	RLBN8
Ah	RLBN7	RLBN6	RLBN5	RLBN4	RLBN3	RLBN2	RLBN1	RLBN0

RLBN0–9: Indicates the number of final labels.

X: Reserved. (This bit is undefined when read by the CPU.)

Error Line Register ERLN

Address Ch/Dh — Read-only

Reset: — X000h

This register reads the number of horizontal lines when an error occurs. This register resets each time the LABop1K starts primary labeling and then counts the horizontal lines.

	7	6	5	4	3	2	1	0
Dh	X	X	X	X	ERLN11	ERLN10	ERLN9	ERLN8
Ch	ERLN7	ERLN6	ERLN5	ERLN4	ERNL3	ERLN2	ERLN1	ERLN0

ERLN0–11: Indicates the value of the vertical address at the time an error occurs.

X: Reserved. (This bit is undefined when read by the CPU.)

Linked Data Combination Count Register CMBN

Address Eh/Fh Read-only

Reset: bit 0 to bit 9 = 0

This register reads the number of label linkage information entries. This register is reset each time the LABop1K starts primary labeling, then counts the label linkage information entries.

	7	6	5	4	3	2	1	0
Fh	X	X	X	X	X	X	CMBN9	CMBN8
Eh	CMBN7	CMBN6	CMBN5	CMBN4	CMBN3	CMBN2	CMBN1	CMBN0

CMBN0–9: Indicates the number of label linkage information entries.

X: Reserved. (This bit is undefined when read by the CPU.)

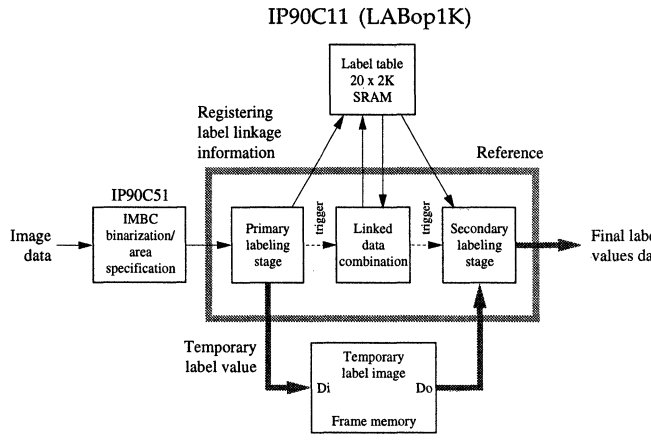
Section 4: Functional Description

4.1 Description

The LABop1K labels raster-scanned binary image data, and includes these three functions in a single chip:

1. Primary labeling
2. Linked data combination
3. Secondary labeling

The LABop1K provides individual command and status bits for these three functions. This allows each function to execute independently, and to run automatically and continuously by simultaneously applying the executing commands.



Internal Processing Block Diagram

4.2 Device Operation

Primary Labeling Processing

The LABop1K performs primary labeling on raster-scanned binary image data. When the CPU sets the CMD register EXE1 bit to 1, the LABop1K enters primary labeling mode when the next VS* signal is asserted.

When in primary labeling mode, the LABop1K receives image data in the effective area defined by the VEN* and HEN* signals, and simultaneously outputs temporary labels to the frame memory and label linkage information to the label table. After starting processing, the LABop1K terminates primary labeling mode when the next VS* is asserted. The effective area must contain at least two lines (i.e., there must be two HEN* periods during the VEN* period). Also, an additional HEN* period is required to process the last line of the effective area. The LABop1K operation is not affected by disabling VEN*. For details, see Section 5, "System Timing."

Note that the LABop1K minimizes the number of temporary labels required, by using a proprietary algorithm for handling large images. (For details, see Section 4.4, "Algorithm.")

The LABop outputs the value 0 (background value) for all input data that falls outside the effective area, regardless of the input data value.

Linked Data Combination Processing

The LABop1K enters label linkage information processing mode when the CMD register's EXE2 bit is set to 1. If this occurs when the EXE1 bit is 1 or primary labeling is under way, the LABop1K enters label linkage information processing mode only after primary labeling processing of the effective area is done. Otherwise, the LABop1K enters the mode immediately.

In this mode, the LABop1K processes the label linkage informationin the label table. This creates a conversion table (look-up table) used to convert between final and temporary labels.

The LABop1K ends label linkage information processing mode after all processing is done.

Secondary Labeling Processing

The LABop1K enters secondary labeling processing mode when the CMD register's EXE3 bit is set to 1. If this occurs when label linkage information processing is underway, secondary labeling processing mode is entered only after the current processing is completed and the next VS* is asserted. Otherwise, the mode is entered immediately after the next VS* is asserted.

In this mode, the LABop1K executes secondary labeling for the temporary label data in the frame memory by referring to the look-up table in the label table.

The LABop1K ends this mode when the next VS* is asserted. Note that if the EXE4 bit is set to 1, the LABop1K continues processing regardless of whether VS* is asserted. If the EXE4 bit is set to 0, the LABop1K ends secondary labeling processing mode when the next VS* is asserted.

Error Handling

If an error occurs while the CMD register's IRM bit and the error mask bit corresponding to the error are both set to 1, the LABop1K asserts the LBERR* signal and stops operation. Refer to the ERRS register to find the cause of the error. Restart the LABop1K by applying a system reset.

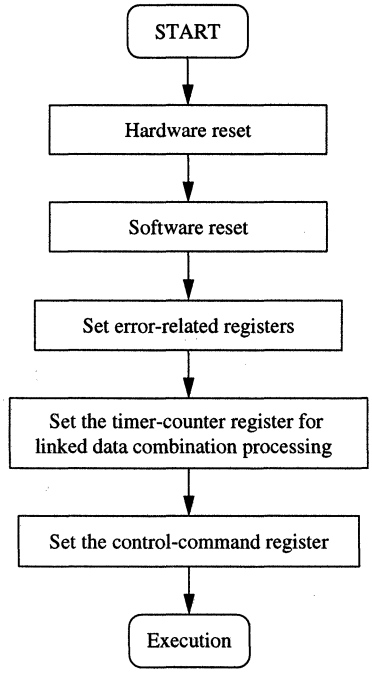
Reset

This operation initializes the LABop1K's registers and internal status. Bi-directional pins are placed in high-impedance state.

The final label output LBDs are set to output (FFFFh) when the LBOE* signal is low, and to high-impedance state when LBOE* is high. Status outputs are deasserted. During a reset, all inputs except RST* are invalid.

4.3 Control Sequence

Internal Register Write Sequence



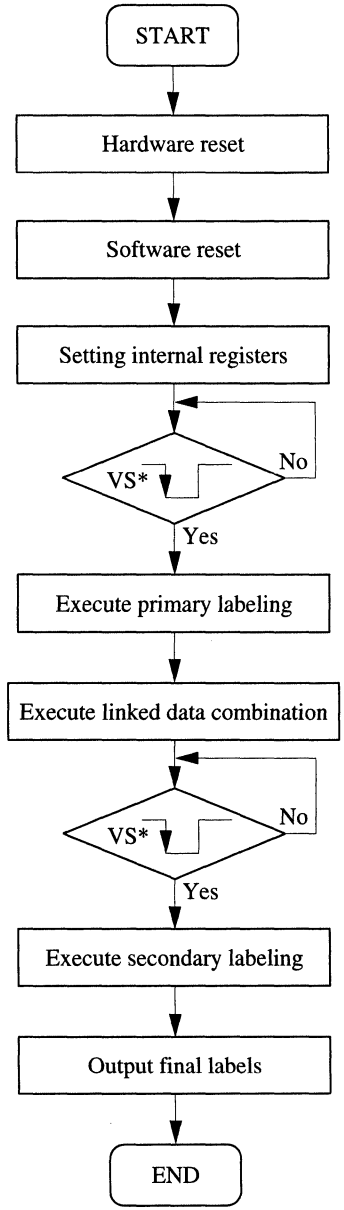
Set the ERRM and ERRS registers to their designated states.

Set the TIME register to the designated value.

Set the CMD register to its designated state.

If the CMD register is set to 07h, for example, a series of operations from primary labeling to secondary labeling is automatically executed using 8-connectedness labeling operation.

Labeling Execution Sequence



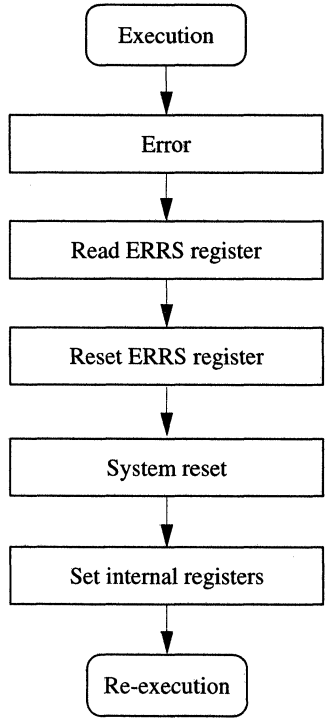
The LABop begins primary labeling when VS* is asserted, and outputs temporary label data to the frame memory and label linkage information to the label table.

Immediately after primary labeling finishes, the LABop begins linked data combination to create a look-up table (in the label table) that shows the relationship between temporary and final labels.

When linked data combination is complete, the LABop begins secondary labeling when the next VS* is asserted, and converts temporary labels to final labels.

Final labels are output.

Error-Handling Sequence



If an error occurs, the LABop asserts status output signal LBERR* by pulling it low, then stops processing.

Reading the ERRS register shows the cause of the error.

Clear the error by writing 0 to the ERRS register.

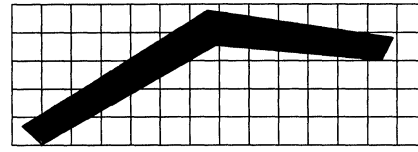
Reset the system using a hardware or software reset.

4.4 Algorithm (Primary Labeling Processing)

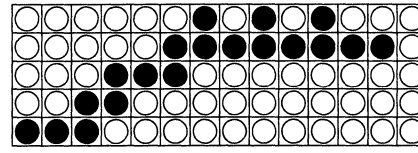
Assignment of Temporary Labels

The LABop1K minimizes the number of temporary labels required. For example, if the target image shown in (a) below is read into the system from a camera, the pixels in the data do not always form a smooth-edged line (due to jags or noise) as shown in (b). In such a case, a conventional algorithm generates varying temporary labels, as shown in (c). However, the LABop1K applies the single label shown in (d) by recognizing that the pixels of the object to be read from a camera are connected.

(a)
Target image



(b)
Entered data
(binary image)



(c)
Example of temporary labels by conventional simple raster-scan operation

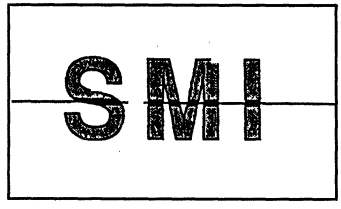
						1		2		3			
					1	1	1	1	1	1	1		
			4	1	1								
		4	4										
5	4	4											

(d)
Temporary labels by the LABop

						1		1		1			
					1	1	1	1	1	1	1		
			1	1	1								
		1	1										
1	1	1											

Run Label

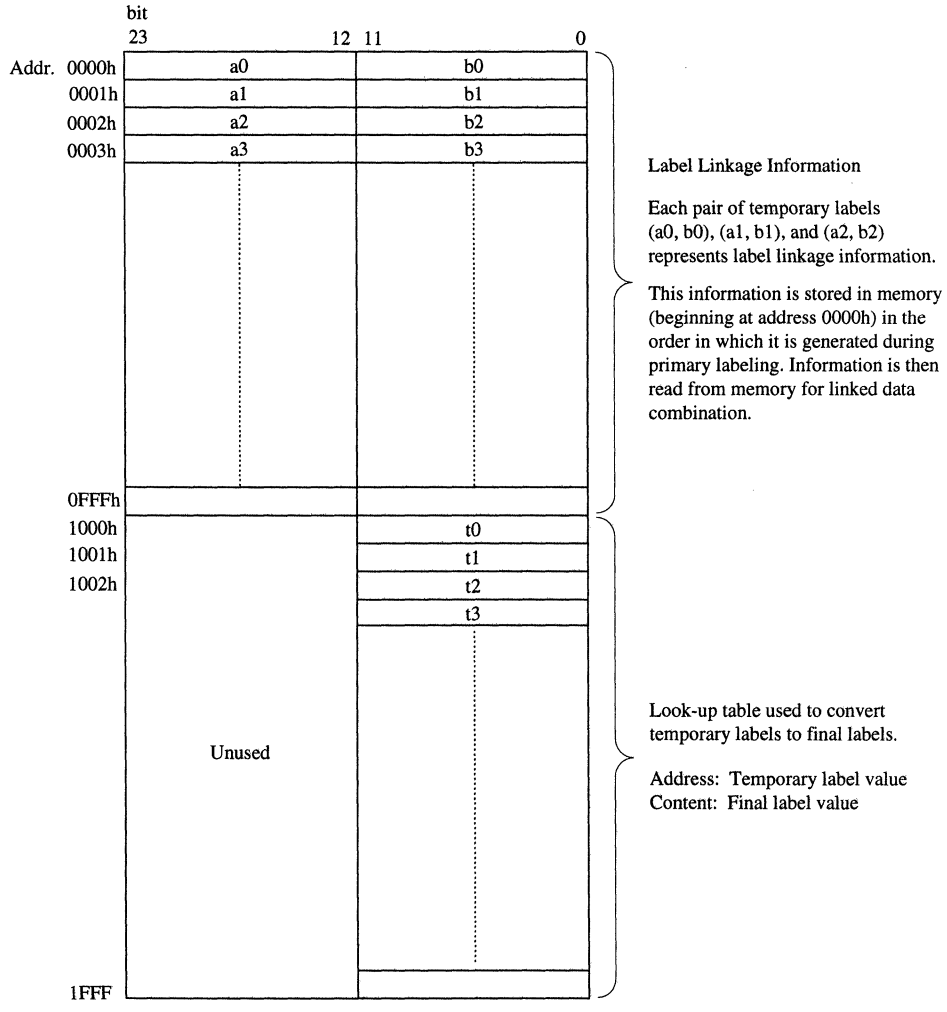
The LABop1K uses a "run label" to minimize the temporary labels during primary processing. A row of consecutive logical 1s of input data is called a "run," and the preliminary label information assigned to each run is called a "run label." The LABop1K incorporates 512 run label buffers, and so can process up to 512 runs per line.



The thick line portions of the line in the figure to the left indicate runs. The line contains six runs.

Therefore, in practical application, the maximum number of processed pixels in the horizontal directions is much greater than the worst case for alternating logical 1s and 0s, which is 512 x 2 = 1024 pixels.

4.5 Label Table Format

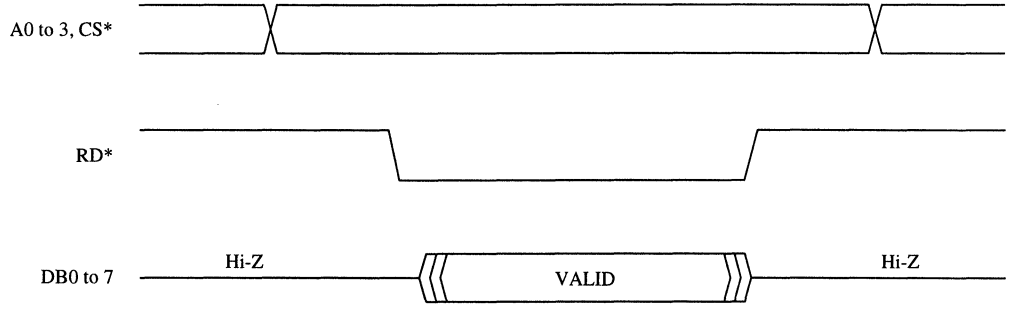


To store label linkage information, memory space equivalent to the number of label linkage information entries is allocated, beginning with address 000h. Memory space equivalent to the number of temporary labels is allocated in the look-up table, beginning with address 400h. The contents of the unused addresses in the label linkage information and look-up-table memory spaces are undefined.

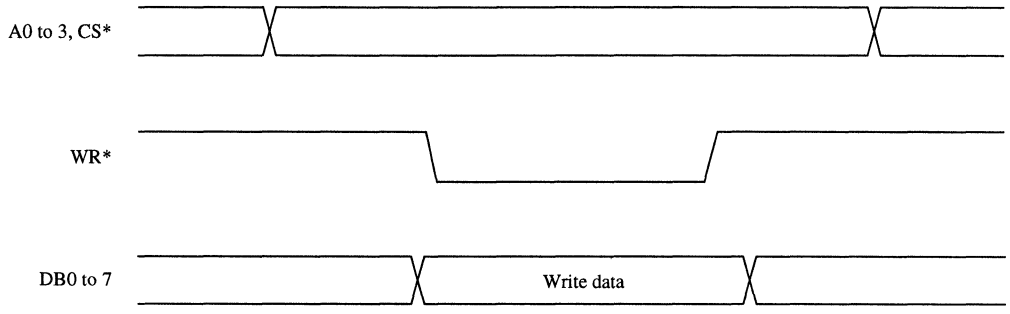
Section 5: System Timing

5.1 CPU Interface

Read Cycle Timing

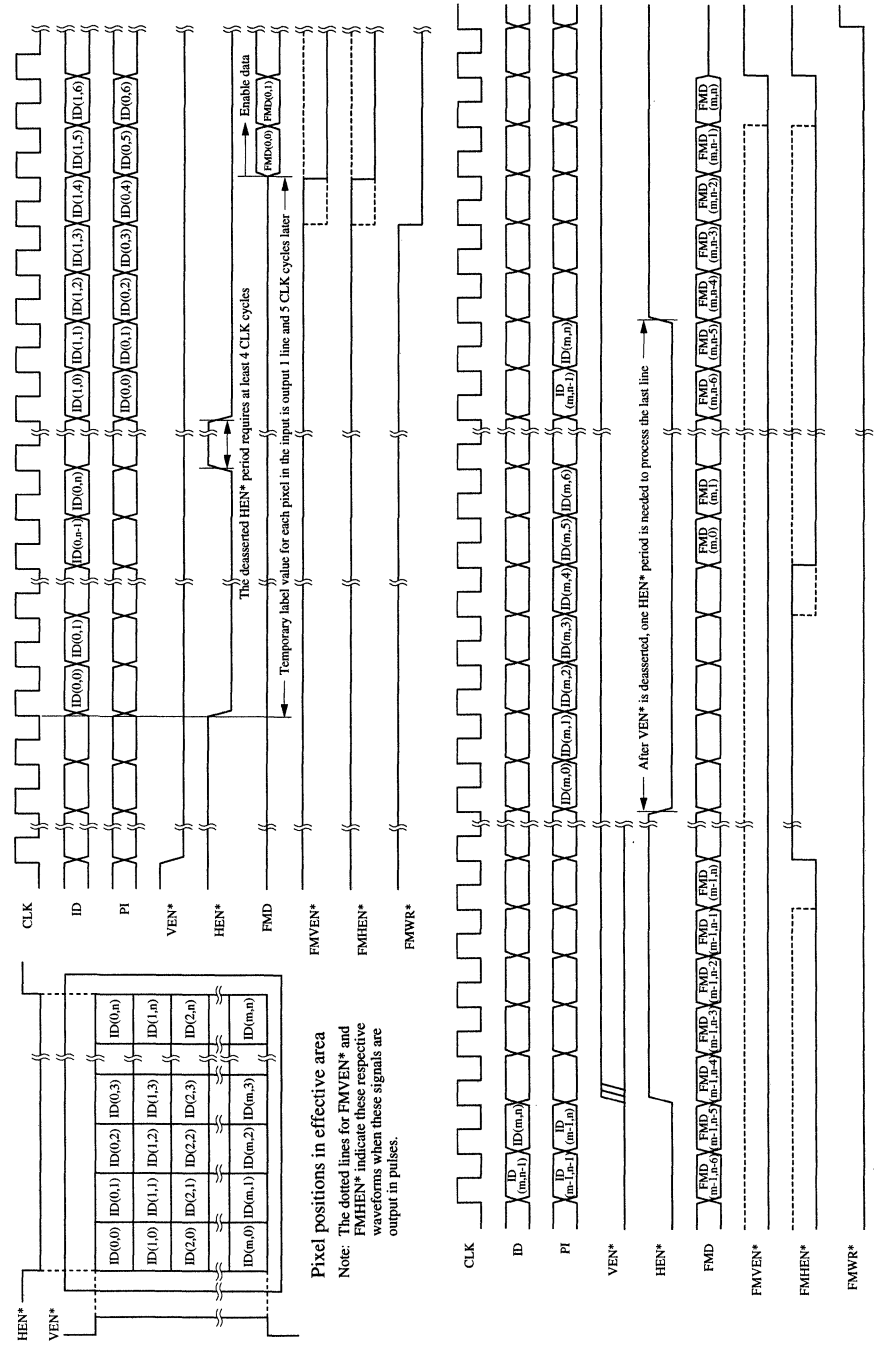


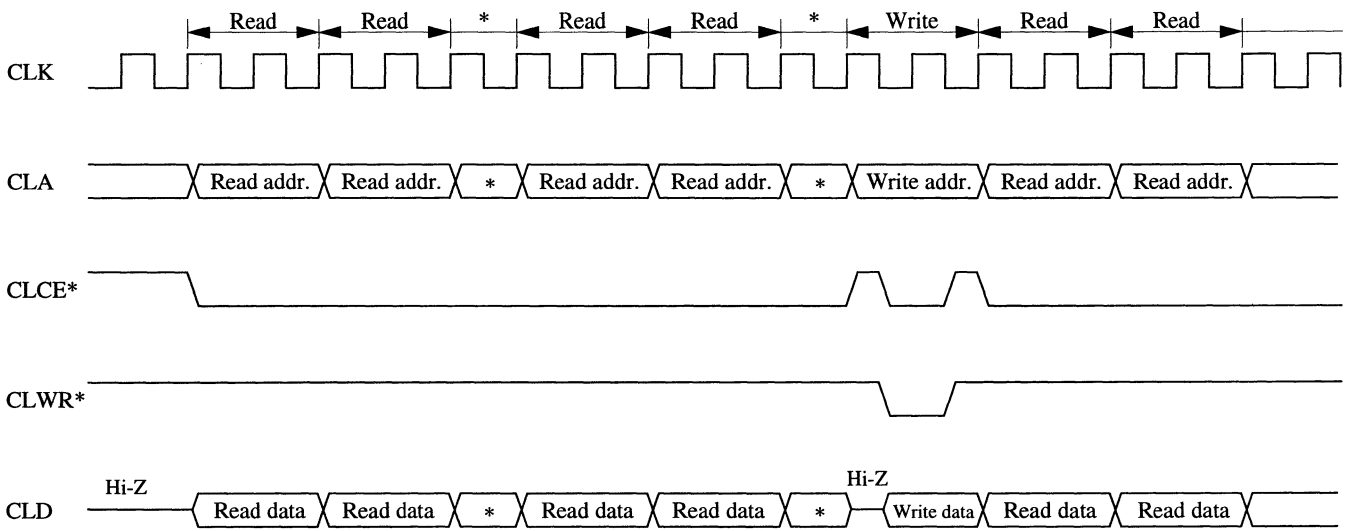
Write Cycle Timing



Note: The CPU interface signals are not synchronized with the image clock.

5.2 Primary Labeling Processing

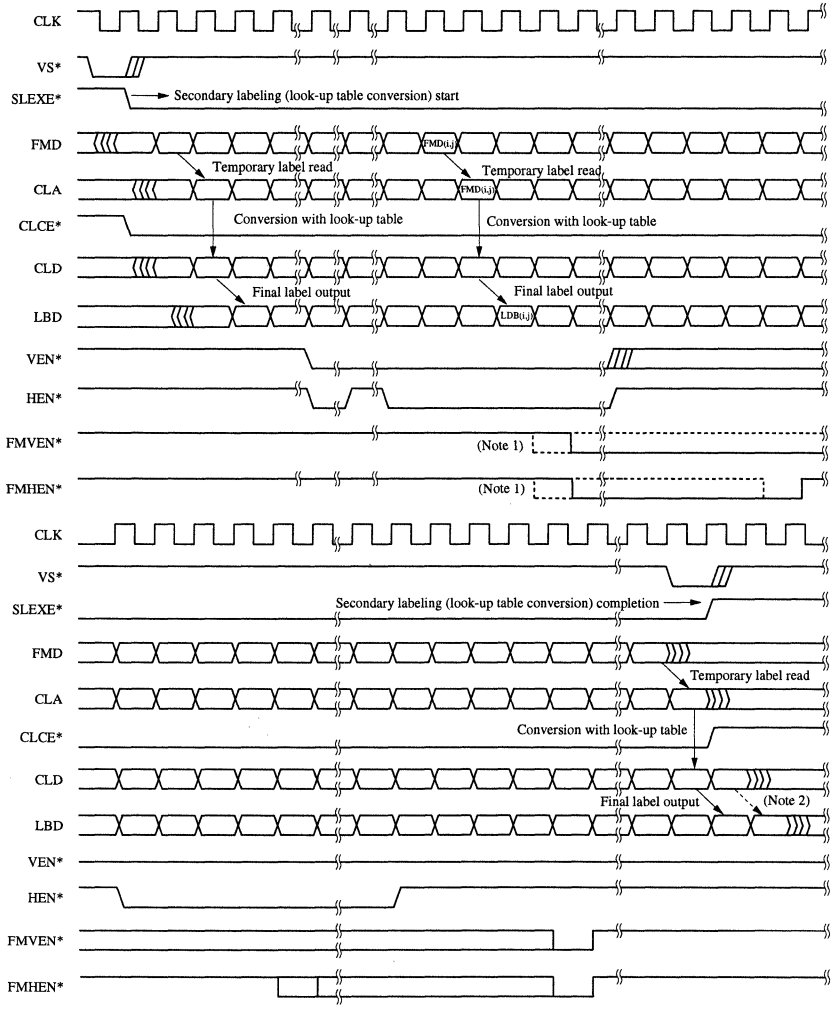




Note: Linked data combination requires two clock cycles each for the read and and the write cycles. A single dummy cycle, marked by *, can be inserted during a read operation. CLA and CLD are indeterminate during this dummy cycle.

5.4 Secondary Labeling Processing

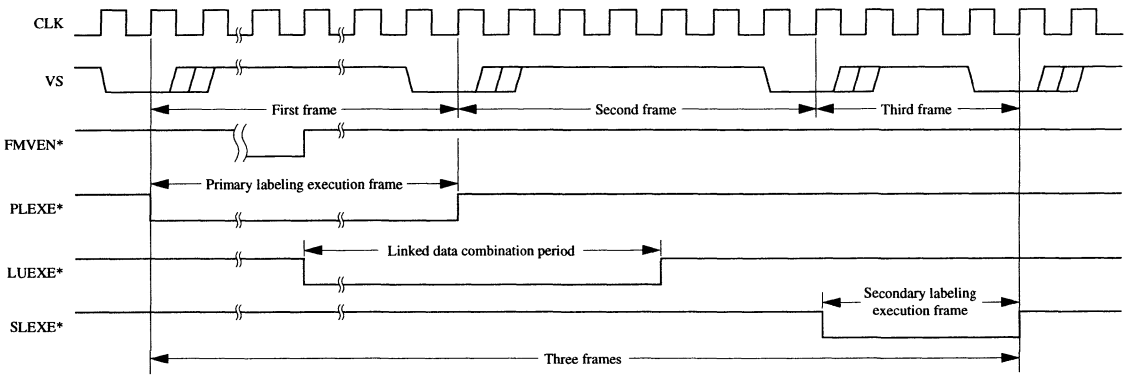
Temporary labels can be converted to final labels using the look-up table during the VS*-to-VS* interval. The LABop1K always latches the FMD value as long as SLEXE* is held low, and outputs it to CLA. At the same time, the LABop1K latches the CLD value and outputs it to LBD. While SLEXE* is held low, temporary labels can be entered into FMD, so system timing can be set as required. The FMVEN* and FMHEN* timing relative to VEN* and HEN* are the same as in primary labeling.



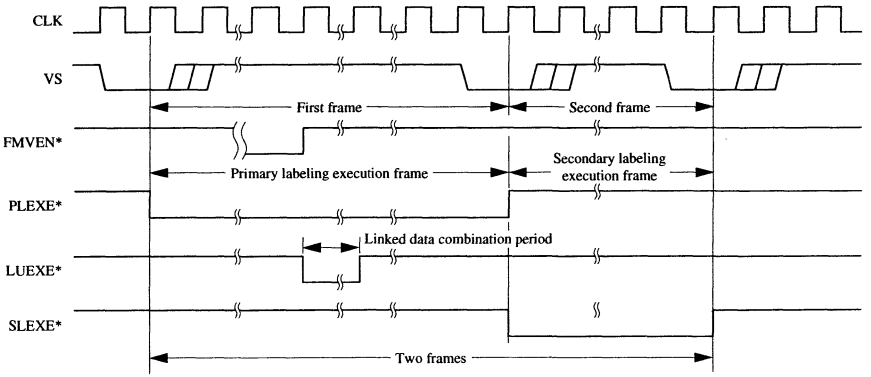
Note 1: The dotted lines for FMVEN* and FMHEN* indicate the respective waveforms when these signals are output in pulses.

Note 2: For high-speed systems that may require two clock cycles for converting temporary labels to final labels using the look-up table (with an external latch provided for CLA to extend SRAM access time), an additional operating cycle to latch CLD and output it to LBD is provided here.

a) Linked data combination processing is not completed within the primary labeling execution frame interval.



b) Linked data combination processing is completed within the primary labeling execution frame interval.



Description of status signals

PLEXE* is asserted and held low during the primary labeling execution frame interval.

LUEXE* is asserted immediately after primary labeling of the effective area is completed, and is de-asserted when linked data combination is complete.

SLEXE* is asserted and held low during the secondary labeling execution frame interval.

Section 6: Electrical Characteristics

6.1 Absolute Maximum Ratings

(Referenced to GND, Ta = 25°C unless otherwise specified)

Parameter	Symbol	Rating	Unit
DC supply voltage	Vdd	-0.3 to 6.5	V
Input/output voltage	VIN	-0.3 to Vdd + 0.3	V
Operating temperature	TOPT	0 to 70	°C
Storage temperature	TSTG	-10 to 80	°C

6.2 Recommended Operating Conditions (Ta = 0 to 70°C)

Parameter		Symbol	Condition	Min.	Typ.	Max.	Unit
Power-supply voltage		Vdd		4.75	5.0	5.25	V
Input voltage	Range	VI	TTL level normal input	0		Vdd	V
	High level	VIH		2.2		Vdd	V
	Low level	VIL		0		0.8	V
Input rising time		TRI	TTL level normal input	0		200	ns
Input falling time		TFI		0		200	ns

6.3 Input/Output Capacitance

Parameter	Symbol	Condition	Typ.	Unit
Input capacitance	CIN	Any input (Note 1)	10	pF
Output capacitance	COUT	Any output	10	pF

Note 1: Does not apply to bi-directional buffers.

6.4 DC Characteristics

Vdd = 5V ± 5%
Ta = 0 to 70°C
GND = 0V

Symbol	Parameter		Condition	Min.	Typ.	Max.	Unit
VIL	Low level input voltage					0.8	V
VIH	High level voltage			2.0			V
IIN	Input current	Without pull-up resistor Without pull-down resistor	VIN = Vdd or GND	-10	±1	10	µA
		With pull-down resistor	VIN = Vdd	35	110	335	µA
		With pull-up resistor	VIN = GND	-35	-115	-350	µA
VOH	High level output voltage (Note 1)		IOH = -4 mA	2.4	4.5		V
			IOH = -6 mA	2.4	4.5		V
VOL	Low level output voltage (Note 1)		IOL = 4 mA		0.2	0.4	V
			IOL = 6 mA		0.2	0.4	V
IOZ	Off-state leakage current		VOH = Vdd or GND	-10	±1	10	µA
IOS	Output short circuits current (Note 2)		Vdd=Max, Vo=Vdd	15	50	130	mA
			Vdd = Max, Vo = 0V	-5	-25	-100	mA

Note 1: IOL = 6 mA and IOH = -6 mA for the FMD11–0 and LBD11–0 pins; 4 mA for other pins.

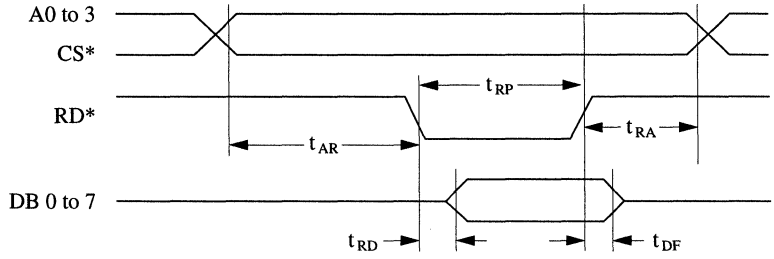
Note 2: When referring to the 4-mA buffers in Note 1, this parameter defines the current that flows when a high output is shorted to GND. Make all efforts to ensure that two or more outputs are never shorted to GND simultaneously. If this nevertheless occurs, the shorting time must be no more than 1 second.

6.5 AC Characteristics

(Note: The load capacitance of all output pins is assumed to be 30 pF.)

CPU Interface

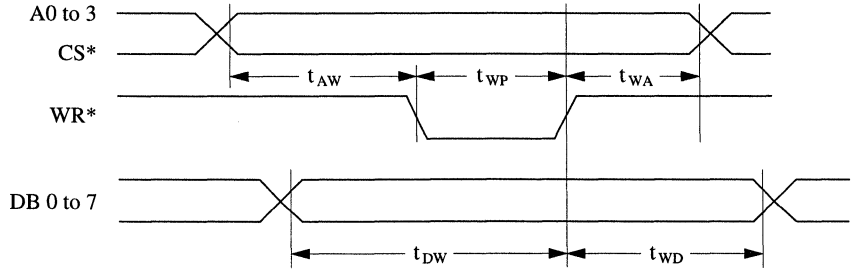
Read Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
A and CS* setup times to RD* falling	t_{AR}	5	—	—
A and CS* hold times to RD* rising	t_{RA}	2	—	—
RD* pulse width	t_{RP}	40	—	—
Delay time from RD* falling to DB enable	t_{RD}	5	—	30
Delay time from RD* rising to DB placed in Hi-Z	t_{DF}	3	—	20

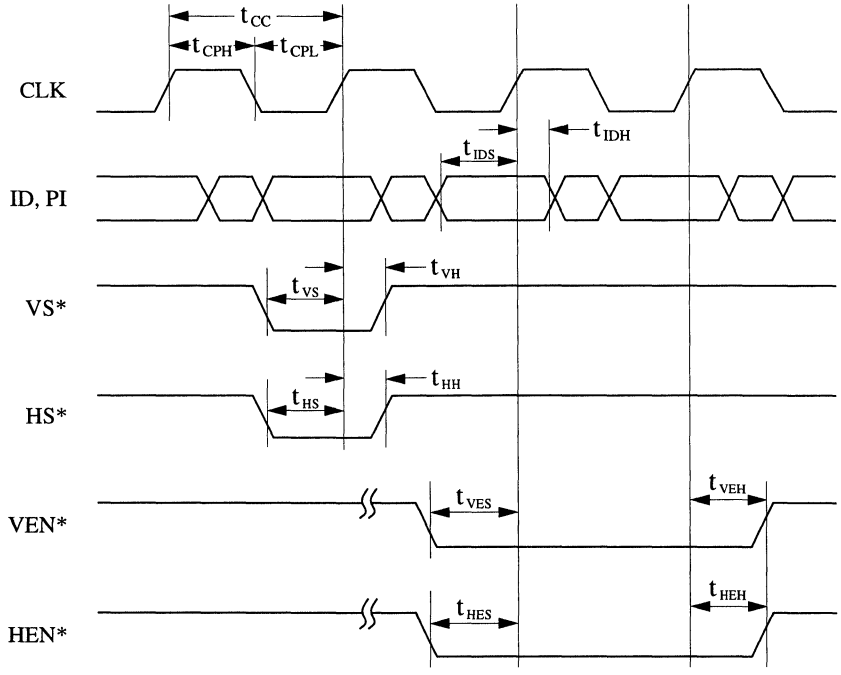
Write Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
A and CS* setup times to WR* falling	t_{AW}	5	—	—
A and CS* hold times to WR* rising	t_{WA}	2	—	—
WR pulse width	t_{WP}	40	—	—
DB setup time to WR* rising	t_{DW}	20	—	—
DB hold time to WR* rising	t_{WD}	2	—	—

Image Data Input

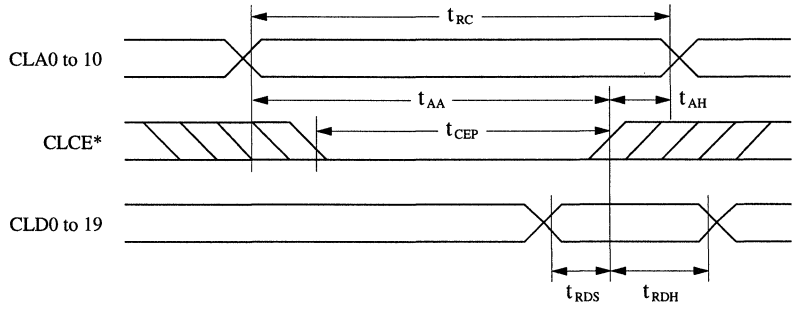


unit: ns

Parameter	Symbol	Min.	Typ.	Max.
CLK cycle	t_{CC}	50	—	—
High level period of CLK	t_{CPH}	15	—	—
Low level period of CLK	t_{CPL}	15	—	—
ID and PI setup times to CLK rising	t_{IDS}	12	—	—
ID and PI hold times to CLK rising	t_{IDH}	5	—	—
VS* falling setup time to CLK rising	t_{VS}	12	—	—
VS* rising hold time to CLK rising	t_{VH}	5	—	—
HS* falling setup time to CLK rising	t_{HS}	12	—	—
HS* rising hold time to CLK rising	t_{HH}	5	—	—
VEN* falling setup time to CLK rising	t_{VES}	12	—	—
VEN* rising hold time to CLK rising	t_{VEH}	5	—	—
HEN* falling setup time to CLK rising	t_{HES}	12	—	—
HEN* rising hold time to CLK rising	t_{HEH}	5	—	—

Label Table Interface

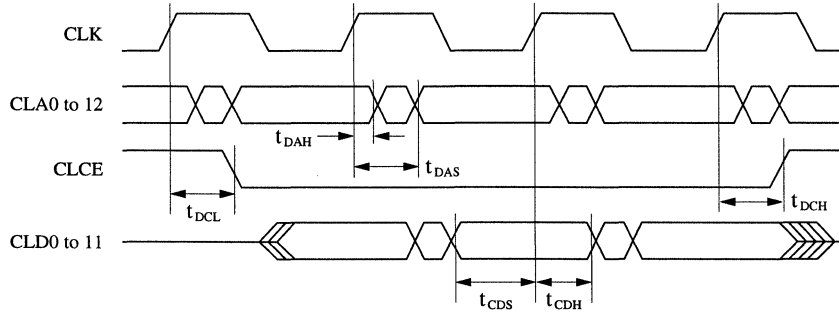
Read Cycle 1 (label linkage information processing)



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Read cycle time	t_{RC}	2tcc-5	2tcc	—
CLA access time to CLCE* rising	t_{AA}	30	—	—
CLA hold time to CLCE* rising	t_{AH}	5	—	—
CLCE* pulse width	t_{CEP}	2tcc-10	2tcc	—
CLD setup time to CLCE* rising	t_{RDS}	4	—	—
CLD hold time to CLCE* rising	t_{RDH}	5	—	—

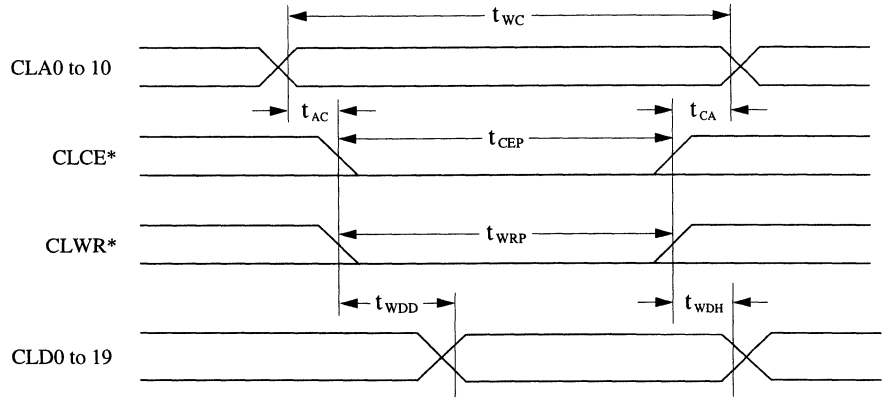
Read Cycle 2 (secondary labeling)



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
CLA hold time to CLK rising	t_{DAH}	5	—	—
Delay time from CLK rising to CLA enable	t_{DAS}	—	—	20
Delay time from CLK rising to CLCE* falling	t_{DCL}	—	—	20
Delay time form CLK rising to CLCE* rising	t_{DCH}	—	—	20
CLD setup time to CLK rising	t_{CDS}	4	—	—
CLD hold time to CLK rising	t_{CDH}	5	—	—

Write Cycle

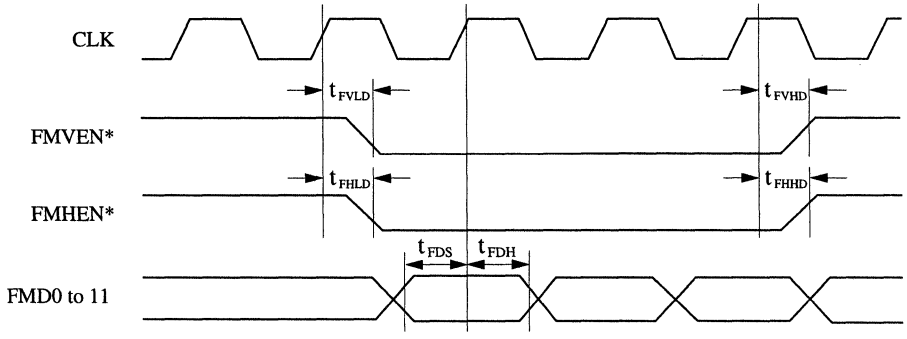


unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Write cycle time	t_{WC}	2tcc-5	2tcc	—
CLA setup time to CLCE* and CLWR* falling	t_{AC}	5	—	—
CLCE* pulse width	t_{CEP}	tcc-5	tcc	—
CLA hold time to CLCE* and CLWR* rising	t_{CA}	5	—	—
CLWR* pulse width	t_{WRP}	tcc-5	tcc	—
Delay time from CLCE* and CLWR* falling to CLD	t_{WDD}	—	—	17
CLD hold time to CLCE* and CLWR* rising	t_{WDH}	5	—	—

Frame Memory Interface

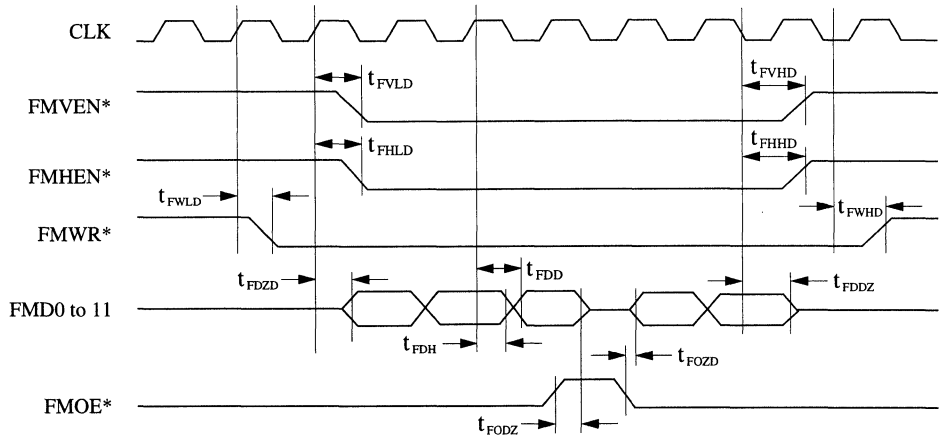
Read Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Delay time from CLK rising to FMVEN* falling	t_{FVLD}	—	—	25
Delay time from CLK rising to FMVEN* rising	t_{FVHD}	—	—	25
Delay time from CLK rising to FMHEN* falling	t_{FHLD}	—	—	25
Delay time from CLK rising to FMHEN* rising	t_{FHHD}	—	—	25
FMD setup time to CLK rising	t_{FDS}	4	—	—
FMD hold time to CLK rising	t_{FDH}	5	—	—

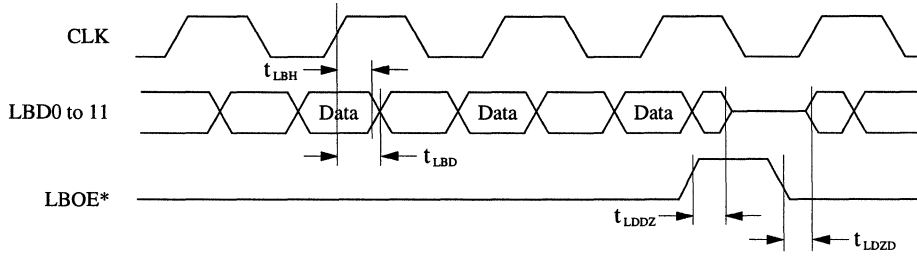
Write Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Delay time from CLK rising to FMVEN* falling	t_{FVLD}	—	—	25
Delay time from CLK rising to FMVEN* rising	t_{FVHD}	—	—	25
Delay time from CLK rising to FMHEN* falling	t_{FHLD}	—	—	25
Delay time from CLK rising to FMHEN* rising	t_{FHHD}	—	—	25
Delay time from CLK rising to FMWR* falling	t_{FWLD}	—	—	25
Delay time from CLK rising to FMWR* rising	t_{FWHD}	—	—	25
Delay time from CLK rising to FMD* enable	t_{FDZD}	—	—	25
Delay time from CLK rising to FMD* placed in Hi-Z	t_{FDDZ}	—	—	25
FMD hold time to CLK rising	t_{FDH}	4	—	—
Delay time from CLK rising to valid FMD	t_{FDD}	—	—	25
Delay time from FMOE* rising to FMD placed in Hi-Z	t_{FODZ}	—	—	30
Delay time from FMOE* falling to valid FMD	t_{FOZD}	—	—	20

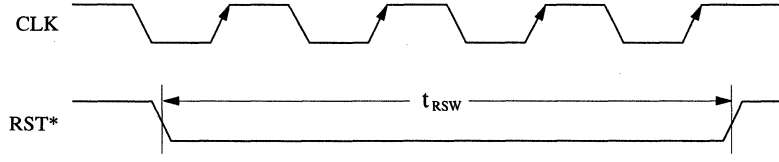
Secondary Labeling



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
LBD hold time to CLK rising	tLBH	4	—	—
Delay time from CLK rising to valid LBD	tLBD	—	—	20
Delay time from LBOE* rising to LBD* placed in Hi-Z	tLDDZ	—	—	20
Delay time from LBOE* falling to valid LBD	tLDZD	—	—	20

Reset Timing



The reset signal must be held active for at least three clock cycles.

unit: clock cycles

Parameter	Symbol	Min.	Typ.	Max.
RST* pulse width (Low period)	tRSW	3	—	—

Section 7: Sample Application

7.1 Sample Labeling Result

The LABop1K executes primary and secondary labeling in real time. However, the time required for label linkage information processing depends on the input image data and the clock rate. The table below lists the clock cycles required for label linkage information processing, as well as the time required (in ms) for the LABop1K to execute label linkage information processing on the sample images in Photos 1 through 3.

Example of Labeling Results

Image	No. of temp. labels	Number of final labels	Number of label linkage information entries	Linked data combination		
				Required clock cycles	Required time (ms)	
					15 MHz	20 MHz
Photo 1	650	366	297	8.5k	0.57	0.43
Photo 2	800	50	750	15k	1.03	0.77

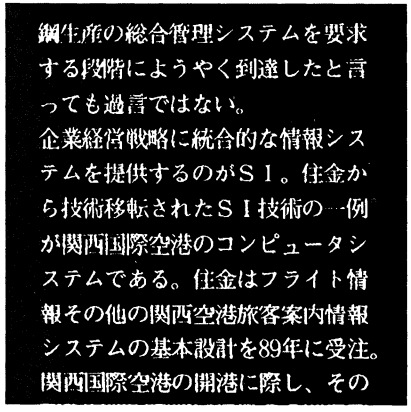


Photo 1: Text data

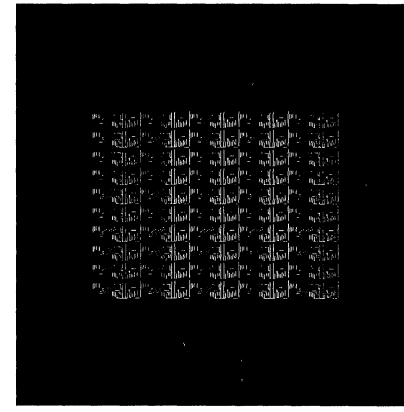
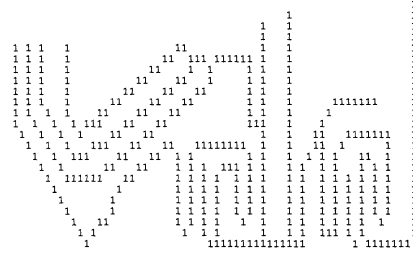
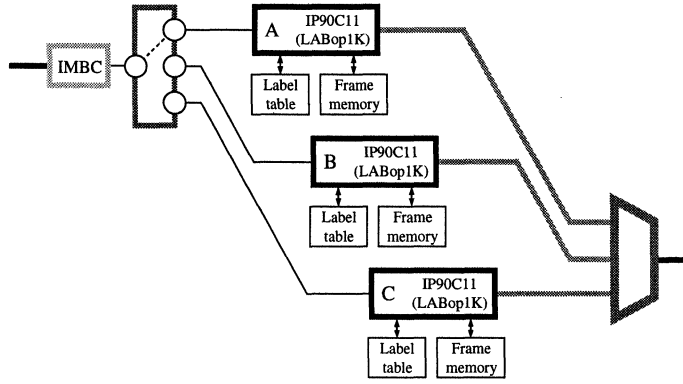


Photo 2: Hattori Pattern, in a 5 x 10 Array (Note)

Note: The diagram below is from T. Hattori, "New Regional Labeling Algorithm for Pipeline Image Processors and Proof of Correctedness", in proc. IECON '91, pages 2005–2010, 1991.

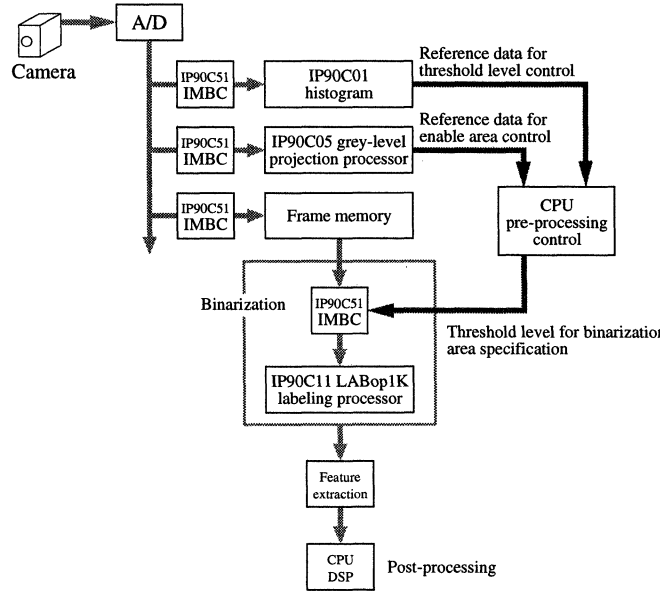


7.2 Real-Time Parallel Processing

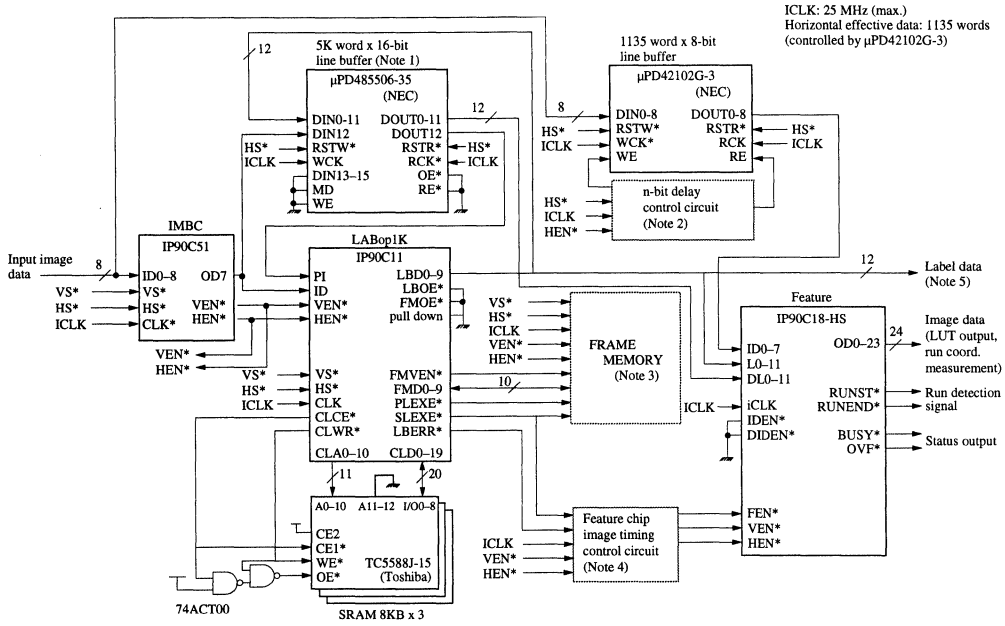


LABop ID (Seq.)	Frame number					
	0	1	2	3	4	.....
A (3n)	Primary	Combination	Secondary	Primary	Combination	Secondary
B (3n+1)		Primary	Combination	Secondary	Primary	Combination
C (3n+2)			Primary	Combination	Secondary	Primary

7.3 Sample Application Using an Image-Processing Device



7.4 Reference Circuits (Labeling Processing + Feature Extraction Processing) IP90C51 + IP90C11 + IP90C18



Note 1: This line buffer is used as a 1H line delay for the IP90C11 and IP90C18.

Note 2: The IP90C11 requires a frame memory design that generates a horizontal pixel delay factor. This line buffer is required to absorb that pixel delay factor. The delay value (n) depends on the frame memory design.

Note 3: This is the frame memory required by the IP90C11.

Note 4: This determines the processing area for the IP90C18. Use IP90C51s as necessary. In certain cases where the IP90C11 executes secondary labeling only, the circuit can be designed to function as if only the IP90C18 were operating.

Note 5: Frame memory should be controlled so that the input image data and label data appear on the same horizontal line (with no 1-line delay differential).

Appendix A: Functional Description and Algorithms for LABop1K

New Approach to and Implementation of an LSI for High-Speed Image Labeling

Nobuo Hayashi, Hiroshi Nittaya, Masahiro Kohno, and Masahiro Kato

Sumitomo Metal Industries, Ltd.
1–8 Fuso-cho Amagasaki Hyogo 660 Japan

Abstract: Image labeling plays an important role in image analysis and pattern recognition. This paper presents the use of a single accelerator LSI chip for image labeling, and proposes a new primary labeling algorithm suitable for hardware implementation. The algorithm determines primary labels by using a simple 2 x 2 matrix window. The LSI performs all labeling processes, including primary labeling, labeling unification, and secondary labeling. It runs at 40 MHz, and can generate up to 4094 primary labels for use in 4- or 8-connectivity.

A.1 Introduction

High-speed labeling of connected components in a binary image is one of the most fundamental problems in image analysis and pattern recognition.

Many algorithms and hardware designs have attempted to achieve high-speed labeling ([1]–[4]). Labeling connected components based on raster scanning consists of three stages: primary labeling, labeling unification, and secondary labeling. In primary labeling, a two-dimensional binary image is scanned from upper left to lower right to generate temporary labels. When the primary labeling process generates different temporary labels on a connected component and later recognizes them to be linked or part of the same connected pattern, the process then generates label linkage information that represents two temporary labels to be linked. In the linked data combination stage, a look-up table is generated by unifying label linkage information, which represents the relationship between temporary labels and final labels. Finally, secondary labeling generates final labels by referring to the look-up table and relabeling temporary labels accordingly.

Conventional algorithms for primary labeling concentrate mainly on achieving high-speed label unification by reducing the number of temporary labels and using relatively large matrix windows ([1], [2]). Previous papers have focused on the label unification process, though they dealt mainly with processing using software, which requires a large two-dimensional memory or complicated memory access to obtain a look-up table ([3], [4]).

This appendix presents a method of using a single LSI chip for high-speed image labeling, and proposes a new algorithm for primary labeling. Section A.2.1, "Primary Labeling," discusses this new algorithm, which determines temporary labels by using a simple 2 x 2 matrix window to simplify the hardware and achieve high-speed processing, while still using the same number of temporary labels as conventional methods. Section A.2.2, "Linked Data Combination," then presents the linked data combination algorithm, which requires one-dimensional memory and is suitable for use in hardware. Finally, Section A.3, "Hardware Architecture," describes the hardware and specifications of the labeling LSI.

A.2 Algorithms

A.2.1 Primary Labeling

In this discussion, two-dimensional image input is represented by a binary number in which pixels marked 1 denote the target, and the 0-pixels denote the background. In primary labeling, a matrix window scans a binary image from upper left to lower right to generate temporary labels and label linkage unification. The primary labeling algorithm uses a simple 2 x 2 matrix window (see Figure A-1) to recognize the target image and its connectivity, and then determines temporary labels according to the 16 conditions of the matrix window shown in Figure A-2.

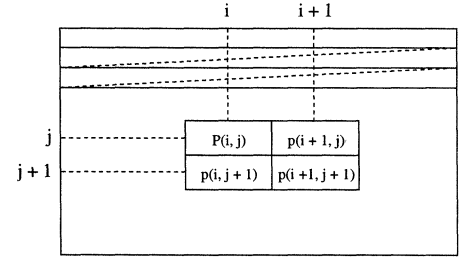


Figure A-1: Raster Scanning a 2 x 2 Matrix Window

The operations "R<-f(RLB)" and "R<- g(R, RLB)" determine temporary labels (the notation "<--" represents a substitution):

R <-- f(RLB):
X <-- RLB
if X = 0 then R <-- LC and LC <-- LC + 1
else R <-- X

R <-- g(R, RLB):
X <-- RLB
(1) if X = 0, R = 0, then R <-- LC and LC <-- LC+1
(2) if X ≠ 0, R = 0, then R <-- X
(3) if X ≠ 0, R ≠ 0, then do nothing
(4) if X ≠ 0, R ≠ 0, X = R, then do nothing
(5) if X ≠ 0, R ≠ 0, X ≠ R, then LIB <-- X, R and R <-- min(X, R)

where:

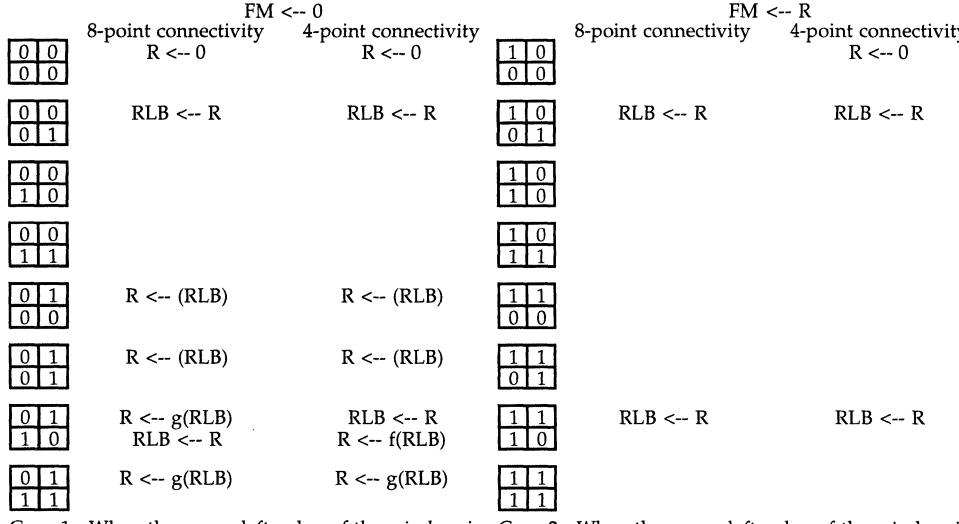
R:	Temporary label register value.
X:	Run label register value.
RLB:	Run label buffer value. This buffer stores run labels and works as FIFO memory.
LIB:	Linkage information buffer value. This buffer stores label linkage information.
LC:	Label counter value that represents the number of temporary labels.

The temporary label register value R is terminated according to the procedures shown in Figure A-2. When the value in the upper-left of the matrix window is 0 (background), 0 is stored in frame memory; when the value in the upper-left of the matrix window is 1 (1 indicates the target image), R is stored in frame memory as a temporary label. Label linkage information is also stored in the linkage information buffer when it is generated.

This algorithm recognizes connectivity, and gives a temporary label for the image that has a depression 1-pixel depth or diagonal pattern, thereby reducing the number of temporary labels generated. The algorithm corresponds to both 4- and 8-point connectivity by slightly modifying the procedure shown in Figure A-2.

A.2.2 Linked Data Combination

The linked data combination process unifies the label linkage information in the linkage information buffer, and generates a look-up table that determines how temporary and final labels correspond. The linked data combination algorithm generates a chain label information table that represents the connectivity chain of temporary labels based on the label linkage information before the look-up table is generated.



Case 1: When the upper-left value of the window is 0, 0 is written to the frame memory (FM).

Case 2: When the upper-left value of the window is 1, R is written to the frame memory (FM).

Figure A-2: Definition of Primary Labeling Operation

The algorithm consists of the three steps shown in Figure A-3: clear, chain, and table generation. Assume the label table is a one-dimensional memory. In the clear and chain steps, the label table works as a chain label information table; in the table generation step, it works as a look-up table. In the chain label information table the notation "(m) = n" means that n represents the contents of the label table denoted by address m, in which m must be greater than n (every temporary label m has chain information with a smaller-numbered temporary label n). When n = 0, temporary label m has no chain information with smaller temporary labels.

The clear step initializes the label chain table by clearing its contents (i.e., any temporary label has no chain information with smaller-numbered temporary labels).

The chain step then generates a chain label information table in which each temporary label has connectivity with only one smaller-numbered temporary label. This table is generated by reading label linkage information (label A_0 connects with label B_0) from the label linkage buffer, and searching for connectivity for each A_0 and B_0 in the chain label information table in the direction of smaller-numbered temporary labels. Let A_i and B_j be the smallest number of temporary labels that connect with A_0 and B_0, respectively. $(A_i) = B_j$ is written in the chain label information table when $A_i > B_j$, and $(B_i) = A_j$ is written in the chain label information table when $A_i < B_j$. As a result, temporary labels A_0 and B_0 are connected with each other on the chain label information table through A_i and B_j connectivity. By applying these operations to all label linkage information in the linkage information buffer, every temporary label either has connectivity with only one smaller-numbered temporary label, or has no connectivity with smaller-numbered temporary labels.

The table generation step generates the look-up table. In this step, the contents of the chain label information table are modified and arranged sequentially from the smallest address number. When the contents of the chain label information table are 0 (temporary label has no connectivity with a smaller-numbered temporary label), a new label is assigned. When the contents of the chain label information table are non-zero (temporary label has connectivity with a smaller-numbered temporary label—already modified to be a final label), this label is assigned. By applying these operations to the contents of the chain label information table sequentially from the smallest address number, the look-up table is generated from the label table.

Since the algorithm requires one dimensional memory and consists of simple memory access and data comparison, this algorithm is suitable for hardware implementation.

(Step 1) Clear
- (1) k <-- 0
- (2) (k) <-- 0
- (3) k <-- k+1
- (4) if k < LC then go to (2)
- (5) end

where (k) represents contents of label table denoted by address k, and LC is the final number of temporary label.

(Step 2) Chain
- (1) k <-- 0
- (2) m, n <-- LIB(k)
- (3) if (m) ≠ 0 then m <-- (m) and go to (3)
- (4) if (n) ≠ 0 then n <-- (n) and go to (4)
- (5) if m < n then (n) <-- m
  else if n < m then (m) <-- n
- (6) k <-- k + 1
- (7) if k<LIC then go to (2)
- (8) end

where LIB(k) represents the two labels linkage information stored in the linkage information buffer addressed by k, and LIC is the number of the label linkage information.

(Step 3) Table generation
- (1) c <-- 1 and k <-- 1
- (2) if (k) = 0 then (k) <-- c and c <-- c + 1
  else (k) <-- (k)
- (3) k <-- k+1
- (4) if k < LC then go to (2)
- (5) end

Figure A-3: Labeling Unification Algorithm

A.3 Hardware Architecture

The labeling system consists of a labeling LSI, one-line delay, frame memory, linkage information buffer, and label table, as shown in Figure A-4. The labeling LSI consists of three main modules (primary labeling, labeling unification, and secondary labeling), and executes these processes sequentially. First, the primary labeling module generates temporary labels and label linkage information from binary image data input and one-line delayed data input. The generated temporary labels and label linkage information are stored in the frame memory and the linkage information buffer, respectively. Next, the linked data combination module generates the chain label information table from the label table based on the label linkage information in the linkage information buffer. After the chain label information table is generated, it is modified to obtain a look-up table. Finally, the secondary labeling module generates final labels by referring to the look-up table and relabeling the temporary labels.

The primary labeling algorithm is implemented using hardware shown in the block diagram in Figure A-5. The labeling LSI executes primary labeling with four-stage pipeline processing and contains the run label buffer, thus providing high-speed hardware performance. The linked data combination algorithm is also implemented in the hardware, so efficient and high-speed linked data combination can be achieved by designing for managing memory access and data comparison.

The main features of the labeling LSI are described below:

- All labeling processes are executed in a single chip.
- The maximum input image data rate is 40 MHz, and the entire labeling process is executed in three frames.
- The maximum number of generated temporary labels is 4094.
- The maximum number of generated label linkage information entries is 4095.
- 4-point and 8-point connectivity are available.

By using the three labeling systems in parallel, real-time video rate labeling can be achieved.

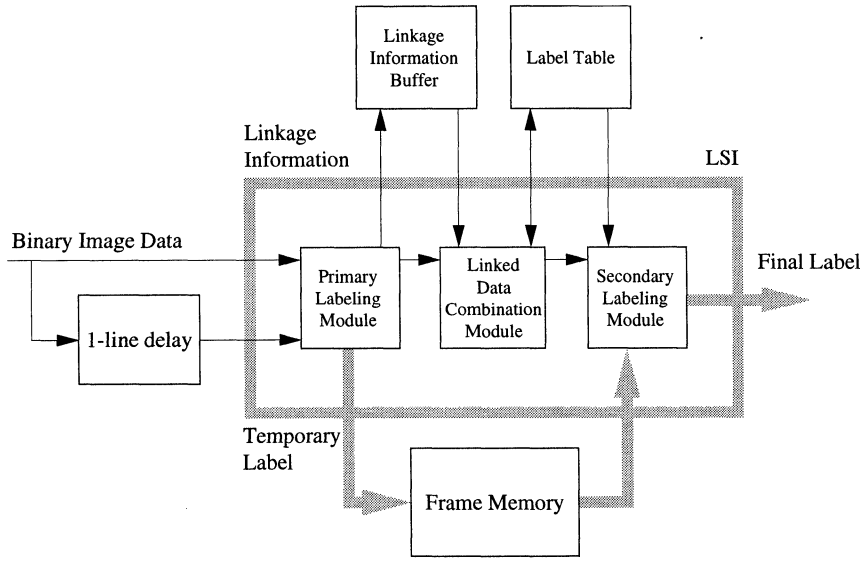
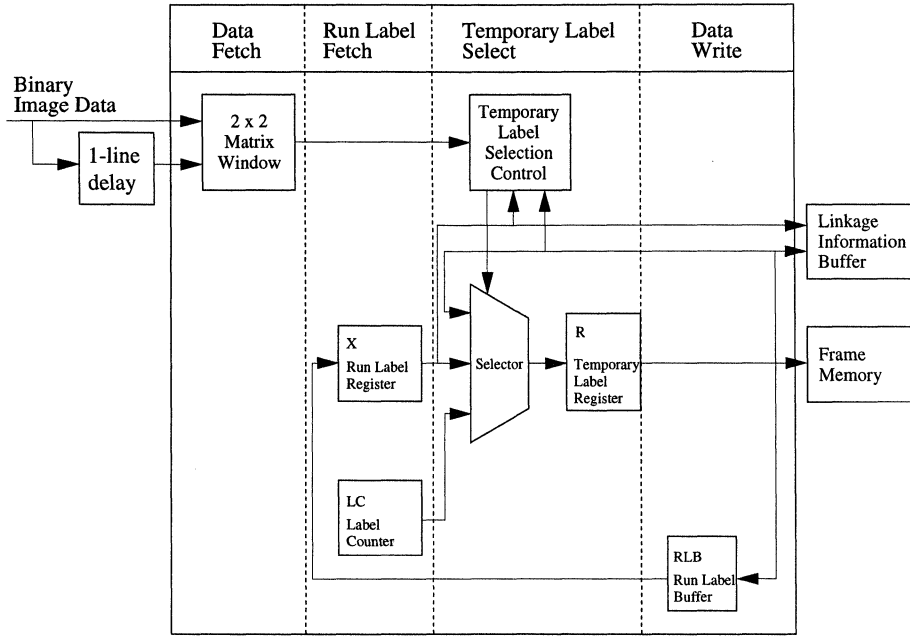
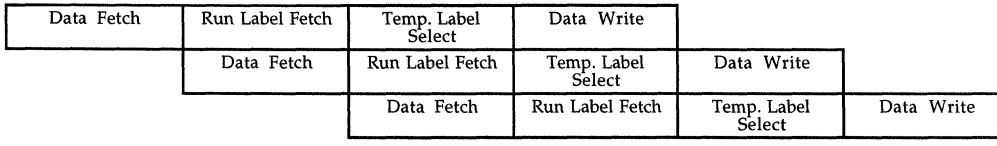


Figure A-4: Block Diagram of the LSI for High-Speed Image Labeling



(a) Block Diagram



(b) Pipeline Processing

Figure A-5: Architecture for Primary Labeling

A.4 Conclusion

Sumitomo has developed an LSI for high-speed image labeling, and a new primary labeling algorithm suitable for hardware implementation. The labeling LSI contains the entire labeling operation on a single chip that runs at 40 MHz. It has many applications, particularly in real-time video rate labeling processing.

Acknowledgment

The authors would like to thank Prof. T Hattori of Kagawa University for his helpful suggestions, and J. Watanabe and H. Tsuge for their encouragement.

References

[1] T. Hattori, "New Regional Labeling Algorithm of Pipeline Image Processor and Proof for the Correctness." in proc. IECON'91, pp. 2005–2030, 1991.

[2] H. Fujiwara, F. Isogsi. "A Method for High-performance labeling processor,"(in Japanese) Computer Vision, No, 56, pp. 1–7, 1988.

[3] Y. Shirai, K. Sugihara, "Algorithms and Hardwares for Labeling Connected Regions," (in Japanese) IECE Tech. Report IE78-9, pp. 29-33, 1978.

[4] T. Goto. Y. Ohta, M. Yoshida and Y. Shirai, "High speed labeling for component labeling," (in Japanese) Tran IEICE D-IT vol. 372-D-II, No. 2, pp. 247-255, 1989.

SMI ASSP

Image Processing LSI Series

IP90C15

Image Data Reduction Processor by Averaging (Sketch)

Sumitomo Metal Industries, Ltd.

Technical Manual Ver. E 1.4

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Overview

1.1 Product Outline

The IP90C15 "Sketch" LSI chip is an average value reduction LSI for 8 bit grayscale image input. The Sketch chip calculates the average value of gray levels in a local area of up to 16 x 16 pixels, using arithmetic add and shift-based division operations, then outputs the results in real time. This device enables fast, highly efficient image data reduction in a simple system configuration at a maximum operating frequency (input image data transfer rate) of 40 MHz. The Sketch chip has distributed processing capability, allowing multiple ships to be connected in cascade fashion to process images of unlimited size.

1.2 Features

- Maximum operating frequency f_{max} = 40 MHz
- Image data reduction function by averaging
  - Fast, efficient average value compression of image data:

    The device calculates the average value of grayscale levels in a local area of the 8-bit input image by performing arithmetic (add and right-shift) operations, and then outputs the result in real time.
  - Maximum local area of 16 x 16 pixels:

    The height and width of a local area can be set independently to any value from 1 to 16 pixels.
  - The divisor can be set to any number regardless of the size of the local area:

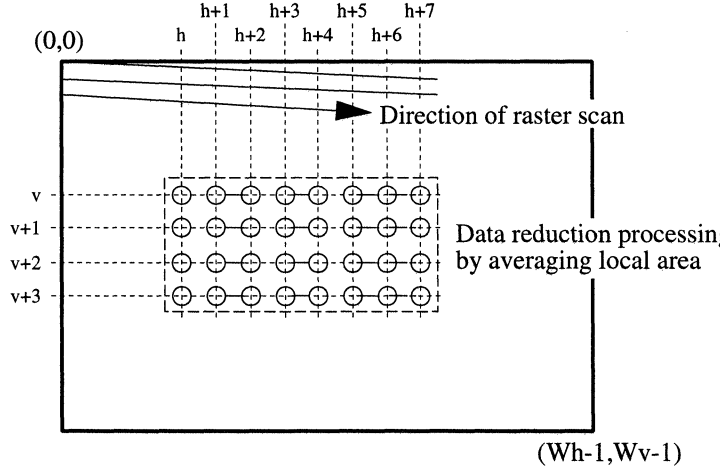
    Division is done by shifting bits to the right by 0 to 15 bits.
  - Internal memory for intermediate results:

    The device has a memory for storing the intermediate results of addition operations, making it unnecessary to attach external memory to the board.
  - Maximum local area width of 512 pixels:

    Multiple Sketch devices can be used for distributed processing (cascade connection) to expand processing of an unlimited number of local areas.
  - Unlimited local area height
- External interface
  - Status output to CPU for memory overflow errors in intermediate addition results.
  - General-purpose interface can be connected directly to the CPU bus.
  - I/O is TTL-level compatible.
- Process: CMOS
- Power supply: 5V single power supply
- Package: 64-pin QFP (mold section = 20 x 14 mm)

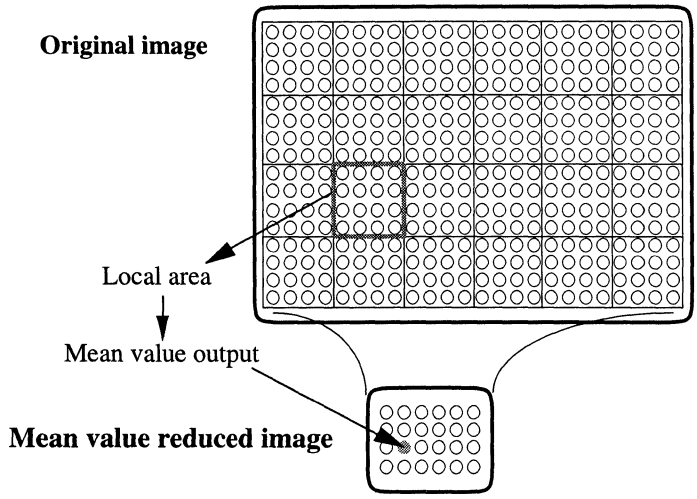
1.3 Input Data and Device Operation

The Sketch chip accepts direct input of raster-scanned 8 bit grayscale images. It calculates the total sum of gray-level values in the specified local area of the image, then outputs the result after dividing the sum by a specified value. Intermediate addition results are stored in the device's internal memory, so no external line delay or memory is needed. The average value in each local area is calculated and output immediately after input is complete.



Input Data for Image Data Reduction LSI (Sketch)
Example: 4 (vertical) x 8 (horizontal) Local Area

In the above example, the device reduces an input image screen to 1/4 of its original height and 1/8 of its original width, thus reducing a screen of (Wh, Wv) to (Wh/4, Wv/8). However, if the screen size is not an integral multiple of the width of the local area (for example, if Wh/4 or Wv/8 in the above example was not an integer), the extra columns on the right edge and extra rows at the bottom are ignored and not processed.



Average Value Reduced Image

Example of Image Reduction by Averaging

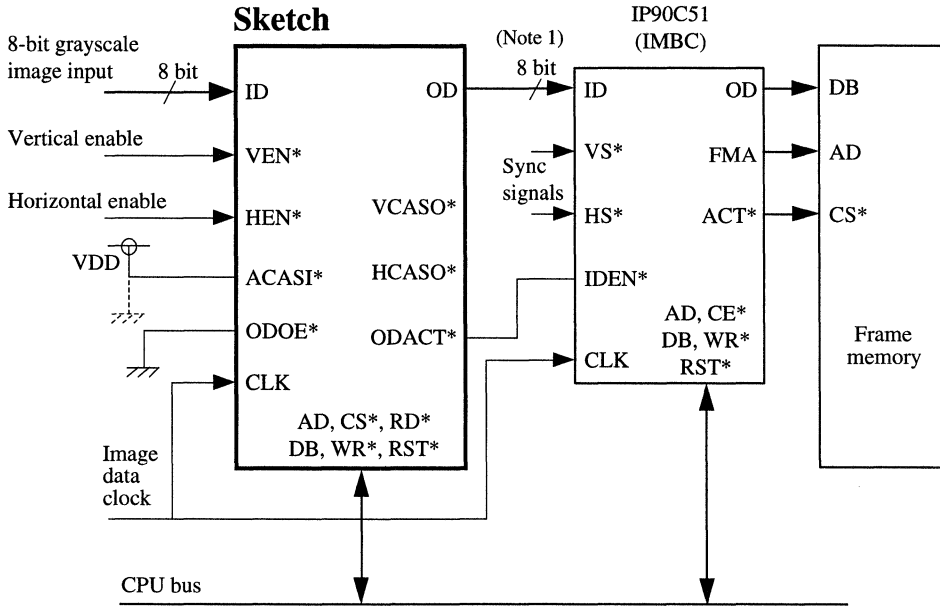
The figure above shows a local area of 4 x 4 pixels specified from within an original image of 24 x 16 pixels. After image reduction by averaging, the image is reduced to 6 x 4 pixels.

Section 2: Sample System Configuration

2.1 Sample Basic System Configuration

The Sketch chip accepts 8-bit grayscale image data input by raster scanning and outputs the average value data of gray levels in each local area when input for that area is complete.

The figure below shows an example in which the frame memory stores this intermittent output average value data using the IP90C51's (IMBC's) pixel control function.



Note 1: Connect the OD7–0 pins on the Sketch chip to the ID7–0 pins on the IMBC chip. Since the sum of gray level values is divided by the number of pixels in the local area, the data is normally output in 8-bit units. Therefore, output data lines OD15–8 are always 0 and need no subsequent processing, and so also need not be connected.

Note 2: The Sketch chip allows the size of the local area and the divisor for the sum of gray level values to be set independently. It also has an OD bus width of 16 bits to provide output of the sum of gray level values for a local area of 16 x 16 pixels (divisor = 1). Therefore, if the divisor is set to a value smaller than the number of pixels in the local image area, the image output may exceed 8 bits. In this case, the data cannot be transferred accurately because the IP90C51 chip (IMBC) image input bus is only 8 bits wide.

Note 3: For descriptions of each signal pin on the Sketch chip, see Section 3.3, "Pin Description," and Section 5, "Outline of Operation."

Note 4: An asterisk (*) after the signal name indicates inverse logic (active low).

Note 5: This example of a basic system configuration is provided only as an illustration, and is not guaranteed to work with particular applications, timings, or other operational conditions. Carefully check the manuals and other data supplied with your devices when designing a circuit.

2.2 Sample Expanded Screen Width Configuration

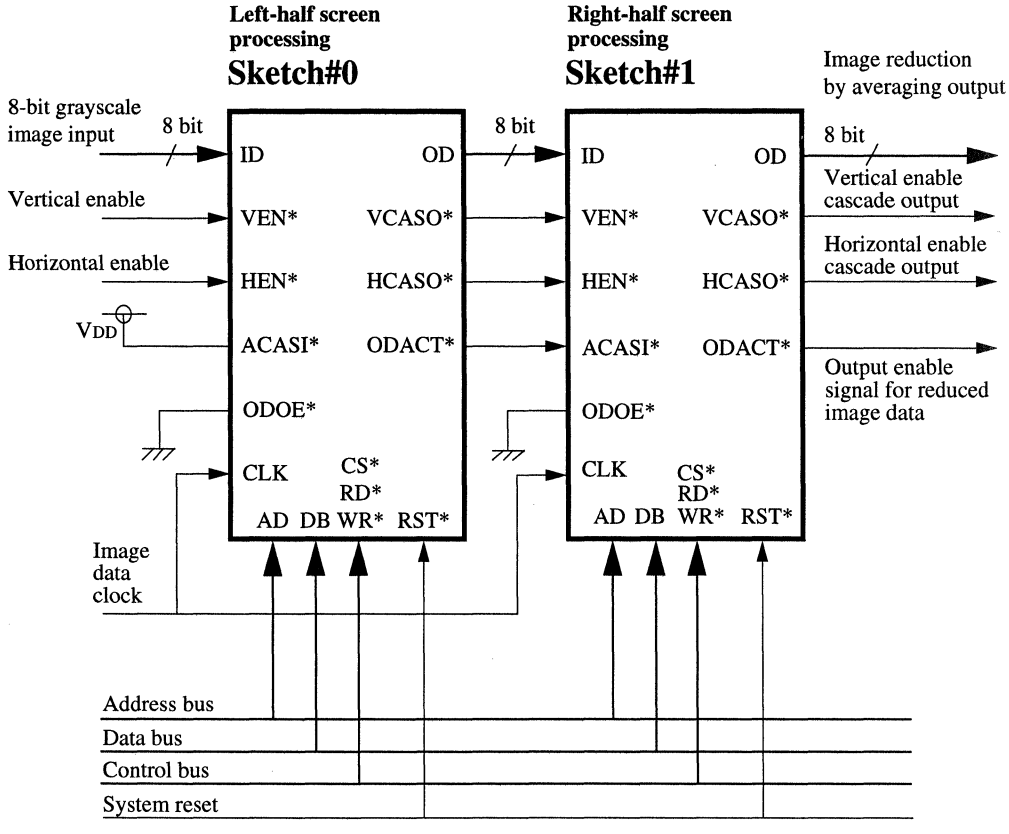
The diagram on the following page shows an example of a system that expands screen width using distributed processing (cascade configuration). The width of the image area processed by a Sketch chip is limited to the width of the specified local area x 512 pixels. For example, if the width of the local area is 2 (because the image is compressed to 1/2), a screen width of up to 1024 pixels can be processed.

Because each Sketch chip includes a distributed processing function, image data exceeding the chip's limit can be sent to the next chip in the cascade connection for processing. This allows any desired screen width to be processed by cascading the appropriate number of Sketch devices. The diagram on the following page shows an example in which the processed image's width is doubled by using two Sketch devices.

With this system, processing is performed so that when the horizontal effective area is entered for each line within a given vertical effective area, Sketch chip #0 starts average value compression processing first. During this time, output results pass through Sketch chip #1. When Sketch chip #0 reaches its processing limit, it passes its received input directly to Sketch chip #1 for processing.

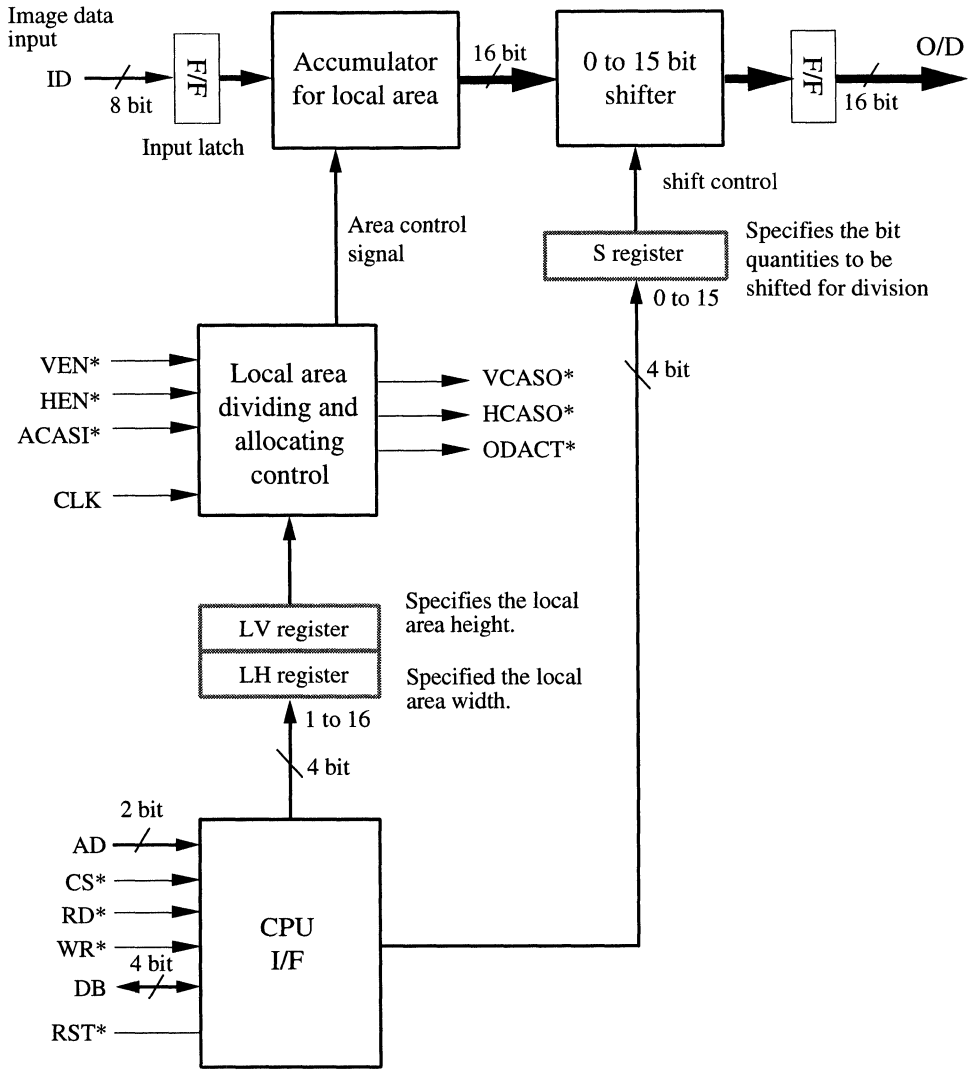
This system imposes no limit on the effective image area's height.

The Sketch chip allows the size of the local area and the divisor for the sum of gray level values to be set independently. It also has an OD bus width of 16 bits to provide output of the sum of gray level values for a local area of 16 x 16 pixels (divisor = 1). Therefore, if the divisor is set to a value smaller than the number of pixels in the local image area, the image output width may exceed 8 bits. In this case, the data will not be transferred to the next cascade device because the image input bus of the Sketch chip is only 8 bits wide.



Note: An asterisk (*) after the signal name indicates inverse logic (active low).

2.3 System Block Diagram

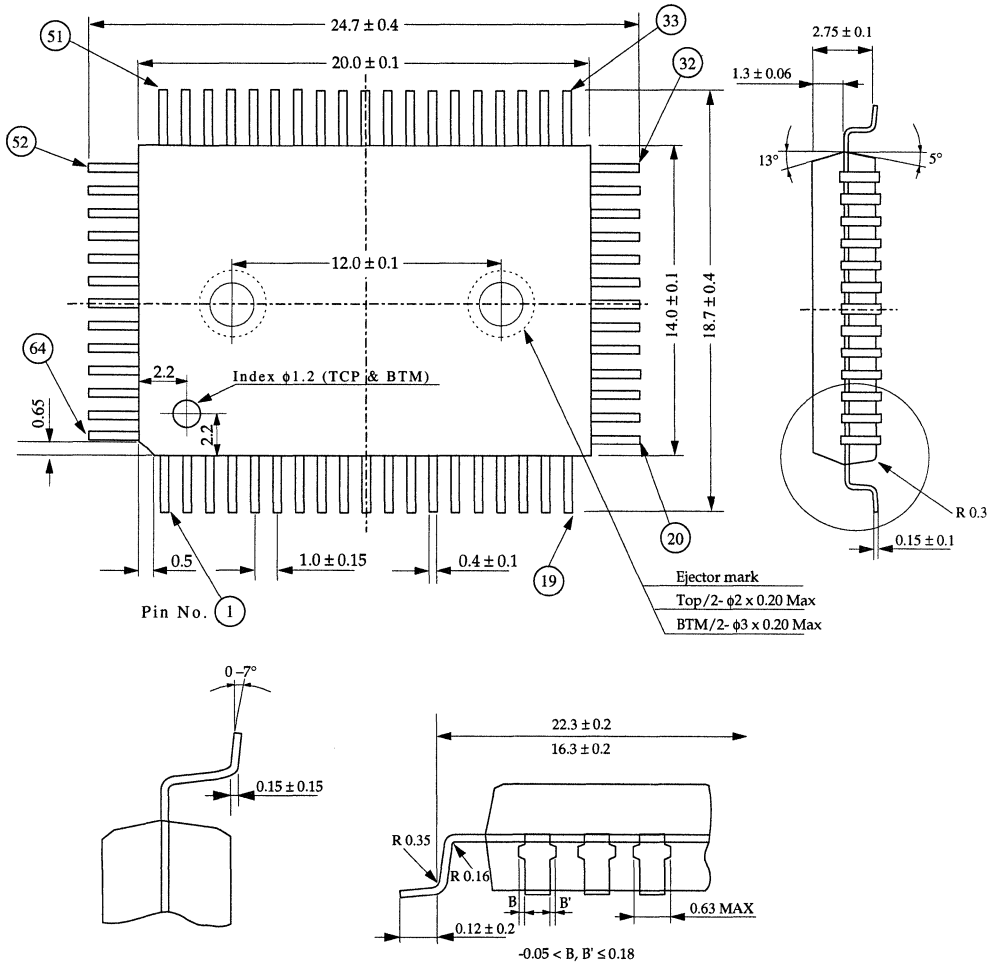


Note 1: This block diagram only shows the general configuration of the Sketch chip; it does not show its functions in detail. Refer to the corresponding sections in this manual for timing and other details.

Note 2: An asterisk (*) after the signal name indicates inverse logic (active low).

Section 3: Pin Descriptions

3.1 Package Dimensions



64-pin plastic QFP units: mm

3.2 Pins and Their Functions

Pin group	Symbol	No. of Pins	Type	Description
Image input bus	CLK	1	I	Clock input f_{max} = 40 MHz
	ID0–7	8	I	Image data input
	VEN*	1	I	Vertical enable signal
	HEN*	1	I	Horizontal enable signal
	ACASI*	1	I	Image output enable signal cascade input
Image output bus	OD0–15	16	O	Average value reduced image output
	VCASO*	1	O	Vertical enable signal cased output
	HCASO*	1	O	Horizontal enable signal cascade output
	ODACT*	1	O	Average value reduced image output enable signal
	ODOE*	1	I	OD output enable signal
CPU bus	AD0–1	2	I	Register select address bus
	CS*	1	I	Chip select
	RD*	1	I	Read enable
	WR*	1	I	Write enable
	DB0–3	4	I/O	Data bus
	RST*	1	I	System reset (Note 1)
Power supply and GND	Vdd	8	PW	5V
	GND	8	PW	Ground
Test	TEST1, TEST2	2	I	Test input pin (Note 2)
Unconnected	N.C.	4	I	No-connection pin
Total number of pins		64		

The Sketch chip is packaged in a 64-pin QFP (rectangular, with a 20 x 14-mm mold section and a 1.0-mm pin pitch).

Note 1: The device is reset with a low pulse having a duration of three clock cycles or more. This pin has an internal pull-up resistor.

Note 2: Test pins (TEST1, TEST 2) are used to test the device's internal logic, and must be held low during normal use. These pins also include internal pull-down resistors.

Note 3: An asterisk (*) after the signal name indicates inverse logic (active low).

3.3 Pin Description

Pin name	Description
ID0–7	8 bit grayscale image data input.
VEN*	Vertical enable signal. When VEN* and HEN* are both low, an area is defined in which the Sketch chip applies average value reduction of the image data.
HEN*	Horizontal enable signal. When VEN* and HEN* are both low, an area is defined in which the Sketch chip applies average value reduction of the image data.
ACASI*	Image output enable signal cascade input. During average value reduction, the Sketch chip sends an image output enable signal to ODACT* regardless of the input value of ACASI*. If the Sketch chip is not performing average value reduction, it passes input from ACASI* directly to ODACT*. If the chip is used in a cascade configuration, ODACT* must be connected to ACASI* in the following stage. Entering a low-level signal through ACASI* to a Sketch chip (in either a single device or the first stage of a cascade configuration) causes ODACT* in another Sketch chip (either a single device or the last stage of a cascade configuration) to output an image output enable signal (low). Entering a high-level signal through ACASI* causes ODACT* to send an image output disable signal (high).
OD0–15	16-bit average value reduction image output. When 8-bit grayscale image data is entered, Sketch can output the sum of the gray levels in a local area of up to 16 x 16 pixels (the device being set to LH = Fh, LV = Fh, DIV = 0). Outputs are therefore 16 bits wide.
VCASO*	Vertical enable cascade signal output. When using Sketch chips in cascade configuration, connect this output to the VEN* in the next Sketch stage. This outputs the VEN* status.
HCASO*	Horizontal enable cascade signal output. When using Sketch chips in a cascade configuration, connect this output to the HEN* signal in the next Sketch stage. If either VEN* or HEN* is high, this pin outputs a high-level signal. If VEN* and HEN* are both low, this pin outputs a high-level signal until the number of local areas in the horizontal direction reaches 512, and a low-level signal thereafter. The HCASO* signal (in either a single Sketch chip or the last stage of a cascade configuration) can also be used to generate an overflow error signal.
ODACT*	Image output enable signal output. This pin outputs an image output enable signal while the Sketch chip is performing average value reduction processing. If the chip is not performing this processing, this pin outputs the input value of ACASI*. When using multiple Sketch chips in a cascade configuration, connect this pin to the ACASI* pin in the next Sketch chip in the sequence.
ODOE*	Average value reduction image output enable signal input. When this pin is high, the data bus lines OD0–15 are kept in a high-impedance state. The data bus OD is asynchronous with the image clock and responds immediately to changes in the input ODOE* value.

Each output pin outputs its corresponding value five clock cycles after the input status changes. However, this does not apply to the data bus OD, though it does respond immediately to the ODOE* input.

Note: An asterisk (*) after the signal name indicates inverse logic (active low).

3.4 Pin Configuration

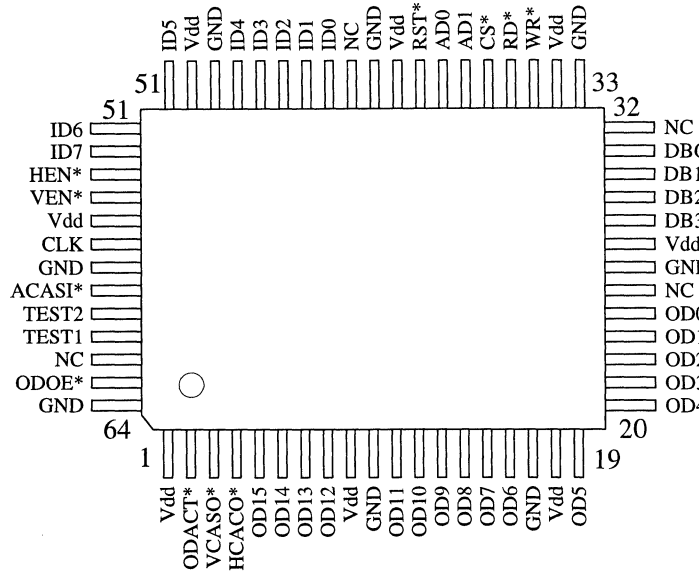


Table of Pin Assignments

Pin	Name	Type	Pin	Name	Type	Pin	Name	Type	Pin	Name	Type
1	Vdd	PW	17	GND	PW	33	GND	PW	49	GND	PW
2	ODACT*	O	18	Vdd	PW	34	Vdd	PW	50	Vdd	PW
3	VCASO*	O	19	OD5	O	35	WR*	I	51	ID5	I
4	HCASO*	O	20	OD4	O	36	RD*	I	52	ID6	I
5	OD15	O	21	OD3	O	37	CS*	I	53	ID7	I
6	OD14	O	22	OD2	O	38	AD1	I	54	HEN*	I
7	OD13	O	23	OD1	O	39	AD0	I	55	VEN*	I
8	OD12	O	24	OD0	O	40	RST*	I	56	Vdd	PW
9	Vdd	PW	25	NC	—	41	Vdd	PW	57	CLK	I
10	GND	PW	26	GND	PW	42	GND	PW	58	GND	PW
11	OD11	O	27	Vdd	PW	43	NC	—	59	ACASI*	I
12	OD10	O	28	DB3	I/O	44	ID0	I	60	TEST2	I
13	OD9	O	29	DB2	I/O	45	ID1	I	61	TEST1	I
14	OD8	O	30	DB1	I/O	46	ID2	I	62	NC	—
15	OD7	O	31	DB0	I/O	47	ID3	I	63	ODOE*	I
16	OD6	O	32	NC	—	48	ID4	I	64	GND	PW

Note: An asterisk (*) after the signal name indicates inverse logic (active low).

Address		Symbol	Width bit	Functional outline
AD1	AD0			
0	0	LH	4	Width of the local area for calculating the average value. The value in this register should be the width of the local area minus 1.
0	1	LV	4	Height of the local area for calculating the average value. The value in this register should be the height of the local area minus 1.
1	0	DIV	4	Shift quantity to be applied for the average value calculation. Use this register to write the desired shift value.
1	1	MODE	4	Mode register. Use this register to specify either software reset or test mode.

Note 1: All registers can be written to or read from.

Note 2: All registers are cleared to 0 by hardware or software resets.

Note 3: Shift registers are used to perform division operations to calculate the average value.

To specify the n^{th} power of 2 as the divisor, write 'n'.

For example, to specify a 2 x 2 local area, the sum of the gray level values must be divided by the square (second power of 2) to obtain the average value of 4 pixels. In such a case, 2 would be written into the DIV register. The sum of gray level values will be shifted 2 bits to the right, yielding 1/4 of the sum.

Note 4: The configuration of the flag values in the mode register are shown below. Test modes cannot normally be used, so the corresponding bits should be set to 0.

b3			b0
t2	t1	t0	r

r: Reset flag register

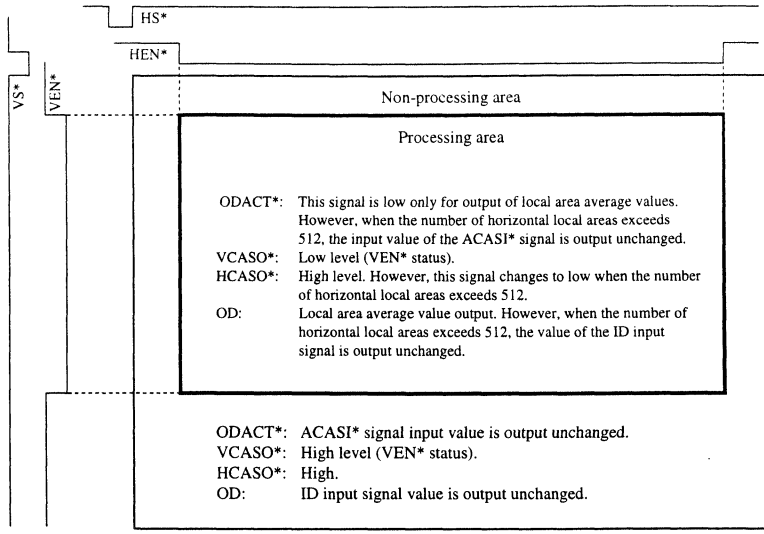
The device is reset by writing 1 to this flag bit. In this case, all registers, except this bit, are cleared to 0. The reset is cleared by writing 0 to this bit.

t0: Test mode 0

t1: Test mode 1

t2: Test mode 2

Section 5: Outline of Operation



Input			Output			
VEN*	HEN*	Operating status	VCASO*	HCASO*	ODACT*	OD
L	H	Waiting for data	L	H	ACASI*	ID
H	X (Note 1)		H	H	ACASI*	ID
L	L	Executes average value reduction processing until the number of pixels in the horizontal direction reaches 512	L	H	Output data enable signal	Average value for local area
		Passes average value reduction processing on to the next stage when the number of pixels in the horizontal direction exceeds 512 (equal to or greater than 513)	L	L	ACASI*	ID

Note 1: "X" indicates an H or L signal.

Note 2: An asterisk (*) after the signal name indicates inverse logic (active low).

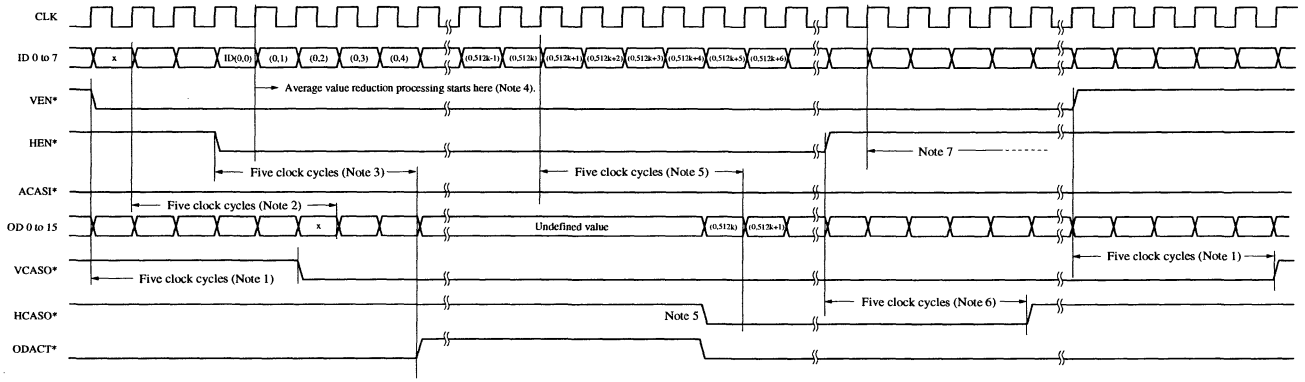
Each output pin outputs its corresponding value five clock cycles after the input pin status changes.

The figure above shows the output state of each of the related signals. The Sketch chip enters standby mode to wait for image data when reset. When this happens, VCASO*, ODACT*, and OD output the values input from VEN*, ACASI*, and ID, respectively, with a delay of five clock cycles, and HCASO* remains high.

The Sketch chip enters processing mode when VEN* and HEN* are both asserted low, and immediately starts average value reduction processing. While processing, the Sketch chip performs an addition operation for each pixel and stores the intermediate results in its internal memory. When the number of pixels in the horizontal direction reaches 512, the internal memory overflows, causing the Sketch chip to interrupt processing and enter a send-to-next stage mode.

In the send-to next stage mode, the Sketch chip sends its input directly to output, thus passing the input image data onto the next Sketch stage in a cascade configuration. The Sketch chip then pulls VCASO* and HCASO* low, causing VEN* and HEN* to be set low in the next Sketch stage, which causes that chip to start average value compression processing. Therefore, when using multiple Sketch devices in a cascade configuration, the VCASO* and HCASO* in each stage must be connected to VEN* and HEN* in each of the following stages. If HCASO* in the last stage is driven low, an error occurs because the input image data has a width greater that the processing capability of the system. If either VEN* or HEN* is driven high, the Sketch chips are brought back into standby mode.

When the Sketch chip reaches the last row of pixels, it uses right-shift division on horizontal pixels, then outputs the result. The image can have any height.



Note 1: When VEN* changes level, VCASO* changes to the same level five clock cycles later.

Note 2: If either VEN* or HEN* is not low (that is, if the Sketch chip is not performing average value reduction processing), data is output from ID to OD five clock cycles after the data is input.

Note 3: If either VEN* or HEN* is not low (that is, if the Sketch chip is not performing average value reduction processing), ODACT* outputs the input value of ACASI* five clock cycles after the value is input. If Sketch is processing, ODACT* outputs a low during cycles in which valid data is output to OD, and a high otherwise.

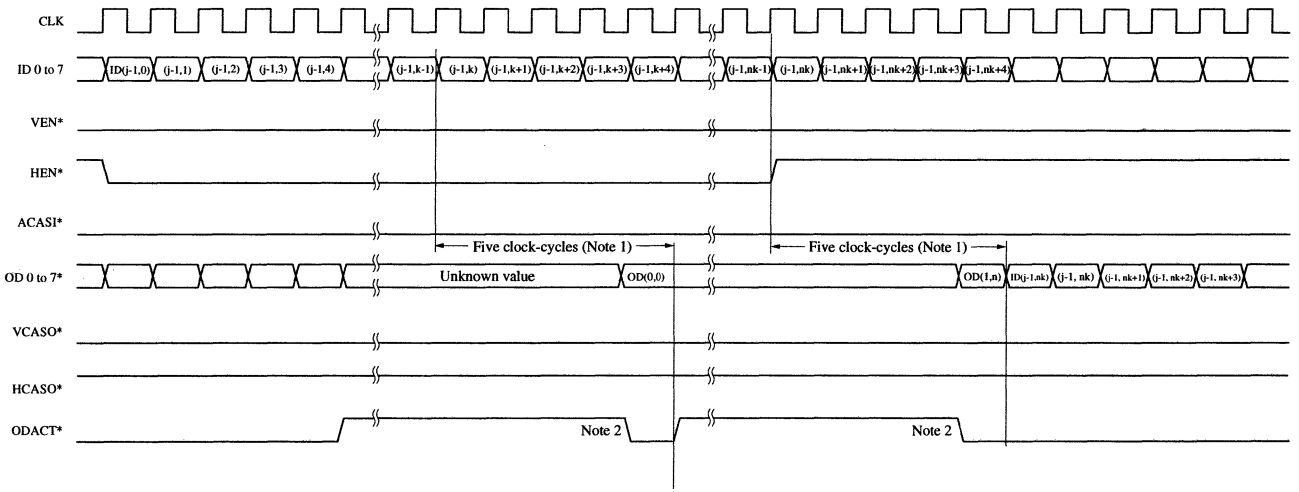
Note 4: If VEN* and HEN* both go low, Sketch immediately starts (or continues) average value reduction processing.

Note 5: If the number of pixels in the horizontal direction exceeds 512, Sketch suspends average value reduction processing, outputs data from ID to OD five clock cycles after the data is input, and pulls HCASO* low. If multiple Sketch devices are cascaded, the next Sketch stage is activated and starts processing.

Note 6: If HCASO* is low when HEN* goes high, HCASO* also goes high five clock cycles later.

Note 7: HEN* must be deasserted (kept inactive) for at least three clock cycles.

Note: Asterisks (*) after the signal names denote that these signals use negative logic (active low).



Note 1: If the local area size is j x k, $OD(0,0) = \sum_{\substack{x=0-j-1 \\ y=0-k-1}} ID(x, y)/2^d$ is output five clock cycles after ID(j-1, k-1) is input (d is the divisor set to the DIV register).

After the above, valid data for this line is output to OD every j clock cycles. The same operation is performed for all n x k-1th lines (n is a positive integer).

Note 2: During average value reduction processing, ODACT* is low during cycles in which valid data is output to OD, and is high otherwise.

Note: The asterisks (*) after the signal names denote that these signals use negative logic (active low).

Section 7: Electrical Characteristics

7.1 Absolute Maximum Ratings

(Referenced to GND, Ta = 25°C unless otherwise specified)

Parameter	Symbol	Rating	Unit
DC supply voltage	Vdd	-0.3 to 6.5	V
Input/Output voltage	V_{IN}	-0.3 to Vdd + 0.3	V
Operating temperature	T_{OPT}	0 to 70	°C
Storage temperature	T_{STG}	-10 to 80	°C

7.2 Recommended Operating Conditions (Ta = 0 to 70°C)

Parameter		Symbol	Condition	Min.	Typ.	Max.	Unit
Power-supply voltage		Vdd		4.75	5.0	5.25	V
Input voltage	Range	V_I	TTL level normal input	0		Vdd	V
	High level	V_{IH}		2.2		Vdd	V
	Low level	V_{IL}		0		0.8	V
Input rising time		T_{RI}	TTL level normal input	0		200	ns
Input falling time		T_{FI}		0		200	ns

7.3 Input/Output Capacitance

Parameter	Symbol	Condition	Typ.	Unit
Input capacitance	C_{IN}	Any input (Note 1)	10	pF
Output capacitance	C_{OUT}	Any output	10	pF

Note 1: Does not apply to bidirectional buffers.

7.4 DC Characteristics

Vdd = 5 V ± 5%
Ta = 0 to 70°C
GND = 0 V

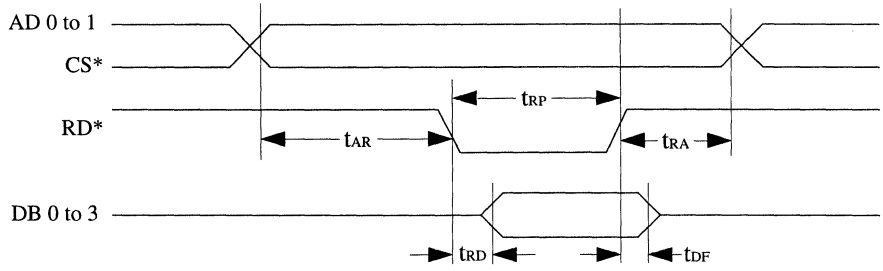
Symbol	Parameter		Condition	Min.	Typ.	Max.	Unit
V_{IL}	Low-level input voltage					0.8	V
V_{IH}	High-level voltage			2.0			V
I_{IN}	Input current	Without pull-up resistor Without pull-down resistor	V_{IN} = Vdd or GND	-10	±1	10	µA
		With pull-down resistor	V_{IN} = Vdd	35	110	335	µA
		With pull-up resistor	V_{IN} = GND	-35	-115	-350	µA
V_{OH}	High level output voltage		I_{OH} = -4 mA	2.4	4.5		V
V_{OL}	Low level output voltage		I_{OL} = 4 mA		0.2	0.4	V
I_{OZ}	Off-state leakage current		V_{OH} = Vdd or GND	-10	±1	10	µA
I_{OS}	Output short-circuit current (Note 1)		Vdd = Max,Vo=Vdd	15	50	130	mA
			Vdd = Max, Vo = 0 V	-5	-25	-100	mA

Note 1: I_{OS} designates current that flows out when the output is shorted to GND in a high state. However, simultaneous shorting of two or more outputs is not allowed. Shorting time is one second or less.

7.5 AC Characteristics

CPU Interface

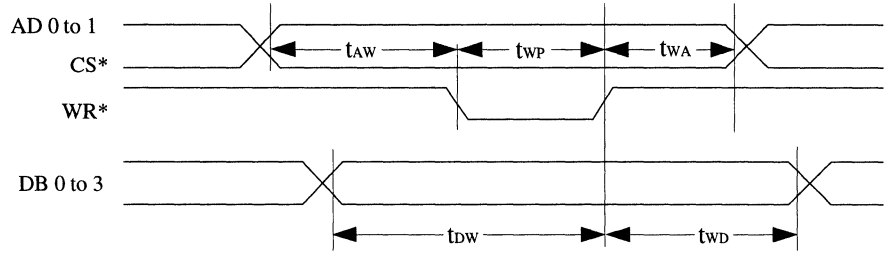
Read Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
AD and CS* setup times to RD* falling	t_{AR}	5	—	—
AD and CS* hold times to RD* rising	t_{RA}	0	—	—
RD* pulse width	t_{RP}	30	—	—
Delay time from RD* falling to DB enable	t_{RD}	5	—	20
Delay time from RD* rising to DB disabled (Hi-Z)	t_{DF}	3	—	15

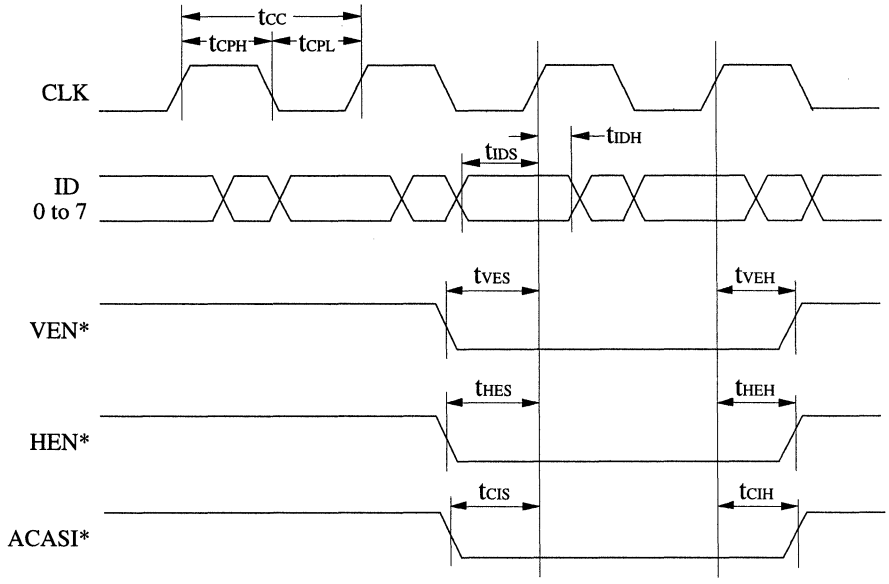
Write Cycle



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
AD and CS* setup times to WR* falling	t_{AW}	5	—	—
AD and CS* hold times to WR* rising	t_{WA}	0	—	—
WR* pulse width	t_{WP}	30	—	—
DB setup time to WR* rising	t_{DW}	15	—	—
DB hold time to WR* rising	t_{WD}	0	—	—

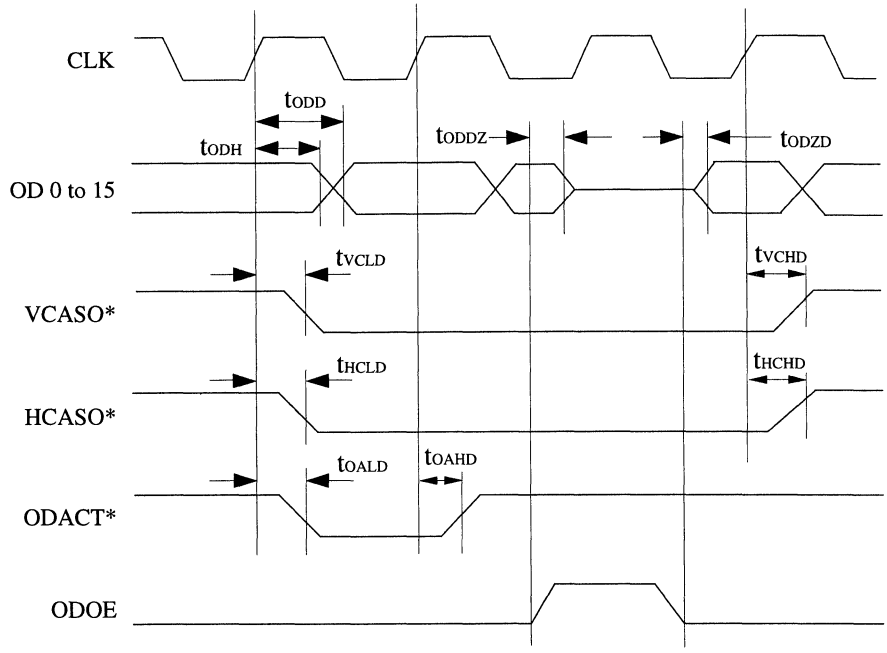
Image Data Input



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
CLK cycle	t_{CC}	25	—	—
High level period of CLK	t_{CPH}	10		
Low level period of CLK	t_{CPL}	10		
ID setup time to CLK rise	t_{IDS}	8	—	—
ID hold time to CLK rise	t_{IDH}	5	—	—
VEN* low setup time to CLK rise	t_{VES}	8	—	—
VEN* low hold time to CLK rise	t_{VEH}	5	—	—
HEN* low setup time to CLK rise	t_{HES}	8	—	—
HEN* low hold time to CLK rise	t_{HEH}	5	—	—
ACASI* low setup time to CLK rise	t_{CIS}	8		
ACASI* low hold time to CLK rise	t_{CIH}	5		

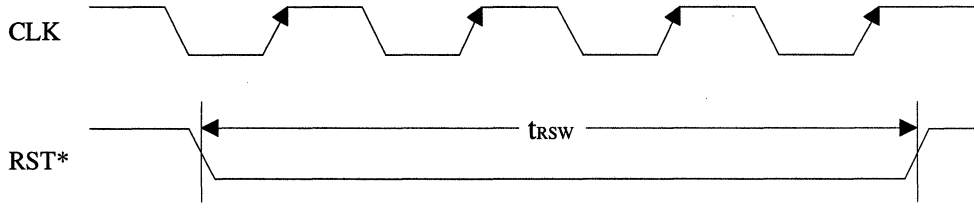
Image Data Output



unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Delay time from CLK rise to OD enabled	t_{ODD}	—	—	20
OD hold time to CLK rise	t_{ODH}	5	—	—
Delay time from CLK rise to VCASO* low	t_{VCLD}	—	—	20
Delay time from CLK rise to VCASO* high	t_{VCHD}	—	—	20
Delay time from CLK rise to HCASO* low	t_{HCLD}	—	—	20
Delay time from CLK rise to HCASO* high	t_{HCHD}	—	—	20
Delay time from CLK rise to ODACT* low	t_{OALD}	—	—	20
Delay time from CLK rise to ODACT* high	t_{OAHD}	—	—	20
Delay time from ODOE* rise to OD disabled (Hi-Z)	t_{ODDZ}	—	—	15
Delay time from ODOE* fall to OD enabled	t_{ODZD}	—	—	15

Reset Timing



The reset signal must be held active for at least 3 clock cycles.

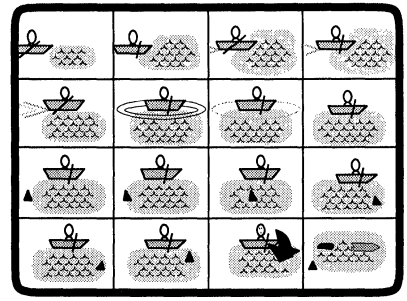
Unit [clock cycles]

Parameter	Symbol	Min.	Typ.	Max.
RST* pulse width (low period)	t_{RSW}	3	—	—

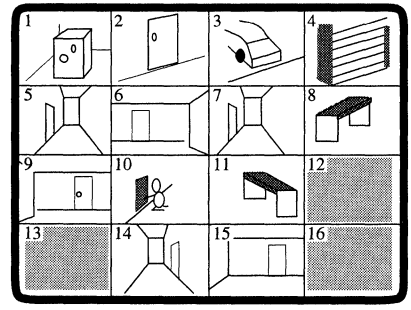
Section 8: Sample Applications

8.1 Applied Images

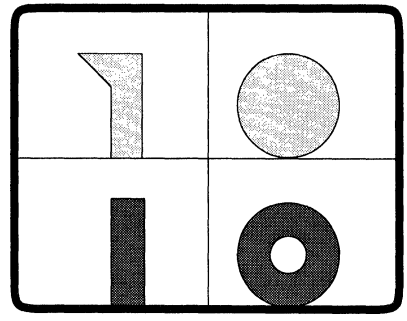
1. Multimedia: High-speed display similar to strobe photography.



2. Security: Simultaneously displaying a large number of monitoring camera shots on a screen.



3. FA: Comparing actual products with good products during an in-process inspection.



Products to be inspected

Good products or standard samples

Inspection spot 1 Inspection spot 2

8.2 Division Operation Using External ROM

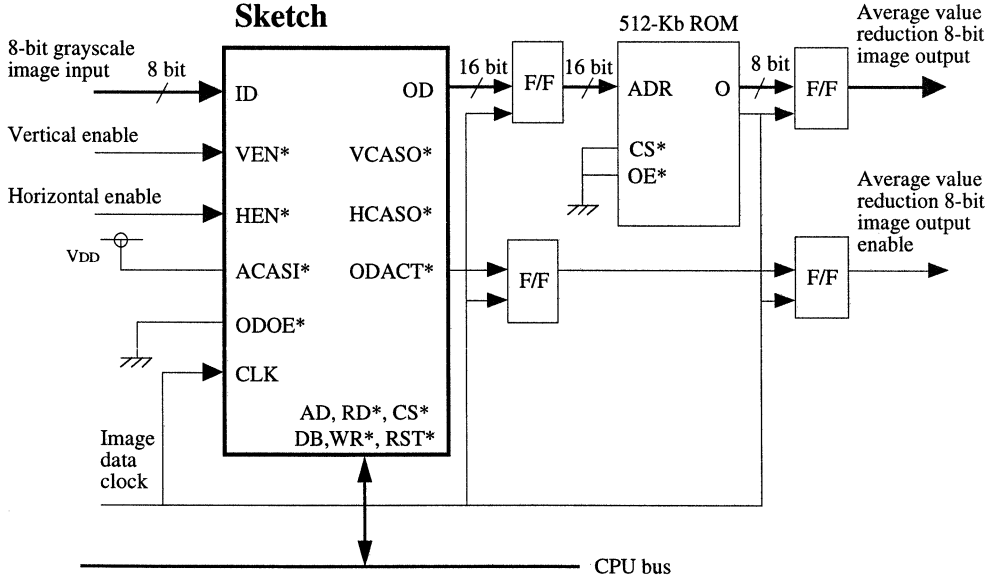
The Sketch chip allows the height and width of a local area to be independently set to any value from 1 to 16. For example, if the width of the local area is set to 5 (LH = 4) and the height to 3 (VL = 2), the number of pixels in this local area is 5 x 3 = 15. However, because average values are calculated using a shift operation, the divisor is limited to integral powers of 2. This means that in the above case, the device could not produce an exact average value reduction image.

For applications that can tolerate some degree of error and so do not require precise average value compressed image output, the divisor 2^4 = 16 (DIV = 4) could be used as an alternative in the above example.

If applications require an exact average value reduction image, use a ROM conversion table like the one shown below to perform the division operation.

In this case, the divisor is left set to 1 (DIV = 0) so that the OD pin outputs the sum of the gray level values of the pixels in the local area. This output is sent to a given ROM address, where it is divided by the specified divisor and output to ROM data. The ROM output data thus contains the results of dividing the address data by the specified divisor.

The disadvantage of this method is that the divisor is a fixed value. If multiple divisors are required, use a ROM of a capacity that can accommodate multiple conversion tables, and pins above A16 for selecting the divisor.

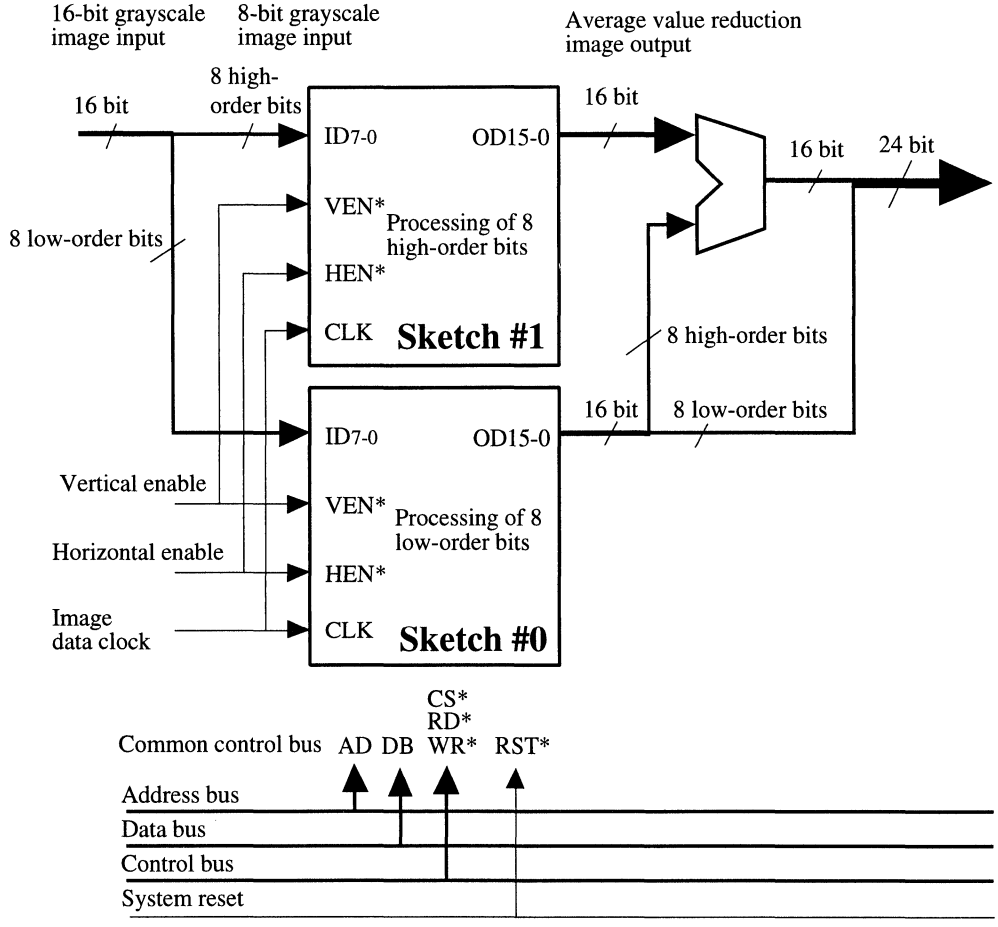


Note: An asterisk (*) after the signal name indicates inverse logic (active low).

8.3 Sample Configuration for Data Width Expansion (16 Bits)

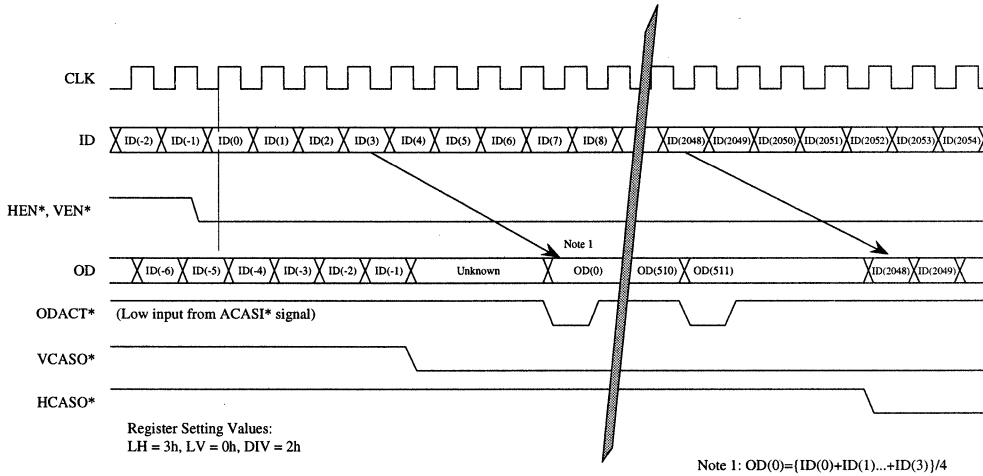
The diagram below shows an example of data width expansion using a parallel processing configuration. A single Sketch device limits the data width of the processed image to 256 gray levels (8 bits). The Sketch device can also output in 16 bits the sum of gray levels of an 8-bit input image in a unit area, without omitting any bit. This capability divides the image input into units of 8 bits, sums the gray levels in each unit using multiple Sketch devices, then uses an external adder to expand the required gray level portion to any desired data width. This provides greater flexibility in handling the complicated summing operation in unit areas by average value compression processing. Furthermore, a division calculation to obtain average values can then be done by dropping the required number of bits from the MSB portion of the data. ROM-based table conversion is recommended for division operations that use arbitrary divisors.

The figure below shows an example of data width doubling (to 16 bits) using two Sketch devices.



Note: An asterisk (*) after the signal name indicates inverse logic (active low).

8.4 Sample Extraction of Average Value Without Designating a Rectangular Processing Area



The above example shows signal timing in a situation in which the average value is extracted from four data groups input in sequence from the ID signal. Processing is executed using a local area setting of 1 x 4 pixels. By setting the local area to 1 x n pixels (where n is the value of register LH plus one, and register LV = 0), the processing area is viewed not as a rectangle but as a line, and the vertical and horizontal enable signals VEN* and HEN* no longer need to designate a rectangular area for processing. Thus, while both VEN* and HEN* are at low level, average values are extracted from every group of data input in sequence from the ID signal, the size of each group being determined by the value of the LH register plus one. The number of data values taken for averaging is then determined by the setting range of the LH register plus one, and so can range from 1 to 16.

Also, after both VEN* and HEN* go low, once the number of data values input from the ID signal exceeds 512 x n (where n = the value of the LH register plus one), the Sketch chip stops processing. To continue processing at this point, set VEN* and HEN* high again, then change them back to low to restart the process.

SMI ASSP

Image Processing LSI Series

IP90C18

Features Extracting Processor (Feature)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.1

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Features and Functions

1.1 Overview of Product Functions

The IP90C18 feature extraction ("Feature") chip can perform the following processes in real time. The processes performed are selected by register settings.

- Cumulative addition processing by area

  Weighted area measurement using 12-bit grayscale values (0–4095) for multiple areas (4096 or 2048 labeled areas).

- Histogram processing

  Grayscale histogram measurement (distribution of frequency of occurrence) of input in 12-bit grayscale resolution (values 0–4095). Uses the input value from cumulative grayscale addition processing as a constant for each area. Suitable for grayscale histogram measurement in applications requiring high precision, such as medical imaging devices. Also performs pixel count (area) measurement of up to 4096 separate labeled areas using 12-bit labeled image input.

- Diagonal coordinate measurement of circumscribed rectangles

  Measures the diagonal coordinates of rectangles circumscribed around all labeled areas (up to 4096 or 2048 labeled areas).

- Perimeter measurement

  Measures the perimeter for each of up to 2047 labeled areas in a labeled image, in vertical, horizontal, and diagonal directions. Suitable for evaluating complexity of shapes, roundness, etc. Also operates in extended perimeter measurement mode, in which it measures only one of the three elements for up to 4095 labeled areas. Functions with either 8-connection or 4-connection labeling logic.

- Area boundary one-point coordinate measurement

  Measures the coordinates of the first pixel input on the boundary of a given area in a raster scan image. Suitable for measuring starting points for image-processing functions that use software for boundary tracking.

- Run coordinate detection

  Detects the coordinates of each end of a run, and outputs these coordinate along with signals indicating the start and end points of runs in the direction of the raster scan.

- Primary moment measurement

  Calculates $\Sigma h \cdot d(h, v)$ or $\Sigma v \cdot d(h, v)$ for each of up to 2048 labeled areas (where $d(h, v) = 0$ or 1, and all areas can be handled as binary images). These results, together with the above histogram measurement (surface area measurement), can be used as input for an external CPU to make center-of-gravity calculations for each labeled area.

- Secondary moment measurement

  Calculates $\Sigma h \cdot h \cdot d(h, v)$, $\Sigma h \cdot v \cdot d(h, v)$, or $\Sigma v \cdot v \cdot d\ (h, v)$ for each of up to 2048 labeled areas (where $d(h, v) = 0$ or 1, and all areas can be treated as binary images). These results, together with the above histogram measurement (surface area measurement) and primary moment measurement functions, can be used as input to an external CPU to calculate the angle of the principal axis of each labeled area.

- H-axis, V-axis grayscale projection processing

  12-bit grayscale projection processing (perimeter distribution measurement) on large areas up to 2048 x 2048 pixels for two-axis simultaneous processing, or 4096 x 4096 pixels for one-axis processing.

- Cumulative grayscale histogram processing

  After grayscale histogram processing, converts the histogram results into cumulative histogram data.

- Table conversion (LUT: Look-up table) function

  The internal high-speed memory can perform 12-bit table conversions of area numbers or grayscale data values.

The table on the following page lists the quantitative measurements the IP90C18 chip can make from grayscale or labeled image data.

Quantitative Features and Measurement Algorithms

Feature		Symbol	Value measured, extraction method, etc.
Grayscale value histogram		H	Measures how often each grayscale value occurs.
Surface area measurement (zero-degree moment)		S	Counts the number of pixels of each area (label) number. Histogram processing indicates the size of each area.
Grayscale-weighted surface area measurement		Sw	Calculates the total of grayscale values of pixels for each area (label) number. Cumulative addition of grayscale values for each area. $\Sigma f(h, v)$: f(h, v) represents the grayscale value at coordinates (h, v).
Diagonal coordinate measurement of circumscribed rectangle (Ferrer axis)	Lower left	H0, V0	Measures the coordinates of the lower left and upper right vertices of the diagonal of an external rectangle circumscribed around each area.
	Upper right	H1, V1	Rect_zone {Ln,(H0, V0), (H1, V1)} Lower left coordinates (H0, V0) and upper right coordinates (H1, V1) with respect to label Ln.
Perimeter		Hperi, Vperi, Dperi	Perimeter is variously defined: here it is considered a complement of aspect ratios, and is counted in terms of vertical, horizontal, and diagonal distances (number of steps between pixels). After measurement, pixels in the diagonal direction (when the aspect ratio is 1) are assigned a weight of $\sqrt{2}$ (processed outside the IP90C18 chip). Requires simultaneous input of two lines.
One-point boundary coordinate measurement		Ha, Va	Measures the coordinates (Ha, Va) of the first pixel input in the raster scan process for each area.
Run coordinate detection		Hn, Vn	For each pixel on the boundary (perimeter) of an area, and outputs a signal indicating the coordinates (Hn, Vn) of the pixel and the start and end points of runs in the direction of the raster scan.
Primary moment		Mh, Mv	$\Sigma h \cdot d(h, v)$: d(h, v) = 0 or 1 (1 if moment exists). $\Sigma v \cdot d(h, v)$: Primary horizontal or vertical moment of each area.
(Center)		Hg, Vg	Mh/S, Mv/S: Primary moment divided by surface area (external processing).
Secondary moment		Mhh, Mhv, Mvv	$\Sigma h \cdot h \cdot d(h, v)$, $\Sigma h \cdot v \cdot d(h, v)$, $\Sigma v \cdot v \cdot d(h, v)$: d(h, v) = 0 or 1 (1 if moment exists). Secondary moment of each area HH, HV, or VV.
Primary axis angle		θ	Angle between primary axis and horizontal. Calculated from the 0-order, primary, and secondary moments (by external processing).
Grayscale projection processing		Hproj, Vproj	Projection processing of each pixel on the H-axis and V-axis (perimeter distribution). Calculates the total of the grayscale values of all pixels in each row and column.
Table conversion		LUT	Look-up table (LUT) conversion using internal results memory.
Cumulative grayscale histogram processing		Hacc	Conversion of histogram results to cumulative histogram data.

1.2 Features and General Specifications

The IP90C18 Feature chip is designed to extract features from images. This section lists the types of feature extraction processing that can be achieved in real time at a maximum operating frequency of 40 MHz, and stored in the chip's internal results memory.

◊ Maximum operating frequency (image data input rate)

- IP90C18-HS: f_{max} = 40 MHz (Note 1)
- IP90C18: f_{max} = 20 MHz

Note 1: Because of constraints on power consumption, use the IP90C18-HS at 25 MHz or below only when performing projection processing on the H-axis or when operating in 2K area measurement mode.

◊ On-chip results memory

Processing results can be accessed from the IP90C18 Feature chip at random, just as if it were an SRAM with 4096 x 24-bit configuration.

1.2.1 Overview of Image-Processing Functions

The Feature chip performs many types of image-measurement functions. Each of these functions can be performed on input images having up to 4096 gradients or separate areas, and up to 4096 x 4096 pixels in size. Different measurement functions can be performed sequentially by using mode switching.

For example, the IP90C18 would calculate the zero-degree and primary moment for each of 4096 areas in three steps: It first measures the zero-degree moment for each of 4,096 areas, then it measures the primary moment for areas 0–2047, and finally it measures the primary moment for areas 2048-4095.

The following section presents an overview of all the IP90C18's measurement functions. For details, see Section 2, "Measurement Processing Algorithms."

◊ Cumulative Processing by Image Area (weighted average surface area measurement)

One of the basic functions of the Feature chip, this process involves calculating the total of the grayscale values for each area, using area number and grayscale value input data. This function is used in weighted average surface area measurement for individual areas.

[4K Area Measurement Mode]

- Maximum number of selected areas: 4096 (area numbers 0–4095)
- Grayscale image input (values to be added): 12-bit
- Width per area: 24-bit

  When the maximum number of areas selected is 4096, the width per area is 24 bits.
- Maximum processed image size: Limits only pixel count per area

  In 4K area measurement mode, the maximum number of pixels per area in worst-case conditions with 12-bit grayscale input is $2^{12} + 1 = 4097$ pixels; with 8-bit grayscale input the maximum is $(2^{24})/(2^{8}-1)$ pixels. The principles applied in measurement processing limit overall pixel count, though the screen aspect ratio and shape are not restricted.

[2K Area Measurement Mode]

- Maximum number of selected areas: 2048 (area numbers 0–2047)
- Grayscale image input (values to be added): 12-bit
- Width per area: 48-bit

  When the maximum number of areas selected is 2048, the width per area is 48 bits.
- Maximum processed image size: Limits only pixel count per area

  In 2K area measurement mode, the maximum number of pixels per area in worst-case conditions with 12-bit grayscale input is $(2^{48})/(2^{12}-1)$ pixels; with 8-bit grayscale input the maximum is $(2^{48})/(2^{8}-1)$ pixels. The principles applied in measurement processing limit overall pixel count, though the screen aspect ratio and shape are not restricted.

◊ Histogram Processing Functions (surface area measurement [zero-degree moment], and grayscale value frequency-of-occurrence, by area)

This function takes the area number or grayscale input values in the input image as constants, and measures surface area by counting how often each area number appears, or uses grayscale values to calculate the frequency-of-occurrence distribution of each grayscale value (grayscale histogram processing). The basic process is the same as in cumulative processing by image area. This histogram measurement process is a preliminary step to cumulative histogram processing, described in Section 1.2.2, "Peripheral Functions."

[4K Area Measurement Mode]

- Input data width (grayscale or label values): 12 bits (gradients or areas 0–4095)
- Maximum pixel processing capacity: 2^{24}-1 pixels (approx. 16.7 M pixels)

  The principles applied in measurement processing do not restrict the screen aspect ratio or shape (though they do limit overall pixel count). Maximum screen sizes of 4096 x 4096 pixels, 16K x 1K pixels, or 64K x 256 pixels can be processed without overflow conditions.

[2K Area Measurement Mode]

- Input data width (grayscale or label values): 11 bits (gradients or areas 0–2047)
- Maximum pixel processing capacity: 2^{48}-1 pixels (approx. 281 T pixels)

  The principles applied in measurement processing do not restrict the screen aspect ratio or shape (though they do limit overall pixel count). Maximum screen sizes of 16M x 16M pixels or 256M x 1M pixels can be processed without overflow conditions.

◊ Circumscribed Rectangle Diagonal Coordinate Measurement Function

This function measures the coordinates of the diagonal (lower left and upper right) vertices of a rectangle circumscribed around each individual area.

[Lower left, upper right coordinate simultaneous measurement: 2K area measurement mode]

- Number of areas (label values): 2048 (0–2047)
- Maximum screen size: 4096 x 4096 pixels (12-bit coordinate values)

[Lower left, upper right coordinate simultaneous measurement: 4K area measurement mode]

- Number of areas (label values): 4096 (0–4095)
- Maximum screen size: 4096 pixels wide, and 4096 pixels high (12-bit coordinate values)

◊ Perimeter Measurement Function

Measures the perimeter of each area. Each area is viewed as divided into its horizontal, vertical, and diagonal components, which are counted and converted from aspect ratio measurement to perimeter measurement. This function has four modes, including simultaneous perimeter measurement in all three component directions, and in each of the directions separately. Each measurement mode can be applied to labeled images using 8-way or 4-way connection labeling.

[Three-way simultaneous measurement mode: 2K area measurement mode]

- Number of areas (label values): 2047 (1–2047, excluding background value 0)
- Maximum length count limits: 2^{16}-1 pixels in each dimension (horizontal, vertical, diagonal)

[One-way measurement mode: 4K area measurement mode]

- Number of areas (label values): 4095 (1–4095, excluding background value 0)
- Maximum length count limits: 2^{24}-1 pixels in each dimension (horizontal, vertical, diagonal)

◊ Area Boundary One-Point Coordinate Measurement Function

In a raster-scanned input image, this function recognizes the first coordinates input for each area as the edge of that area (the boundary pixel) and stores them in memory. This information can then be used as the starting point by a boundary tracing algorithm to measure the coordinates of the edges of any given area.

- Number of areas (label values): 4096 (0–4095)
- Maximum screen size: 4096 x 4096 pixels (12-bit coordinate values)

◊ Run Coordinate Detection Function

Detects both ends of a run of the same area values, and outputs a signal indicating the starting and ending points of runs in the direction of the raster scan signal (along with coordinate values) over the OD0–OD23 pins.

- Number of areas (label values): 4096 (0–4095)
- Maximum screen size: 12 bits each for h and v coordinates (4096 x 4096 pixels)

◊ Primary Moment Measurement Function

The results of primary moment measurement can be combined with histogram measurement (surface area or zero-order moment measurement) to allow an external CPU to calculate centers of gravity for each labeled area.

[2K Area Measurement Mode]

- Maximum number of areas (label values): 2048 (0–2047)
- Maximum processing screen size: 4096 x 4096 pixels (when the maximum number of areas is 2048)
- Measurements made for each area: $Mh = \Sigma h \bullet d(h, v)$ or $Mv = \Sigma v \bullet d(h, v)$, where $d(h, v) = 0$ or 1

[4K Area Measurement Mode]

- Maximum number of areas (label values): 4096 (0–4095)
- Maximum processing screen size: 256 x 256 pixels
- Measurements made for each area: $Mh = \Sigma h \bullet d(h, v)$ or $Mv = \Sigma v \bullet d(h, v)$, where $d(h, v) = 0$ or 1

◊ Secondary Moment Measurement Function

The results of secondary moment measurement can be combined with histogram measurement (surface area or zero-order moment measurement) and primary measurement to allow an external CPU to calculate primary axis angles (the angle formed between the primary axis and horizontal) for each labeled area.

[2K Area Measurement Mode]

- Maximum number of areas (label values): 2048 (0–2047) or 4096 (0–4095)
- Maximum processing screen size: 4096 x 4096 pixels (when the maximum number of areas is 2048)
- Measurements made for each area: $Mhh = \Sigma h \cdot h \cdot d(h, v)$, $Mhv = \Sigma h \cdot v \cdot d(h, v)$, or $Mvv = \Sigma v \cdot v \cdot d(v, v)$, where $d(h, v) = 0$ or 1

[4K Area Measurement Mode]

- Maximum number of areas (label values): 4096 (0–4095)
- Maximum processing screen size: 64 x 64 pixels (when the maximum number of areas is 4096)
- Measurements made for each area: $Mhh = \Sigma h \cdot h \cdot d(h, v)$, $Mhv = \Sigma h \cdot v \cdot d(h, v)$, or $Mvv = \Sigma v \cdot v \cdot d(v, v)$, where $d(h, v) = 0$ or 1

◊ Grayscale Projection Processing Function (Perimeter Distribution Measurement)

This function performs grayscale value projection processing (perimeter distribution measurement) on the H- and V-axes simultaneously, or on either axis separately.

[Two-axis simultaneous projection measurement mode]

This mode performs grayscale projection processing simultaneously on the X- and Y-axes. Half of results memory (2048 words each) is used to store results of processing on each axis.

- Maximum processing screen size: 2048 x 2048 pixels

  If the grayscale input uses the maximum bus width of 12 bits (4096 gradient values), the maximum number of pixels (depth) in the summation process becomes $(2^{24}-1)/(2^{12}-1) = 2^{12}+1$, or 4097 pixels. As a result, input image sizes up to 2048 x 2048 pixels can be used without overflow.

[One-axis projection mode (4K area measurement mode)]

This mode performs grayscale projection on the H-axis or V-axis. Because the entire 96 Kbits of results memory can be used to store the results of projection processing on one axis, the 4096 words x 24 bits/word configuration of results memory can be used to increase the measurement capability up to 4096 pixels on one axis (either axis of an image of up to 4096 x 4096 pixels).

- Maximum processing screen size: 4096 x 4096 pixels

  If the grayscale input uses the maximum bus width of 12 bits (4096 gradient values), the maximum number of pixels (depth) in the summation process becomes $(2^{24}-1)/(2^{12}-1) = 2^{12}+1$, or 4097 pixels. As a result, input image sizes up to 4096 x 4096 pixels can be used without overflow. In addition, with the commonly used 8-bit (256-gradient value) grayscale input, the maximum number of pixels (depth) is increased to $(2^{24}-1)/(2^{8}-1) > 2^{16}$, or a maximum of 65K pixels. In this case, therefore, the input image size can be up to 4096 pixels (measurement direction length or projection axis length) x 65K pixels (projection pixel count or projection direction length) without overflow.

[One-axis projection mode (2K area measurement mode)]

This mode performs grayscale projection on the H-axis or V-axis. The entire 96 Kbits of results memory can be used to store the results of projection processing on one axis. By using a results memory configuration of 2048 words x 48 bits/word, measurement capability is extended to projection values of up to 48 bits for each of 2048 pixel values.

- Maximum processing screen size: 2048 pixels in the direction measured

  If the grayscale input uses the maximum bus width of 12 bits (4096 gradient values), the maximum number of pixels (depth) in the summation process is only limited by the condition $(2^{48}-1)/(2^{12}-1) > 2^{36}$ pixels, and therefore the maximum is on the order of 64G pixels. In addition, with the commonly used 8-bit (256-gradient value) grayscale input, the maximum number of pixels (depth) is increased to $(2^{48}-1)/(2^{8}-1) > 2^{40}$, or a maximum of 1 trillion pixels, enough to prevent overflow in virtually any application. Note that at 40 MHz the amount of time required to input 2048 x 1T pixels is $(2 \times 10^{3} \times 10^{12})/(40 \times 10^{6}) = 5 \times 10^{7}$ seconds, which is equivalent to 575 days, or roughly
  1–1/2 years of continuous measurement.

- Maximum operating frequency limitations

  To prevent excessive power use and resulting high temperatures from affecting reliability, keep the maximum operating frequency at or below 25 MHz when using one-axis projection mode with 2K area measurement mode and projection on the H-axis only.

◊ Table Conversion (Look-Up Table) Function

The IP90C18 chip's internal memory (in 4096 x 24 bits/word configuration) can be used for look-up table (LUT) conversion. The grayscale input signal pins (ID0–ID11) or the area number input signal pins (L0–L11) can be used to identify addresses in results memory, and the read-out data can be output from the image output signal pins (OD0–OD23). This configuration can also be used to dump the contents of results memory.

1. First, values previously assigned to corresponding 12-bit (4096-value) input values are written from the CPU interface (DB0–DB23). This enables the values stored in the corresponding addresses to be output from the image data output pins (OD0–OD23). This function is used in compensation of grayscale input values, conversion of area (label) numbers, and computations involving one-to-one input/output functions (square root, 2/3, shift calculation, code extension, etc.).
2. Following the end of a separate calculation process, the values stored in results memory can be dumped as output from the image data output pins (OD0–OD23). Alternatively, the results can be used as they are as conversion coefficients in a conversion table: this could provide a memory table of compensation coefficients for use with line sensor data.

- Input data (grayscale or label values) width: 12 bits (0–4095 gradients or area numbers)
- Output data width: 24 bits (16.7 M values)

1.2.2 Peripheral Functions

In addition to the functions described in Section 1.2.1, "Overview of Image-Processing Functions," the IP90C18 Feature chip provides a number of peripheral functions designed to facilitate image-processing measurements. This section describes these peripheral functions.

◊ Cumulative Histogram Processing Function

- The internal addresses of results memory are connected to the internal counter (up/down counter), and selected sequentially.
- The data from each selected address is added to the data from that address on subsequent cycles, so that data values are cumulatively increased at each address. At the end of the process, the contents of results memory represent a cumulative frequency-of-occurrence distribution pattern.
- The range of cumulative addition is:
  - 4K area measurement mode: 0–4095
  - 2K area measurement mode: 0–2047
- Automatic execution from histogram measurement

  Automatic cumulative histogram measurement can begin automatically when histogram measurement processing is complete.
- Power-saving cumulative histogram mode

  If 2K area measurement mode is used at clock (iCLK) frequencies greater than 25 MHz, excessive power use and the resulting high temperatures can damage the chip. To avoid this, write 1 to the power-saving cumulative histogram mode flag in the mode register (MOD): this flag reduces processing speed by half, thereby reducing power consumption by approximately half and preventing damage to the chip.

◊ Processing Field Count Control Function

Processing field count controls are designed to automatically stop processing after a designated number of fields (input screens) have been processed. This function uses an 8-bit counter, and can designate any number of fields from 1 to 255.

This function can be set to stop automatically after n fields (n = 1 to 255) are measured, or to allow continuous execution until 0 is written to the exec flag.

◊ Pixel-by-Pixel Data Enable Function (Pixel Control)

The data enable control pin (IDEN*) can control whether each individual pixel is included in measurement. Because the coordinate counter does not stop, this function can extend the data width of the area measured. For details, see Section 4.3, "Sample Configuration: Extended Systems."

◊ One-Shot Memory Clear Function

Results memory must always be cleared before executing the next measurement. The IP90C18 chip can clear all bits in results memory at once (in 1 μs) by means of a setting in the memory clear register.

◊ Background Designation Function

Perimeter measurement and run coordinate detection differ from other feature extraction processes in that they require a background area number to be designated. Area boundaries are recognized by using the number 0 to represent background area.

The data enable control signals (IDEN* and DIDEN*) can designate background area on a pixel-by-pixel basis. IDEN* designates background for signal pins L<11..0>, and DIDEN* designates background for signal pins DL<11..0>.

◊ Overflow Flag Function for Cumulative Addition by Area, or for Grayscale Histogram Values

A selection of register settings is available to use either the MSB of the results memory data value or an overflow flag. If the overflow flag function is used, the width of the data in results memory is decreased by 1 bit.

◊ Vertical Enable Signal (VEN*) Enabled by the FEN* Function

This signal serves as an internal enable/disable control for the vertical enable signal (VEN*) on a field-by-field basis, synchronizes multiple processing circuits (LSIs) with the field signal, and selects the field in which to start processing.

◊ External Interface

- Allows status output to the CPU:
  1) Busy status output (BUSY*)
  2) Overflow output (OVF*)

  Cumulative addition, area overflow.
- Can be used to mask the overflow signal (OVF*) from any or all overflow sources (use with CPU-side interrupt control).
- Allows the CPU to verify execution status, overflow source, etc. by reading the appropriate registers.
- Allows direct connection to an 8/16/24(32)-bit CPU bus.

  Note that because the Feature chip has a bus width of 24 bits, connection to a 32-bit bus interface requires that the highest 8 bits be pulled up or otherwise protected on the external interface side.
- Input/output interface level: TTL level compatible

◊ Low Power Demand

CMOS process technology

◊ Power Supply

+5V single source

◊ Package

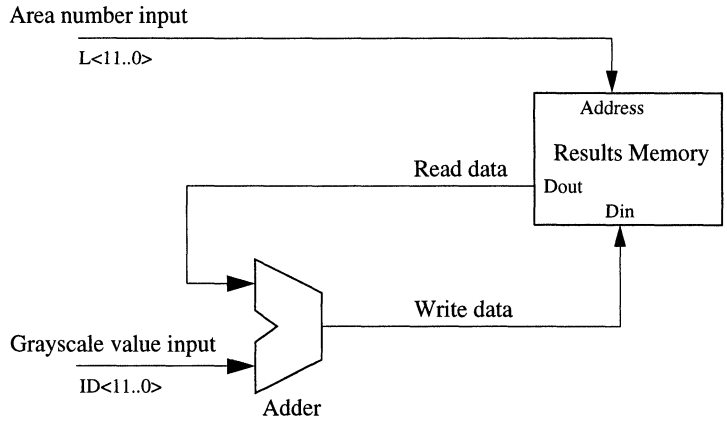
160-pin plastic QFP

Section 2: Measurement Processing Algorithms

2.1 Cumulative Summation by Area (Grayscale-Weighted Surface Area Measurement)

- This function uses number input (L<11..0>) to measure the cumulative sum of grayscale data values (ID<11..0>) for each designated area (cumulative addition processing). The result is the zero-degree moment of an area expressed in grayscale values.
- Performing cumulative summation of values for each area makes it possible to derive a grayscale-weighted surface area measurement.
- Summation by area is the Feature chip's most basic function. The first step in this process is to clear all values in results memory to 0 using the memory clear function. The chip then reads the value at each memory address designated by the area number input signal L<11..0>, adds the grayscale data input value ID<11..0>, and stores the result at the same address. This process is illustrated below.

Note: The diagram below is intended only as a basic illustration of the process described above, and does not accurately represent all operations or timing of the Feature chip. For details about functions or timing, see the appropriate sections of this manual.



2.1.1 Basic Applications of Summation by Area

- Calculates the sum of grayscale values for each pixel in any given area. This provides basic data for measuring the surface area of objects, as well as determining the location of an object.
- Example 1: Starting with an image that has been labeled by linked component labeling, the input data (which consists of image area numbers and corresponding grayscale values) is used to calculate the sum of grayscale values for each separate area number.
  - This enables the grayscale-weighted surface area to be calculated for each area.
- Example 2: Area numbers are read sequentially from values previously stored in frame memory, and the sum of grayscale values is calculated for each area number.
  - This enables simple pattern matching processes.

Designated area within frame memory
Area Parameter Table (APT)

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	5	5	5	5	5	5	5	5	5	5	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	5	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Input grayscale values

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	15	66	66	66	0	0	0	0	0	0	0	0
0	0	0	0	0	0	66	66	66	0	0	0	0	3	0	0	0
0	0	0	0	1	0	66	66	66	0	0	0	0	0	0	0	0
0	0	0	0	0	0	66	66	66	66	66	66	66	0	0	0	0
0	0	0	0	0	0	66	66	66	66	66	66	66	0	0	0	0
0	0	0	0	0	0	66	66	66	66	66	66	66	0	0	0	0
0	2	0	0	0	0	66	66	66	66	0	0	0	0	0	0	0
0	0	0	0	0	0	66	66	66	66	0	0	0	0	0	0	0
0	0	0	0	0	0	66	66	66	38	0	0	0	0	0	0	0
0	0	0	0	8	0	0	0	0	0	0	0	0	9	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

area 1:

0	0	0	0	66	66	66	0

sum = 198

area 5:

					66				
					66				
0	0	66	66	66	66	66	66	66	0
					66				
					66				
					38				
					0				
					0				

sum = 764

2.2 Pixel Count Measurement (Histogram Measurement)

Histogram measurement processing is an application of the Feature chip's most fundamental process: summation by area. Figure A below illustrates how the Feature chip measures the frequency of occurrence of grayscale values (histogram measurement). The first step is to clear all values in results memory to 0 using the memory clear function. The chip then reads the value at each memory address designated by the grayscale value input signal ID<11..0>, adds a constant (normally 1), and stores the result at the same address. Grayscale histogram measurement can be performed by processing a single screen of grayscale image input.

Measuring pixel count (surface area) by area can also be thought of as a frequency-of-occurrence distribution of area numbers. This process is illustrated in Figure B, below. Again, the first step is to clear results memory using the memory clear function, then to read the value at each memory address designated by the area number input signal L<11..0>, add a constant (normally 1), and store the result at the same address. Measuring pixel count by area can be performed by processing a single screen of labeled (area number) image input.

Note: The diagrams below are intended only as basic illustrations of the processes described above, and do not accurately represent all operations or timing of the Feature chip. For details about functions or timing, see the appropriate sections of this manual.

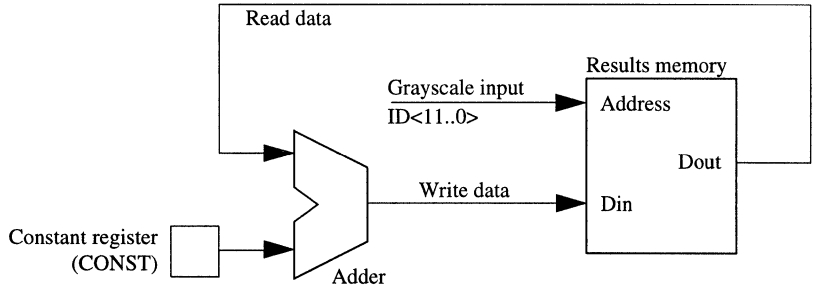


Figure A: Grayscale Histogram Measurement

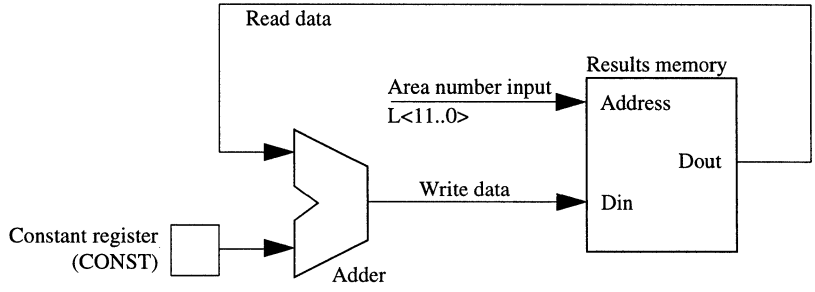
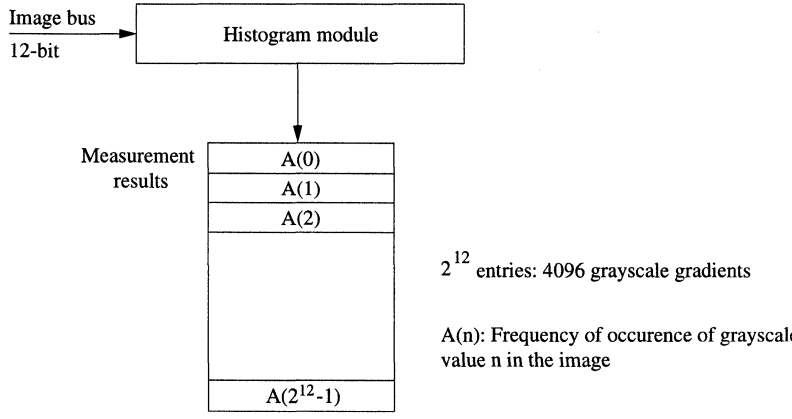


Figure B: Labeled Image Surface Area Measurement

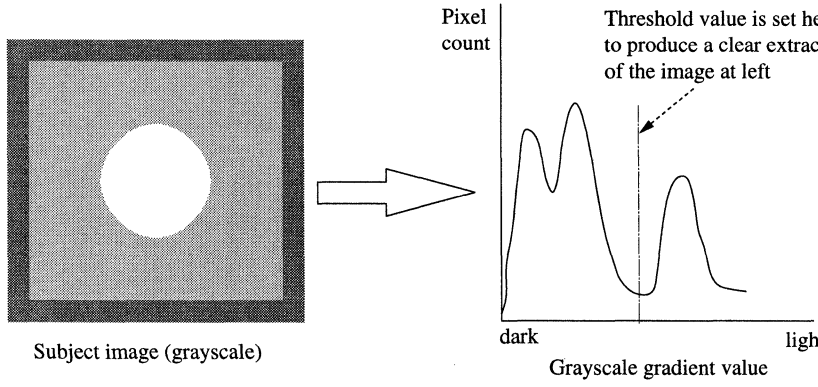
2.2.1 Grayscale Histogram Processing

The Feature chip's grayscale histogram process can measure how often 4096 separate grayscale values occur on an input screen. The chip measures grayscale values input through the grayscale value input signal ID<11..0>. By changing register settings, this process can also count values input on the L<11..0> input signal.



<Sample applications>

The grayscale histogram process can quantify the grayscale value composition of images for such applications as image-quality compensation or enhancement, or for binary conversion.



2.2.2 Histogram Processing of Area Numbers

The Feature chip can calculate how often each of 4096 separate area numbers occurs in an input screen. The chip measures area numbers input through the area number input signal L<11..0>. Counting area number values for each numbered area can be considered equivalent to measuring its surface area. This is also the same process used to calculate the zero-degree moment of an area. By changing register settings, this process can also count values input on the ID<11..0> input signal.

Histogram Processing Example

Input area number image
(labeled image)

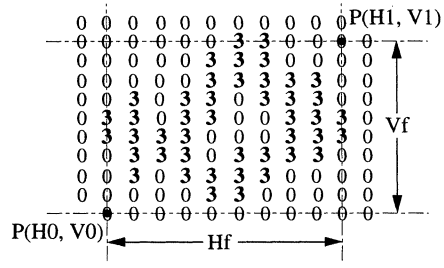
0	**5**	0	**5**	0	0	0	0	0	0
0	0	**5**	**5**	0	0	0	**9**	0	0
0	0	0	**5**	0	0	0	**9**	0	0
0	0	0	0	0	0	**9**	**0**	0	0
0	**11**	0	0	0	0	**9**	**9**	0	0
0	**11**	0	0	0	0	0	0	0	0

Result of measurement

Area no.	Pixel count
0	48
⋮	⋮
5	5
6	0
7	0
8	0
9	5
10	0
11	2
⋮	⋮
4095	0

2.3 Measurement of Diagonal Axis of a Circumscribed Rectangle (Feret's Diameter)

This process can measure specific areas existing within a given area within an image, calculate the approximate location of a given area, and obtain basic data about specific areas of interest (AOIs) within an image. The process circumscribes a rectangle around each given area, measures the coordinates of the lower left and upper right corners of the rectangle, and stores the results in memory. Each pixel within the area can then be identified as a pair of horizontal and vertical coordinate values. The process thus gives four values for each area measured: P(H0, V0), P(H1, V1), Hf, and Vf.



This process measures the coordinates of the lower left corner P(H0, V0) and upper right corner P(H1, V1) of the rectangle around each area. The horizontal Ferrer axis (Hf) and vertical Ferrer axis (Vf) are computed separately using these diagonal coordinates.

This process can measure the coordinates of the diagonal vertices of the circumscribed rectangle for each area in the image in two ways: measuring the coordinates of the lower left and upper right vertices simultaneously, or measuring the coordinates of either vertex by itself.

[Measuring lower left and upper right coordinates simultaneously: 2K area measurement mode]

Simultaneously measures the coordinates of the lower left and upper right vertices of the circumscribed rectangle about each area.

- Number of areas (label values): 2048 (0–2047)
- Maximum size of area measured: 4096 x 4096 pixels (coordinate values to 12-bit width)

[Measuring the lower left or upper right coordinates alone: 4K area measurement mode]

Measures the coordinates of either the lower left or upper right vertex of the circumscribed rectangle about each area.

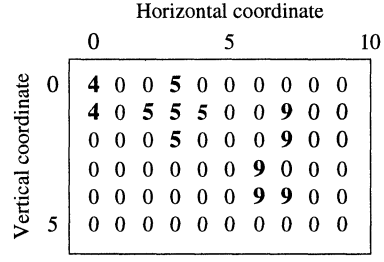
- Number of areas (label values): 4096 (0–4095)
- Maximum size of area measured: Either height or width of 4096 pixels (coordinate values to 12-bit width)

2.4 Area Boundary One-Point Coordinate Measurement

This process measures the coordinates of pixels that fall on the boundary between background and area data values of each area. Using a raster-scan input image, the process measures the coordinates of the first pixel input with each given area number (label value), and writes the results to memory. In addition, the process stores the area number of the pixel in the origin (0, 0) (at the upper left corner of the input image) in the origin area number register (LBL00).

By using this process to detect the coordinates of one pixel on the boundary of each area, the chip obtains a starting point for CPU processing using boundary tracking algorithms without having to scan the image. Image processing involves a wide variety of processing algorithms, using a different algorithm for practically every application. When the CPU or DSP accesses frame memory to search an image, such a load is placed on the system that nearly all the time required for image processing is consumed in accessing memory. The Feature chip reduces this load by offering functions such as the area boundary one-point coordinate measurement described here, the measurement of the diagonal axis of circumscribed rectangles described previously, and the run coordinate detection function described later. These functions reduce the load on the CPU or DSP, and enable the CPU to focus on the essential problems of image recognition and decision processing.

Example: Input area number image (labeled image)



One-point coordinate measurement by area (area start point coordinates)

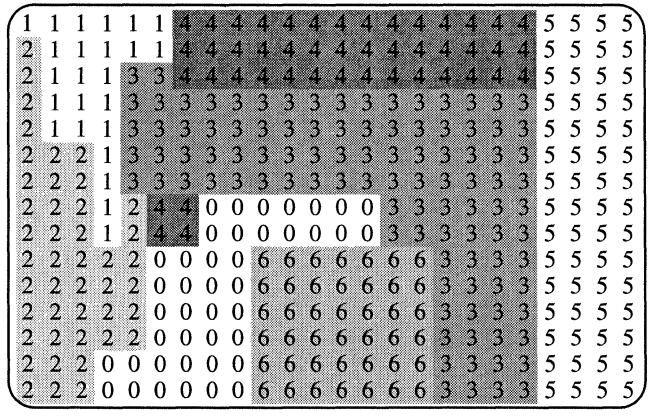
Area no.	Measurement result: start point coordinates	
	H	V
4	0	0
5	3	0
9	7	1

Value in origin area number register (LBL00): 04h

2.4.1 Images Subject to Diagonal Axis of Circumscribed Rectangle, and Area Boundary One-Point Measurement

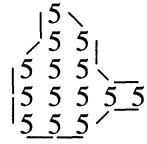
The following example illustrates the type of filled-in image that can be used in processes such as measuring the diagonal axis of a circumscribed rectangle, or area boundary one-point measurement. Labeled images are naturally suitable for this type of measurement. Whether the results of each process are meaningful or not depends on the specific application; however, these procedures can handle difficult situations such as the so-called "four-color problem," in which areas of four filled-in images all come into contact at the same point with no background values present. They can also process scattered areas such as the areas labeled 4 in the illustration below, by using a single rectangle to circumscribe the two separate areas.

Sample image:



2.5 Perimeter Measurement by Area

This process examines 2 x 2-pixel local areas of 8-way or 4-way connected raster scan images, and applies the algorithms described below to combinations of local areas to measure the perimeter. Perimeter measurements are made up from three separate measurements of vertical, horizontal, and diagonal components, a breakdown that also aids in measuring aspect ratios.



Simplified example of a perimeter measurement: determining the length of the perimeter of the area at left with area number (label value) 5.

Vertical: 4

Horizontal: 4

Diagonal: 4

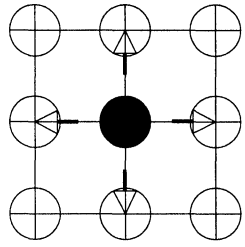
Perimeter: $4 + 4 + 4\sqrt{2} \approx 13.66$

The above illustration shows how the perimeter of area number (label number) 5 would be measured. All values outside this area are considered to be background (label number 0). The aspect ratio is 1, and the measurement application assumes that diagonal distances are multiplied by $\sqrt{2}$. The Feature chip measures the perimeter in terms of three components: vertical, horizontal, and diagonal. The CPU or DSP can then total the three components, as well as perform any supplementary processing using aspect ratio, etc.

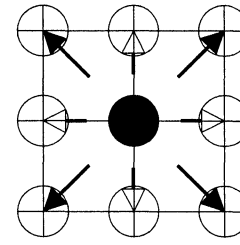
Note that the algorithm for this process determines a perimeter length of 0 for any area that consists of only one pixel.

[Reference 1: 4-Way and 8-Way Connectedness Labeling]

The following is a general explanation of labeling processing with respect to image resolution and measurement. (For details, consult a textbook on image processing.) Labeling (also called linked component labeling) is the process of attaching a particular value to adjoining pixels so that adjacent points can be seen as belonging to a continuous area. Labeling normally accepts binary images as input, and outputs a labeled image with area numbers (label values) allocated to each pixel on the screen.

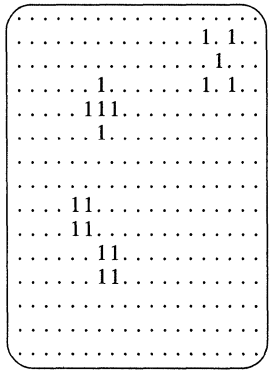


4-connectedness labeling: search for connections in 4 directions from the central point

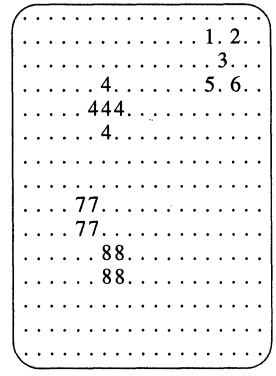


8-connectedness labeling: search for connections in 8 directions from the central point

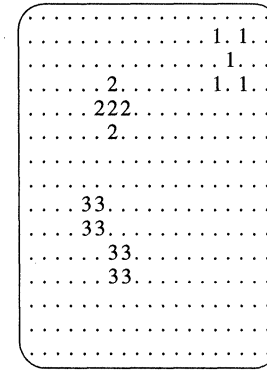
Figure (A) below shows a binary image input for labeling processing. Figures (B) and (C) represent the resulting output images.



(A) Binary image before labeling



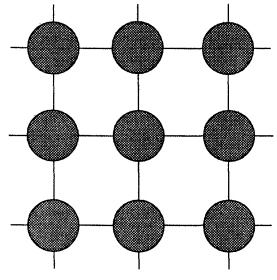
(B) Result of 4-connection labeling



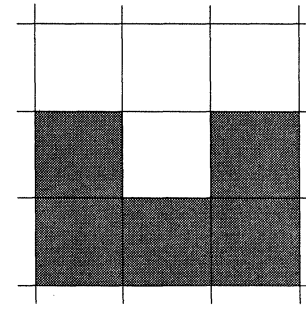
(C) Result of 8-connection labeling

[Reference 2: Treatment of Pixels]

Texts on the subject of image processing point out that there are two ways of defining the concept of a pixel for use in digital image processing. One approach is to treat the pixel as a point at the vertex of coordinates X and Y: that is, as having no area of its own. The other approach is to treat the pixel as a minimum unit of planar space having a surface area defined as 1 because it exists across a given unit of length along both the X and Y axes (an inter-pixel length). In discussing feature extraction in this section, pixels are referred to as "point pixels" when treated as points, and as "plane pixels" when treated as units of surface area.



Binary image viewed as pixel dots in a lattice network



Binary image viewed as pixel planes in a grid

Now consider an area one pixel in size and an area of four pixels in a 2 x 2 square configuration as applied to measuring perimeters and surface areas. The diagrams below illustrate both cases viewed as point pixels and also as plane pixels. The accompanying table indicates (with an "F") which concepts the Feature chip uses in which measurement process.

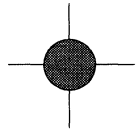


Image of one pixel in a point array

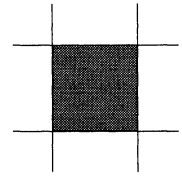


Image of one pixel in a plane array

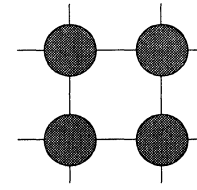


Image of four pixels in a point array

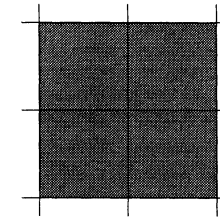


Image of four pixels in a plane array

Subject image	One pixel		Four-pixel square	
Pixels viewed as:	Point	Plane	Point	Plane
Perimeter:	0 (F)	4	4 (F)	8
Surface area (histogram):	0	1 (F)	1	4 (F)

The Feature chip uses point pixels to measure perimeters, and plane pixels to measure surface area when calculating in terms of the frequency of occurrence of pixel values within a given area (histogram measurement). In theory, all pixels (both point pixels and plane pixels) should be treated identically; however, the difference between the point and plane concept is only significant when dealing with extremely small areas, and in practice these small areas are most frequently ignored when eliminating noise. Therefore, there are few if any problems with using both concepts at the same time. Furthermore, no clear mathematical distinction has been made as to the correctness of one concept or the other, and no particular problems are posed by using whichever concept works best for realizing particular continuous (analog) phenomena in terms of discrete mathematics (sampling values, digital values). For example, although the Feature chip uses the point pixel concept for measuring, perimeter measurement could also be seen in terms of plane pixels by thinking of the distance between the centers of plane pixels.

2.5.1 Algorithm for Perimeter Measurement of Images Labeled with 8-Way Connection Labeling

The following section describes the algorithm used for perimeter measurement in raster-scan images that have been labeled using 8-way connection labeling. This algorithm detects the boundary between continuously labeled areas and the background by looking at 2 x 2-pixel local areas, and counts vertical, horizontal, and diagonal perimeter lengths independently.

Input data local area:

p3	p2
p1	p0

p3	p2	p1	p0	Local area label data	Perimeter V	H	D	p3	p2	p1	p0	Local area label data	Perimeter V	H	D
0	0	0	0	0 0 0 0	0	0	0	1	0	0	0	* 0 0 0	0	0	0
0	0	0	1	0 0 0 *	0	0	0	1	0	0	1	* 0 0 *	0	0	2
0	0	1	0	0 0 * 0	0	0	0	1	0	1	0	* 0 * 0	1	0	0
0	0	1	1	0 0 * *	0	1	0	1	0	1	1	* 0 * *	0	0	1
0	1	0	0	0 * 0 0	0	0	0	1	1	0	0	* * 0 0	0	1	0
0	1	0	1	0 * 0 *	1	0	0	1	1	0	1	* * 0 *	0	0	1
0	1	1	0	0 * * 0	0	0	2	1	1	1	0	* * * 0	0	0	1
0	1	1	1	0 * * *	0	0	1	1	1	1	1	* * * *	0	0	0

Notes:

1) In the above table, whenever a given pixel p_x (where x = 0 to 3) has the value 0, that pixel is considered background. A 1 indicates a non-background value.

2) In the local areas in the above table, pixel values of 0 indicate that the pixel contains background. Pixels marked with an asterisk (*) have non-background values.

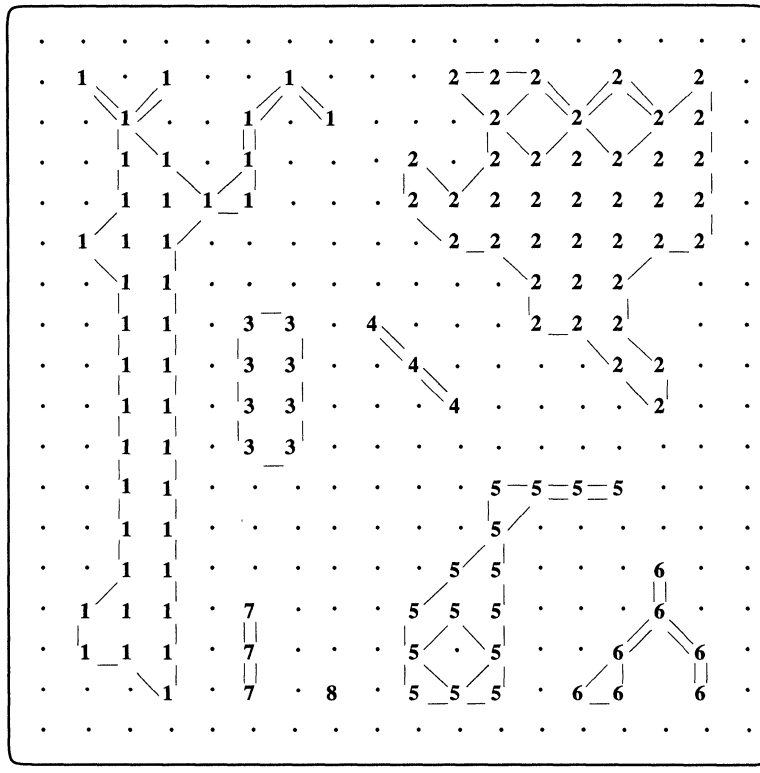
3) Linked areas consisting of a single pixel have a perimeter of 0.

4) If an area contains an enclosed "hole," the interior perimeter of the hole is included in the perimeter calculation.

   To find only the external perimeter, use this procedure: perform labeling using inverted binary image data, then set all labels other than background to 0 before performing perimeter measurement.

2.5.2 Example: Perimeter Measurement of Images Labeled with 8-Way Connection Labeling

Input Image



Measurement Results

Label value	Perimeter		
	Vertical	Horizontal	Diagonal
1	24	2	16
2	9	5	21
3	6	2	0
4	0	0	4
5	7	7	7
6	5	1	5
7	4	0	0
8	0	0	0

2.5.3 Algorithm for Perimeter Measurement of Images Labeled with 4-Way Connection Labeling

The following section describes the algorithm used for perimeter measurement in raster-scan images that have been labeled using 4-way connection labeling. This algorithm detects the boundary between continuously labeled areas and the background by looking at 2 x 2-pixel local areas, and counts vertical, horizontal, and diagonal perimeter lengths independently.

Input data local area:

p3	p2
p1	p0

p3	p2	p1	p0	Local area label data	Perimeter V	H	D	p3	p2	p1	p0	Local area label data	Perimeter V	H	D
0	0	0	0	0 0 0 0	0	0	0	1	0	0	0	* 0 0 0	0	0	0
0	0	0	1	0 0 0 *	0	0	0	1	0	0	1	* 0 0 *	0	0	0
0	0	1	0	0 0 * 0	0	0	0	1	0	1	0	* 0 * 0	1	0	0
0	0	1	1	0 0 * *	0	1	0	1	0	1	1	* 0 * *	0	0	1
0	1	0	0	0 * 0 0	0	0	0	1	1	0	0	* * 0 0	0	1	0
0	1	0	1	0 * 0 *	1	0	0	1	1	0	1	* * 0 *	0	0	1
0	1	1	0	0 * * 0	0	0	0	1	1	1	0	* * * 0	0	0	1
0	1	1	1	0 * * *	0	0	1	1	1	1	1	* * * *	0	0	0

Notes:

1) In the above table, whenever a given pixel p_x (where x = 0 to 3) has the value 0, that pixel is considered background. A 1 indicates a non-background value.

2) In the local areas in the above table, pixel values of 0 indicate that the pixel contains background. Pixels marked with an asterisk (*) have non-background values.

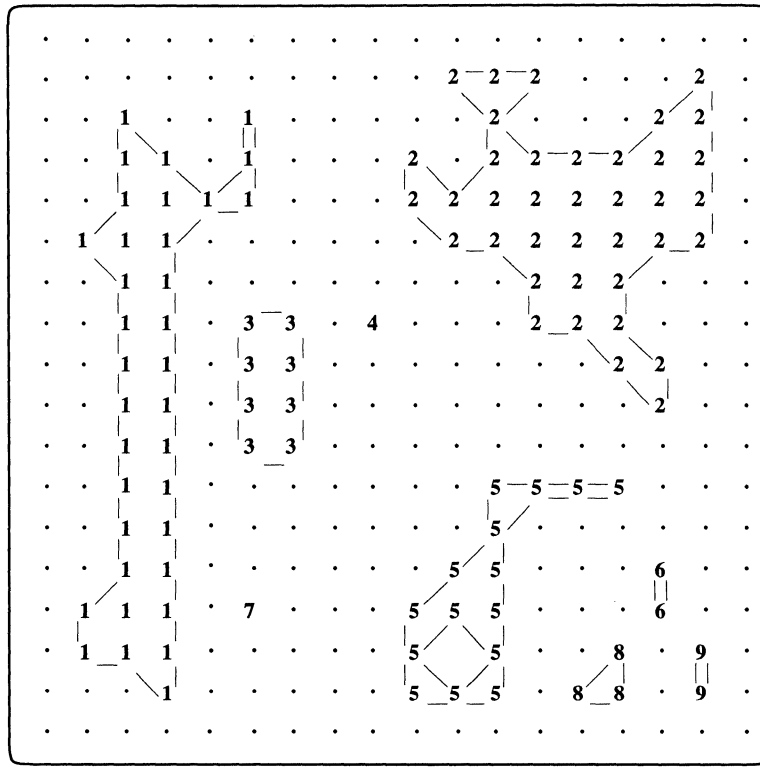
3) Linked areas consisting of a single pixel have a perimeter of 0.

4) If an area contains an enclosed "hole," the interior perimeter of the hole is included in the perimeter calculation.

   To find only the external perimeter, use this procedure: perform labeling using inverted binary image data, then set all labels other than background to 0 before performing perimeter measurement.

2.5.4 Example: Perimeter Measurement of Images Labeled with 4-Way Connection Labeling

Input Image



Measurement Results

Label value	Perimeter		
	Vertical	Horizontal	Diagonal
1	24	2	8
2	9	7	13
3	6	2	0
4	0	0	0
5	7	7	7
6	2	0	0
7	0	0	0
8	1	1	1
9	2	0	0

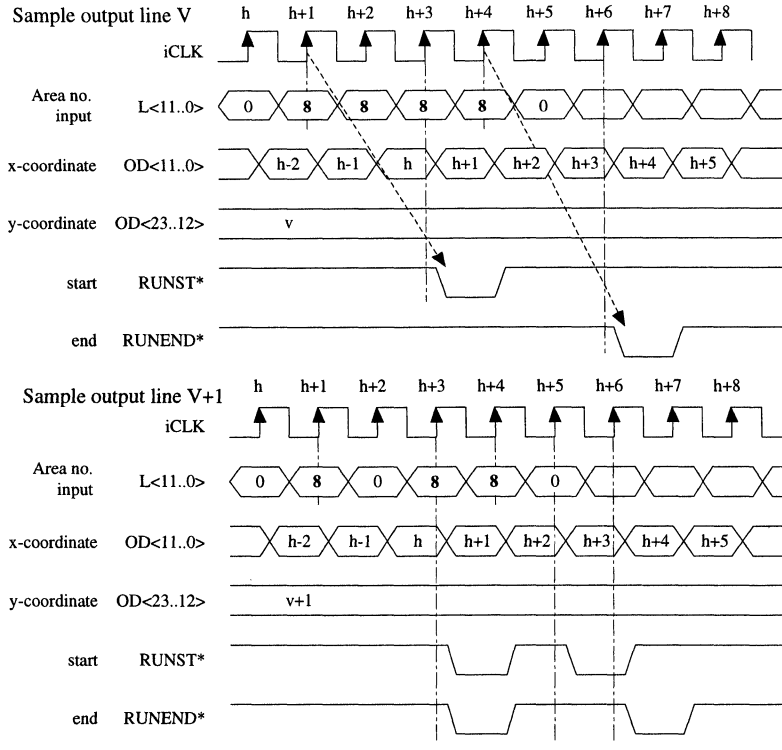
2.6 Run Coordinate Detection

- This process operates on any labeled image that uses a background value of 0.
- For any area (with area number ≥ 1) designated using the area number input signal (L<11..0>), this process outputs a run start signal (RUNST*) and run end signal (RUNEND*), along with the coordinates of both ends of all runs on every line over the output signal pins OD<23..0>. Note that the run coordinate detection and measurement process differs from other measurement processes in that the resulting data is of variable length and can therefore only be output to external circuits, and cannot be stored in results memory. The process includes a delay of two clock cycles before the output of the run start and run end signals.
- This function can be used as a preliminary step for binary image run length processing, by beginning with the starting point for the area of interest and measuring the length of each run of 0 and each run of 1. Also, because it can operate on all areas simultaneously, this function provides an effective way to detect and measure boundary pixels. It can locate the smallest unit of data having two-dimensional coordinates. This function can be used as a preliminary step for special measurements of individual areas by an external DSP.
- Run coordinate detection can be applied in combination with other measurement processes. Note, however, that this requires the load capacity of the OD<23..0> signal pins to be kept at 30 pF or lower.

[Sample processing]

The area below represents an input image with background value 0 and area number 8:

	h	h+1	h+2	h+3	h+4	h+5
v	0	8	8	8	8	0
v + 1	0	8	0	8	8	0



2.7 Measuring Moments

The IP90C18 can measure the zero-degree moment, the primary moment, and the secondary moment about the origin of any two-dimensional image in real time. Using multiple IP90C18 chips enables a system to calculate all three moments at the same time. Any one of the three moments can be calculated for up to 2048 areas (with 48-bit results data width) or 4096 areas (with 24-bit results data width) simultaneously. Using two IP90C18 chips makes it possible to expand the number of simultaneous measurements up to 4096 areas with a 48-bit results data width, without requiring special external circuits. The number of areas can be expanded still further by using a simple decoder circuit plus additional IP90C18 chips.

Although the IP90C18 can directly measure zero-degree moments, primary moments, and secondary moments about the origin, a separate DSP or CPU is required when calculating center of gravity, or primary or secondary moments about the center of gravity or the principal axis.

2.7.1 Moment Characteristics

If the grayscale value f(x, y) of any given point (x, y) in a two-dimensional image f represents the mass of point (x, y), then the moment of inertia and center of gravity of the particular area in which that point lies represent moment quantities and moment characteristics with respect to that area. These moment characteristics can provide information about the location and shape of the area.

Equation (1) is the general formula for the p- and q-order moments of two-dimensional image f about its center of gravity (Xg, Yg).

$$M_{pq} = \sum_x \sum_y (x - X_g)^p \cdot (y - Y_g)^q \cdot f(x, y) \qquad (1)$$

In the following discussion, a capital M represents a moment characteristic about the center of gravity, while a small m represents a moment characteristic about the origin. The Feature chip's direct measurement capability is limited to binary images with respect to each area, and to the following six types of moment characteristic measurement, representing zero-order up to secondary moments about the origin point: m_{00}, m_{10}, m_{01}, m_{20}, m_{11}, m_{02}. The following sections describe each of these measurement functions.

2.7.2 Zero-Order Moment Characteristics and Surface Area

The zero-degree moment is here considered a representation of weight, using grayscale values as mass, and is calculated using this formula:

$$M_{00} = m_{00} = \sum_x \sum_y f(x, y) \qquad (2)$$

Equation (2) can be used to calculate a grayscale-weighted surface area. Here f(x, y) represents the grayscale value of the pixel located at coordinates (x, y). Equation (3) below can be used to calculate the surface area for areas in a labeled image. Here, because the n^{th} object is represented in binary values, the quantity d(x, y) must equal 0 or 1.

$$M_{00} = m_{00} = \sum_x \sum_y d(x, y) \qquad (3)$$

The Feature chip can calculate in real time, simultaneously with the input of the raster-scanned image, the moment m_{00} described by equation (2) for the zero-degree moment of a grayscale image, or the moment m_{00} described by equation (3) for the zero-degree moment of a binary image. However, the primary moment and secondary moment characteristics described below can only be calculated for areas within binary images. This means that area number (labeled image) input is valid for all three types of moment characteristic calculations, although grayscale image input is only valid for zero-degree moment grayscale-weighted surface area measurement.

2.7.3 Primary Moment Characteristics and Center of Gravity

Equations (4) and (5) give the definition of the primary moment used by the Feature chip. The chip uses an internal coordinate counter to calculate the sum of the coordinates of pixels in each area, which allows it to calculate the primary moment about the origin. For any given input image, this calculation can be performed using either of these equations:

$$M_{10} = \sum_x \sum_y x \cdot d(x,y) \tag{4}$$

$$M_{01} = \sum_x \sum_y y \cdot d(x,y) \tag{5}$$

From this calculation and the zero-degree moment, an external DSP or CPU can calculate the moment of any given area about its center of gravity using equations (6) and (7):

$$M_{10} = \sum_x \sum_y (x - X_g) \cdot d(x,y) = \sum_x \sum_y x \cdot d(x,y) - X_g \cdot \sum_x \sum_y d(x,y) = m_{10} - X_g \cdot m_{00} \tag{6}$$

$$M_{01} = \sum_x \sum_y (y - Y_g) \cdot d(x,y) = \sum_x \sum_y y \cdot d(x,y) - Y_g \cdot \sum_x \sum_y d(x,y) = m_{01} - Y_g \cdot m_{00} \tag{7}$$

Here X_g and Y_g represent the X and Y axis coordinates of the center of gravity. Because the center of gravity is defined as the point at which the primary moment is 0, these coordinates can be represented by equations (8) and (9). Because the values m_{00}, m_{10}, and m_{01} calculated by the Feature chip are measured in real-time, virtually simultaneously with the input of the raster-scanned image, these quantities can be derived by simple calculations using an external CPU or DSP.

$$X_g = \frac{m_{10}}{m_{00}} = \frac{\sum_x \sum_y x \cdot d(x,y)}{\sum_x \sum_y d(x,y)} \tag{8}$$

$$Y_g = \frac{m_{01}}{m_{00}} = \frac{\sum_x \sum_y y \cdot d(x,y)}{\sum_x \sum_y d(x,y)} \tag{9}$$

2.7.4 Secondary Moment and Angle of Primary Axis

Equations (10, (11), and (12) give the definition of the secondary moment used by the Feature chip. The chip uses an internal coordinate counter to calculate the sum of the coordinates of pixels in each area, which allows it to calculate the primary moment about the origin. For any given input image, this calculation can be performed using any of these equations:

$$M_{20} = \sum_x \sum_y x^2 \cdot d(x,y) \tag{10}$$

$$M_{02} = \sum_x \sum_y y^2 \cdot d(x,y) \tag{11}$$

$$M_{11} = \sum_x \sum_y x \cdot y \cdot d(x,y) \tag{12}$$

From this calculation, an external DSP or CPU can calculate the moment of each area about its center of gravity using these equations:

$$M_{20} = \sum_x \sum_y (x - X_g)^2 \cdot d(x,y) = \sum_x \sum_y (x^2 - 2X_g x + X_g{}^2) \cdot d(x,y) \tag{13}$$

$$= \sum_x \sum_y x^2 \cdot d(x,y) - 2X_g \sum_x \sum_y x \cdot d(x,y) + X_g{}^2 \sum_x \sum_y d(x,y)$$

$$= m_{20} - 2\frac{m_{10}}{m_{00}} m_{10} + \left(\frac{m_{10}}{m_{00}}\right)^2 m_{00} = m_{20} - \frac{m_{10}{}^2}{m_{00}}$$

$$M_{02} = \sum_x \sum_y (y - Y_g)^2 \cdot d(x,y) = \sum_x \sum_y (y^2 - 2Y_g x + Y_g{}^2) \cdot d(x,y) \tag{14}$$

$$= \sum_x \sum_y y^2 \cdot d(x,y) - 2Y_g \sum_x \sum_y y \cdot d(x,y) + Y_g{}^2 \sum_x \sum_y d(x,y)$$

$$= m_{02} - 2\frac{m_{01}}{m_{00}} m_{01} + \left(\frac{m_{01}}{m_{00}}\right)^2 m_{00} = m_{02} - \frac{m_{01}{}^2}{m_{00}}$$

$$M_{11} = \sum_x \sum_y (x - X_g)(y - Y_g) \cdot d(x,y) \tag{15}$$

$$= \sum_x \sum_y x \cdot y \cdot d(x,y) - Y_g \sum_x \sum_y x \cdot d(x,y) - X_g \sum_x \sum_y y \cdot d(x,y) + X_g Y_g \sum_x \sum_y d(x,y)$$

$$= m_{11} - \frac{m_{01}}{m_{00}} m_{10} + \frac{m_{10}}{m_{00}} m_{01} + \left(\frac{m_{10}}{m_{00}}\right)\left(\frac{m_{01}}{m_{00}}\right) m_{00}$$

$$= m_{11} - 2\frac{m_{10} \cdot m_{01}}{m_{00}} + \frac{m_{10} \cdot m_{01}}{m_{00}} = m_{11} - \frac{m_{10} \cdot m_{01}}{m_{00}}$$

From this secondary moment about the center of gravity, the chip can calculate an important indicator of the orientation of an area (or object): its primary axis of inertia. The primary axis of inertia can be described in terms of the angle q between the direction in which an area is growing (its primary axis) and the X axis, here called the angle of the primary axis. Because the values m_{00}, m_{10}, m_{01}, m_{20}, m_{11}, and m_{02}, calculated by the Feature chip, are measured in real-time, virtually simultaneously with the input of the raster-scanned image, the value of the angle q can be derived by simple calculations using an external CPU or DSP.

$$\theta = \frac{1}{2}\tan^{-1}\frac{2M_{11}}{M_{20}-M_{02}} = \frac{1}{2}\tan^{-1}\frac{2\left(m_{11}-\frac{m_{10}\cdot m_{01}}{m_{00}}\right)}{\left(m_{20}-m_{02}-\frac{m_{10}^{\ 2}-m_{01}^{\ 2}}{m_{00}}\right)} \qquad (16)$$

$$= \frac{1}{2}\tan^{-1}\frac{2\left(m_{00}\cdot m_{11}-m_{10}\cdot m_{01}\right)}{m_{00}\left(m_{20}-m_{02}\right)-m_{10}^{\ 2}+m_{01}^{\ 2}}$$

The primary axis can be represented by equation (17):

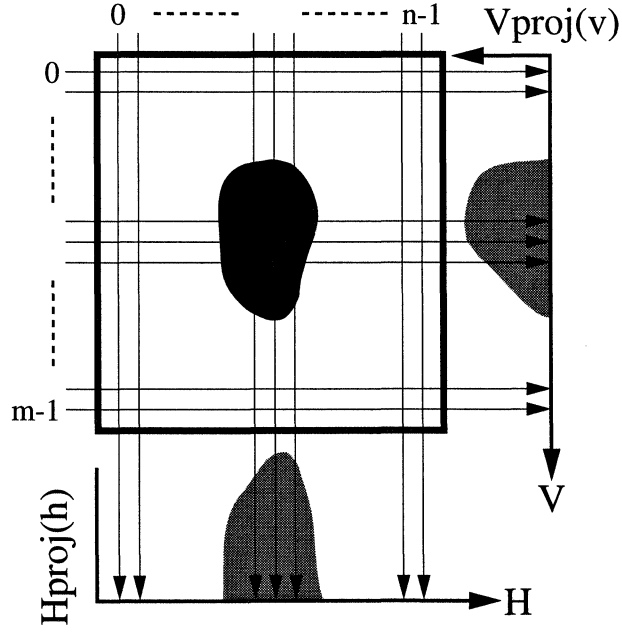
$$\left(y-Y_g\right) = \tan\theta\cdot\left(x-X_g\right) \qquad (17)$$

2.8 Grayscale Projection Processing (Peripheral Distribution Measurement) Algorithms

Grayscale projection processing, sometimes called peripheral distribution, contributes to higher speeds in image-processing functions. This process is often featured in texts and articles on image processing, because it can provide a simple, clear measurement of the characteristics of processes image areas.

Because of limits on circuit size, projection processing in previous systems has for the most part been practical only for binary images. Projection processing of binary images, however, does not permit sufficient accuracy or provide enough information to be of much use as a short-cut in image processing. These objectives require the use of grayscale projection processing, a function that has been difficult to realize. The IP90C18 Feature chip meets this demand, however, and provides projection processing of 12-bit grayscale values (1096 gradients).

Vertical (H-axis) projection processing is the process of measuring the sum of the grayscale values of all pixels on the screen having a given H-axis coordinate. Similarly, horizontal (V-axis) projection processing is the process of measuring the sum of the grayscale values of all pixels on the screen having a given V-axis coordinate. The Feature chip can perform H- and V-axis projection processing simultaneously, or on either axis separately.



Horizontal (V-axis) grayscale projection: $V_{proj}(v) = \sum_{h} f(h, v)$

Vertical (H-axis) grayscale projection: $H_{proj}(h) = \sum_{v} f(h, v)$

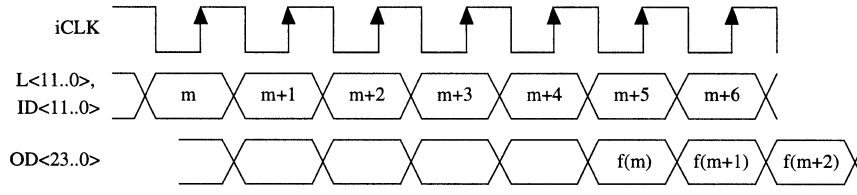
f(h, v) = grayscale value of pixel (h, v)

2.9 Table Conversion (LUT) Processing

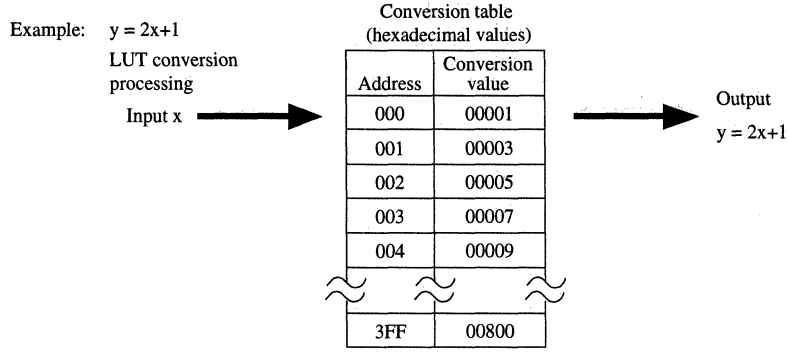
- The Feature chip includes a 12-bit input, 24-bit output look-up table (LUT) conversion function that can be used for a wide variety of processes. For each address in results memory input through the area number input signal (L<11..0>) or grayscale value input signal (ID<11..0), the LUT function outputs the corresponding value over the image data output (OD<23..0>) signal. Note that table conversion processing differs from other measurement functions in that the results of conversion can only be output to external circuits, and cannot be written to results memory. The process includes a delay of four clock cycles from input of grayscale values or area number values to the output of image data.
- This function is convenient for converting incoming area number values to new area numbers. For example, after area numbers (label values) are assigned to each area in an image and the surface area of each area measured using histogram processing, the smallest areas can be deleted (converted to background) by writing the background value to the corresponding addresses in memory. The same function can naturally be used to renumber any or all areas.

  Because this function can produce 24-bit output from 12-bit input data, it can be used to extend an area between labeled images, or to change label numbers. It provides a convenient way to perform a variety of procedures ranging from pixel-by-pixel to full-screen conversion, such as deleting all areas that include boundary pixels from the coordinates of a rectangle circumscribed about a given area, or merging overlapping areas by assigning them the same area number.
- This function can also convert grayscale input values to other values. This enables binary conversion or contrast enhancement of the results of grayscale histogram processing, or gamma-compensation. Because this function can produce 24-bit output from 12-bit input data, it can be used to expand the dynamic range of grayscale images, and is convenient for processes ranging from pixel-by-pixel conversion to full-screen conversion.
- The one-to-one replacement of input values by output values is ideal for constant-value calculations. Thus, expected measurement values can be written into results memory, for use in one-to-one calculation of sums, differences, products, dividends, etc., with respect to actual measurement results. Other uses include bit positioning (bit-shift and bit-rotation calculations), bit-masking calculations (conversion of selected bit positions to 0 or 1), and bit inversion processing.

  Input/output example:



Processing example



2.10 Cumulative Histogram Processing

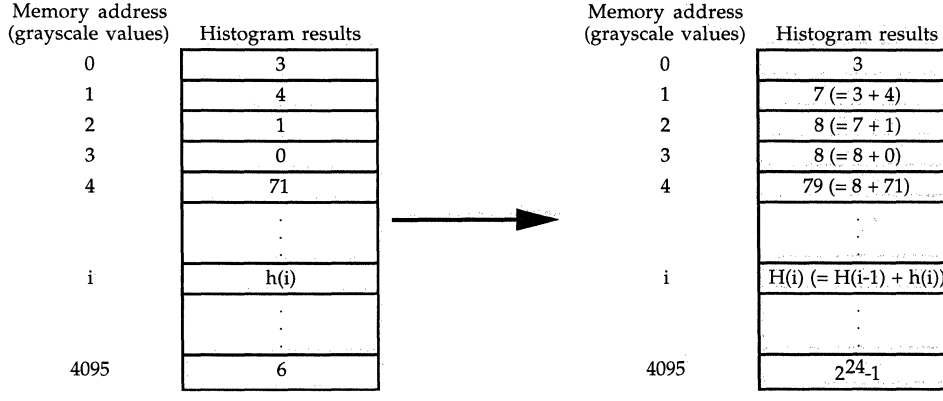
This function sequentially selects each address in results memory by means of an internal counter. The data in each selected address is then added to the data in that address on subsequent cycles, and the results are written again to that address. When the operation is ended, results memory thus contains a cumulative histogram distribution. The starting address is 0, and the ending address is 2047 in 2K area measurement mode and 4095 in 4K area measurement mode.

Cumulative histogram H(i):

$$H_i = \sum_{k-0}^{i} h(k) \qquad (1)$$

where h(k) is the histogram of gradient value k (in results memory address).

Processing example (4096 gradients, 2^{24}-1 pixel processing)



Section 3: Pin Functions

3.1 Pin Lists and Function Descriptions

Pin group	Pin	I/O	No.	Function	Remarks
Image data input	L0–11	I	12	Image area no. input (no delay)	Input of labeling area
	DL0–11	I	12	Image area no. input (one-line delay)	Labeling area input, delayed by one line; pull-up resistance (Note 2)
	ID0–11	I	12	Pixel grayscale data input	Input of grayscale-value image
Image timing	iCLK	I	1	Clock signal	f_{max} = 20/40 MHz
	VEN*	I	1	Vertical data enable signal	L: Enable, H: Disable
	HEN*	I	1	Horizontal data enable signal	L: Enable, H: Disable
	FEN*	I	1	Field enable/disable control for VEN* signal	L: Enable H: Disable (Note 3)
	IDEN*	I	1	Input data enable	L: Enable, H: Disable. Can be controlled by individual pixel (Note 3).
	DIDEN*	I	1	Extended background setting signal for DL0–DL11	L: Set background pixel value H: Use image data value (Notes 2, 3, and 5)
Image data output	OD0–23	O	24	Image data output	Output LUT, run coordinates, results memory dump
	ODEN*	I	1	OD output enable	ODEN* = H: OD0–OD23 High-Z ODEN* = L: OD0–OD23 drive enabled
	RUNST*	O	1	Run start signal	Run coordinate detection, start boundary pixel detection signal
	RUNEND*	O	1	Run end signal	Run coordinate detection, end boundary pixel detection signal
Status output	BUSY*	O	1	Processing 'busy' flag	L: Processing H: Not processing
	OVF*	O	1	Overflow flag	L: Overflow condition H: Normal (Note 4)
CPU Interface	R/M*	I	1	Register/memory selection	L: Results memory H: Register (Note 6)
	AD0–13	I	14	Control bus address input	CPU address bus
	DB0–23	I/O	24	Control bus data input/output	CPU data bus (Note 7)
	CS*	I	1	Chip select signal	
	WR*	I	1	Write enable signal	
	RD*	I	1	Read enable signal	
	BUSW0, BUSW1	I	2	Bus width: 8, 16, or 24 (Note 7)	(BUSW1, BUSW0) = (0, 0): 24-bit access (BUSW1, BUSW0) = (0, 1): 16-bit access (BUSW1, BUSW0) = (1, 0): 8-bit access (BUSW1, BUSW0) = (1, 1): (reserved)
	RST*	I	1	System reset	Does not clear memory; pull-up resistance connected
Test pin	TEST*	I	1	LSI internal test pin	Set high for normal use
Power source	Vdd	PW	19	5V	(Note 8)
	GND	PW	24	Ground	(Note 8)
Total pin count:			160		

Note 1: An asterisk (*) following a pin name denotes inverse logic.

Note 2: Used only when calculating the circumference.

Note 3: Set to low for normal use.

Note 4: Processing continues regardless of overflow conditions.

Note 5: The DIDEN* signal is the data/background ID signal for area signals DL0–DL11. DIDEN* is used as an extension (background recognition) of the number of processing areas for perimeter measurement.

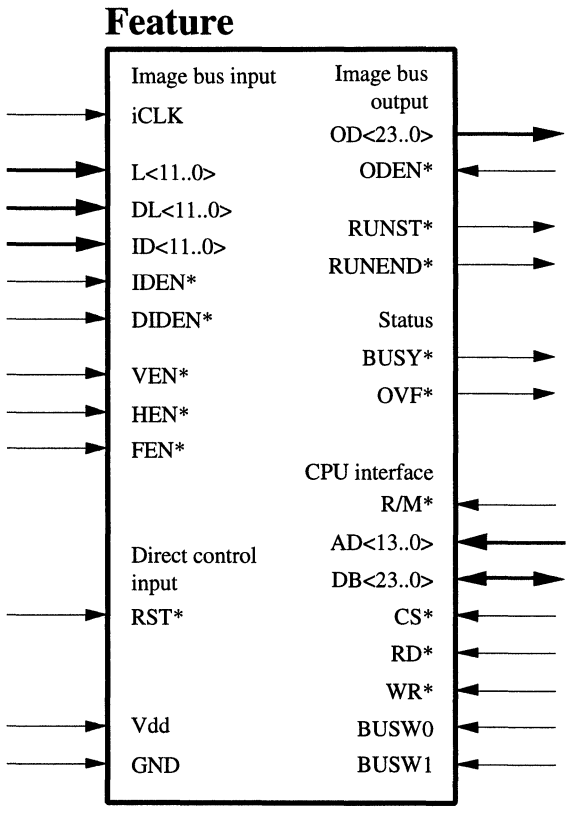
Note 6: Functions as a control bus address input signal (AD14) during access through the CPU bus. Set this bit low for access to internal results memory, or high for access to internal registers. To access internal results memory, designate the memory address as a byte address (regardless of bus width) and send it through control bus address input (AD0–AD13). The number of registers is limited to 16, but each register is mapped in 32-bit word units of 4 bytes each, as designated through control bus address input bits AD2, AD3, AD4, and AD5. For register access, control bus address bits AD6–AD13 are not decoded. For details, see Section 6.1, "Register Lists and Address Maps."

Note 7: According to the settings of the BUSW0 and BUSW1 pins, unused bits beginning at the high bit (MSB) end are placed in high-impedance status. Therefore, pull any unused input or I/O pins up or down, as appropriate. If any CMOS-LSI input pins are left unconnected, abnormal currents may occur in the LSI chip.

Note 8: Connect all power supply pins. This will help prevent abnormal operation due to pulse noise, etc.

3.1.1 Schematic Pin Diagram

The diagram below shows pin placement and function on the IP90C18 chip.



3.2 Pin Assignments

The IP90C18 and IP90C18-HS have identical pin assignments.

Pin No.	Name	Type
1	GND	PW
2	L0	I
3	L1	I
4	L2	I
5	L3	I
6	L4	I
7	L5	I
8	L6	I
9	L7	I
10	L8	I
11	L9	I
12	L10	I
13	L11	I
14	DL0	I
15	DL1	I
16	DL2	I
17	DL3	I
18	DL4	I
19	DL5	I
20	GND	PW
21	Vdd	PW
22	DL6	I
23	DL7	I
24	DL8	I
25	DL9	I
26	DL10	I
27	DL11	I
28	ID0	I
29	ID1	I
30	ID2	I
31	ID3	I
32	ID4	I
33	ID5	I
34	ID6	I
35	ID7	I
36	ID8	I
37	ID9	I
38	ID10	I
39	ID11	I
40	Vdd	PW

Pin No.	Name	Type
41	GND	PW
42	IDEN*	I
43	DIDEN*	I
44	FEN*	I
45	VEN*	I
46	HEN*	I
47	ODENB	I
48	Vdd	PW
49	GND	PW
50	OD0	O
51	OD1	O
52	OD2	O
53	Vdd	PW
54	GND	PW
55	OD3	O
56	OD4	O
57	OD5	O
58	GND	PW
59	OD6	O
60	OD7	0
61	Vdd	PW
62	GND	PW
63	OD8	O
64	OD9	O
65	OD10	O
66	Vdd	PW
67	GND	PW
68	OD11	0
69	OD12	0
70	OD13	0
71	Vdd	PW
72	GND	PW
73	OD14	O
74	OD15	O
75	OD16	O
76	GND	PW
77	OD17	O
78	OD18	O
79	OD19	O
80	Vdd	PW

Pin No.	Name	Type
81	GND	PW
82	OD20	O
83	OD21	O
84	OD22	O
85	OD23	O
86	Vdd	PW
87	GND	PW
88	RUNST*	O
89	RUNEND*	O
90	BUSY*	O
91	OVF*	O
92	Vdd	PW
93	GND	PW
94	TEST*	I
95	BUSW0	I
96	BUSW1	I
97	Vdd	PW
98	GND	PW
99	iCLK	I
100	GND	PW
101	RST*	I
102	CS*	I
103	WR*	I
104	RD*	I
105	AD0	I
106	AD1	I
107	AD2	I
108	AD3	I
109	AD4	I
110	AD5	I
111	AD6	I
112	AD7	I
113	AD8	I
114	AD9	I
115	AD10	I
116	AD11	I
117	AD12	I
118	AD13	I
119	R/M*	I
120	Vdd	PW

Pin No.	Name	Type
121	GND	PW
122	Vdd	PW
123	GND	PW
124	DB0	I/O
125	DB1	I/O
126	DB2	I/O
127	DB3	I/O
128	Vdd	PW
129	GND	PW
130	DB4	I/O
131	DB5	I/O
132	DB6	I/O
133	DB7	I/O
134	Vdd	PW
135	GND	PW
136	DB8	I/O
137	DB9	I/O
138	Vdd	PW
139	GND	PW
140	DB10	I/O
141	DB11	I/O
142	DB12	I/O
143	DB13	I/O
144	GND	P/W
145	DB14	I/O
146	DB15	I/O
147	DB16	I/O
148	Vdd	PW
149	GND	PW
150	DB17	I/O
151	DB18	I/O
152	DB19	I/O
153	DB20	I/O
154	Vdd	PW
155	GND	PW
156	DB21	I/O
157	DB22	I/O
158	DB23	I/O
159	Vdd	PW
160	GND	PW

3.3 Pin Placement Diagram

3.3.1 The IP90C18

160-pin QFP package (molded area 28 mm^2, pin pitch = 0.65 mm)

Note: The IP90C18 and IP90C18-HS have identical pin placements; only the embossed label is different.

3.3.2 The IP90C18-HS

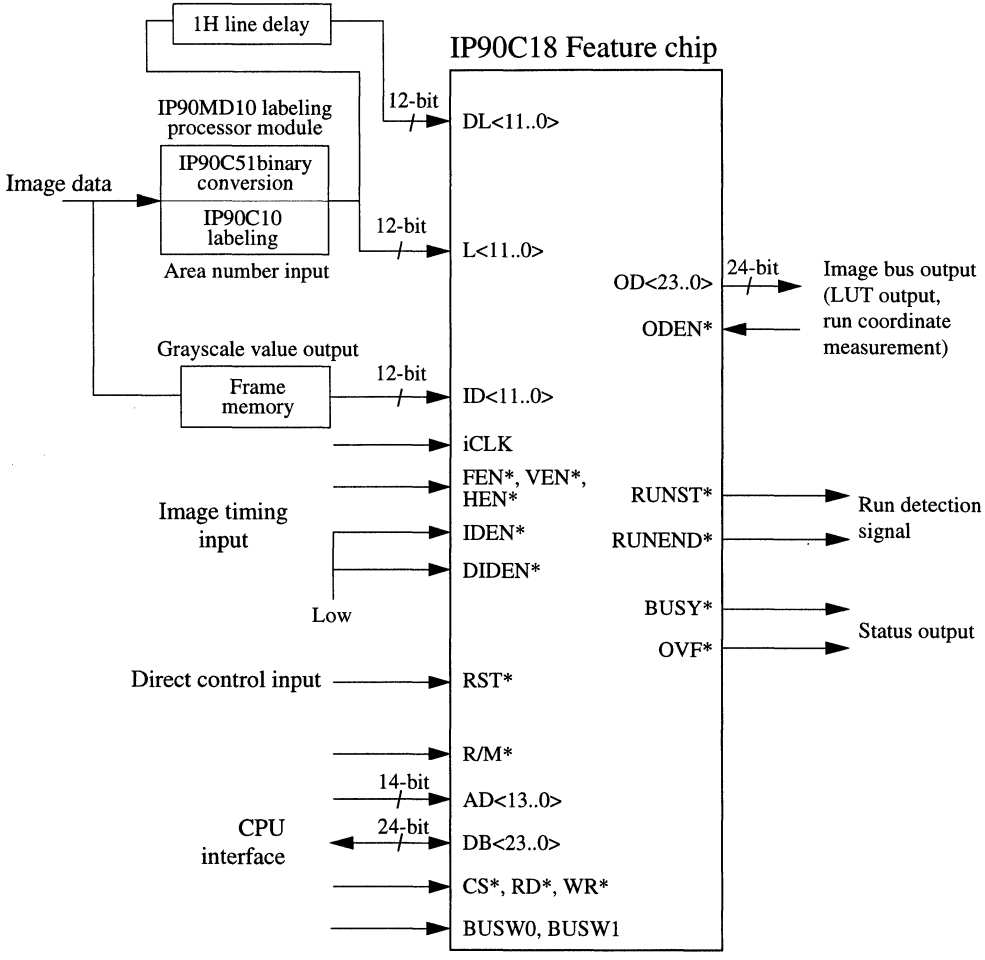
160-pin QFP package (molded area 28 mm^2, pin pitch = 0.65 mm)

Note: The IP90C18 and IP90C18-HS have identical pin placements; only the embossed label is different.

Section 4: Sample System Configuration and Operation

4.1 Sample Configuration: Basic System

The diagram below shows the configuration of a basic system using the IP90C18 Feature chip. The grayscale data and the area number (label) image for the items to be measured are input sequentially, in the direction of the raster scan. In this example, the grayscale and area number (label) images are connected together, and a delay of one line is applied to the perimeter measurement: this would allow the chip to perform all basic measurements.



Note 1: An asterisk (*) after a signal name denotes inverse logic.

Note 2: This block diagram is intended only as a functional description, and does not show all functions of the IP90C18 Feature chip. For full descriptions of functions, timing, and other information, see specific sections of this manual.

4.2 Operating Description

4.2.1 Image Synchronization and Control Signals (VEN*, HEN*)

This section gives a general description of the image synchronization and control signals used in the IP90C18's image input/output system. For details about signal timing, see Section 8.5, "AC Characteristics."

Note: Throughout this section, an asterisk (*) following a signal name denotes inverse logic.

4.2.1.1 Image Clock Synchronization

The IP90C18's processing circuits are synchronized with the image clock signal (iCLK). Note, however, that some functions (such as addresses for results memory, and the CPU bus signals that set the chip's internal modes) are synchronized with the CPU bus control signals (CS*, RD*, WR*) rather than with iCLK.

On the rise of iCLK, an input flip-flop latches the following input signals:

- area number signals (L<11..0>, DL<11..0>) } these make up the input signal
- pixel grayscale data signals (ID<11..0>) } for the image system
- image synchronization and control signals (FEN*, VEN*, HEN*, IDEN*, DIDEN*, RST*)

The status output signals (BUSY*, OVF*) and measurement signal output (OD<23..0>, RUNST*, RUNEND*) are output when iCLK rises after an external load drive delay.

4.2.1.2 Raster Scan Input

Image data is input in synchronization with the pixel clock and in the direction of the raster scan, and is input simultaneously through the area number signals (L<11..0>, DL<11..0>) and the pixel grayscale data signals (ID<11..0>) corresponding to L<11..0>. The Feature chip's internal circuits have a pipeline configuration that allows real-time throughput, during which feature extraction is performed and the results stored in memory. The area number input signals (L<11..0>, DL<11..0>) and pixel grayscale data input signals (ID<11..0>) include some signals not required for feature extraction. For details, see Section 5.2, "External Signal Connections and LSI Internal Connection Settings for Measurement Processes."

4.2.1.3 Frame Recognition Image Synchronization Signals (VEN* and HEN*)

Image data entering in raster scan format is put into two-dimension format using the vertical data enable signal VEN* and the horizontal data enable signal HEN*. The Feature chip defines a field of data as a cycle that begins when VEN* goes low, and ends when VEN* goes high again. The chip also defines each line in the field as a cycle that begins when HEN* goes low and ends when HEN* goes high again. Thus, a field of data is the data input while VEN* is low, and a line of data is the data input while HEN* is low. With non-interlaced data, each frame of the image consists of a single field, but with interlaced data, each frame consists of two fields. Thus, a frame of interlaced data consists of a first field defined by the first VEN* and a second field defined by the second VEN*.

A frame to be processed by the Feature chip starts when the execute flag in the execute register is set to "start feature extraction," followed by the input of a low-level VEN* signal (while FEN* has been low for at least one clock cycle).

The Feature chip has a field count control function that stops feature extraction automatically after processing a designated number of fields. This function is controlled by a register setting that can designate an automatic stop after processing 1–255 fields, or disable the count function to allow continuous processing (until stopped by the execute flag).

To input and measure a particular number of fields, designate the number (1–255) by setting the field count control register, then start execution by writing a 1 to the exec bit in the execution register.

Field count controls can also be applied from external circuits. For example, a logical circuit can be added to hold FEN* high after the input of a field: this is convenient for use with the IP90C55 image data stream controller chip (IMSC), which has a function that outputs one or two rectangular areas of interest (AOI-n*) from an image space.

When disabling the field count control function to allow continuous processing, set the field count control register to 00h, and start execution by writing 1 to the exec bit in the execution register. (To stop execution, write 0 to the exec bit.)

The Feature chip uses VEN* and HEN* internally as reset and start-count signals for the Vcnt and Hcnt counters, which are used to recognize vertical and horizontal image coordinates, respectively. When the change of VEN* to low is detected at the rise of iCLK, a VENpls pulse of one clock width is generated by synchronized differentiation within the chip. This pulse resets Vcnt to 0 at the start of the field. Then, when the change of HEN* to low is detected at the rise of iCLK, an HENpls_hl pulse of one clock width is generated by synchronized differentiation within the chip. HENpls_hl resets Hcnt to 0 at the start of each line.

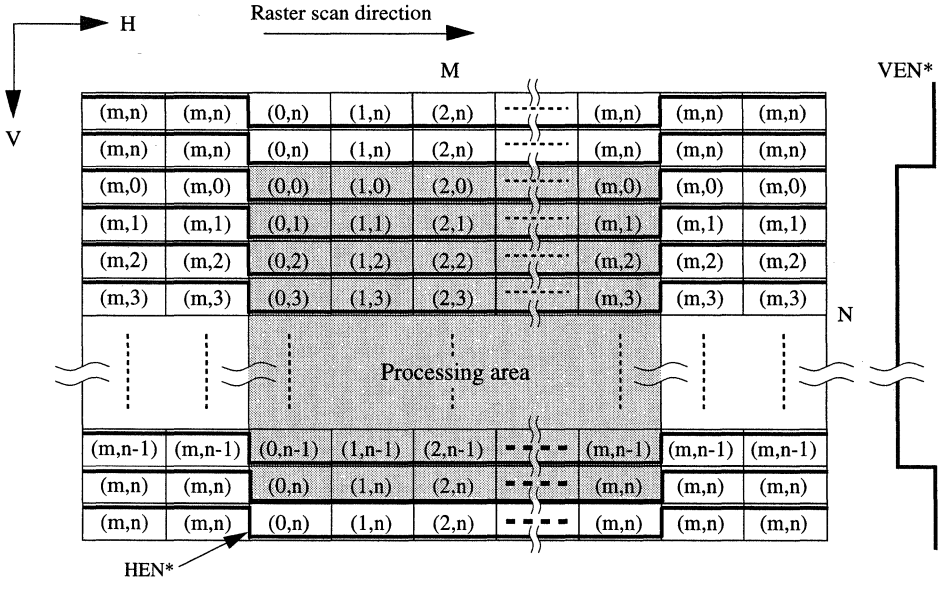
4.2.1.4 Effective Area Enable Signals (VEN*, HEN*)

VEN* is the vertical data enable signal and the enable signal for Vcnt. It works with HEN*, which is the horizontal data enable signal and the enable signal for Hcnt. The IP90C18 performs feature extraction processing on image data that is input when VEN* and HEN* are both low.

4.2.2 VEN* and HEN* Signals and Spatial Coordinates

The following diagram illustrates the relation between the VEN* and HEN* signals and the coordinates of each pixel in an image.

- Input image (M x N): Raster scan input of pixel data values, synchronized with the iCLK signal.
- Processing area (m x n): m = 0–4095 pixels, n = 0–4095 pixels.



Horizontal Coordinates

When the change of HEN* to low is detected at the rise of iCLK, an HENpls_hl pulse of one clock width is generated by synchronized differentiation within the chip. This pulse is the reset and start-count signal for the horizontal coordinate counter Hcnt. Thus, the horizontal coordinates (Ln) of image data point IDn (which are picked up at the rise of iCLK when the change of HEN* to low is detected) become the 0 point. Hcnt then starts counting iCLK signals as long as HEN* is low.

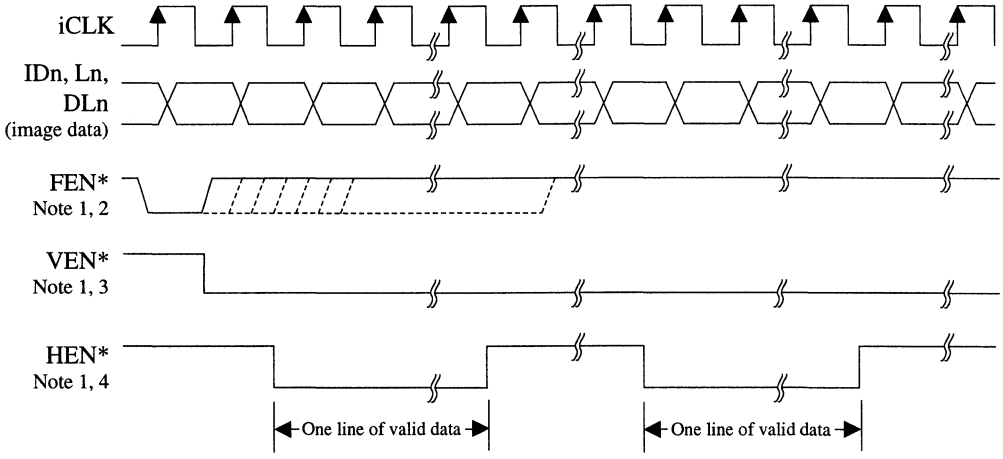
Vertical Coordinates

When the change of VEN* to low is detected at the rise of iCLK, a VENpls pulse of one clock width is generated by synchronized differentiation within the chip. VENpls is the reset and start-count signal for the vertical coordinate counter Vcnt. Thus, the vertical coordinates (Ln) of image data point IDn (which are picked up at the rise of iCLK when the change of VEN* to low is detected) become the 0 point. Vcnt then starts counting the number of times HEN* goes high while VEN* stays low.

Also note these points:

- Processing is performed on image data input while VEN* and HEN* are both low. (The chip's exec flag must also be set.)
- As shown above, set HEN* low for each line, and high for areas not to be processed.

4.2.3 Input Image Data Transfer Format



Note 1: An asterisk (*) following a signal name denotes inverse logic.

Note 2: VEN* is enabled one clock count before the next rise of iCLK after VEN* falls while FEN* is low. FEN* facilitates synchronization with other image-processing LSI devices and sequencers. Keep FEN* low whenever synchronization with other LSI devices is not needed (this is normal).

Note 3: Keep VEN* low while the data that makes up a field is being input (since the de-assertion of VEN* from low to high marks the end of a frame of valid data).

Note 4: Keep HEN* low while the data that makes up a line is being input.

4.2.4 IDEN* and Feature Extraction

The IDEN* signal pin enables and disables pixel-by-pixel processing of input image data (IDn, Ln, DLn). When IDEN* is set high to disable image data in the rectangular processing area defined by VEN* and HEN*, the IP90C18's feature extraction functions operate as shown in the following table.

Note that IDEN* functions independently of the coordinate counters Hcnt and Vcnt, so that the coordinate count does not stop when IDEN* goes high. When stopping the coordinate counters, take care to avoid causing glitches in iCLK.

IDEN* Signal and Feature Extraction

Feature Extraction Process	Change in chip operation when IDEN* is high (VEN* and HEN* are active low)
Grayscale histogram	Adds 0 to current grayscale values
Surface area measurement	Adds 0 to current area values
Grayscale-weighted area measurement	Sets current pixel grayscale value to 0 (adds 0 to current area values)
Coordinates of circumscribed rectangles	Ignores current coordinate values (coordinate counter output); existing diagonal coordinates are unchanged
Area end 1-point coordinates	Adds 0 to current area values
Primary and secondary moments	Adds 0 to current area values
Run coordinate extraction	Even if the current area value is for a boundary pixel, the RUNST* and RUNEND* signal outputs remain high
Perimeter measurement	Any perimeter measurement of current 2x2 pixel local areas is not counted (adds 0)
Grayscale projection	Sets current pixel grayscale value to 0 (adds 0)
Table conversion	No effect
Cumulative histogram	No effect

Sample Uses of the IDEN* Signal

- Feature extraction from areas of any shape:

  IDEN* can enable feature extraction from an area of any shape that lies within a rectangular processing area defined by the VEN* and HEN* signals.

- Expansion of processing data width across multiple chips:

  When performing histogram processing of more than 4096 values, the data width can be expanded by decoding the high bit and assigning its value to the IDEN* signal pin. Thus, for example, 16-bit histogram processing could be done by a system containing 16 Feature chips. The lower 12 bits are used as the measurement value for each chip, and the upper 4 bits are decoded and assigned to the IDEN* signal. A single-chip configuration could provide the same results by using 16 measurement iterations.

  This type of arrangement can also be used to expand the range of area numbers, as shown in Section 4.3, "Sample Configuration: Expanded System," or to expand the range of grayscale pixel values (the dynamic range).

4.2.5 Background Identification Signals (IDEN* and DIDEN*)

4.2.5.1 Background Identification Signals (IDEN* and DIDEN*)

Perimeter measurement differs from other feature extraction measurements in that it involves the area boundary detection process, which requires the use of a distinct background area number for boundary recognition. The background area is designated not by area number input value (L<11..0>, DL<11..0>), but by pixel units determined by the background identification signals IDEN* (the background identification signal for L<11..0>) and DIDEN* (the background identification signal for DL<11..0>). When measuring a perimeter, if a 2x2-pixel local area includes a boundary of a labeled area, the chip has to determine which linked areas (area numbers) the 2x2-pixel area belongs to. This requires inputting the area number signal L<11..0> and applying a one-line delay to the area number signal DL<11..0>. Thus, for perimeter measurement, identifying background areas pixel-by-pixel requires IDEN* (delayed by one line) and DIDEN*.

4.2.5.2 Example of Using the Background Identification Signals

As an example of how IDEN* and DIDEN* can be used, consider a system that expands the maximum number of areas (maximum number of labels) using multiple Features chips.

When a Feature chip measures a perimeter in three directions simultaneously, the maximum number of areas per chip (excluding the area number for background values) is 2047. However, by connecting several chips, each chip can measure the perimeter of each area (excluding the area number for background values) in three directions simultaneously. For details, see Section 4.3, "Sample Configuration: Expanded Systems."

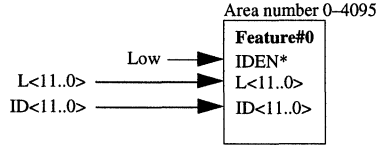
4.3 Sample Configuration: Expanded System

The IP90C18 Feature LSI chip can process areas and grayscale values in much greater quantities—12 bits (4096) or 11 bits (2048)—than previous devices. Its image-processing field can be used in a particularly wide range of specifications, and provides a wide variety of capabilities for various applications. The Feature chip is designed for systems that process expanded numbers of areas, pixel value resolutions, processing area sizes, or other characteristics.

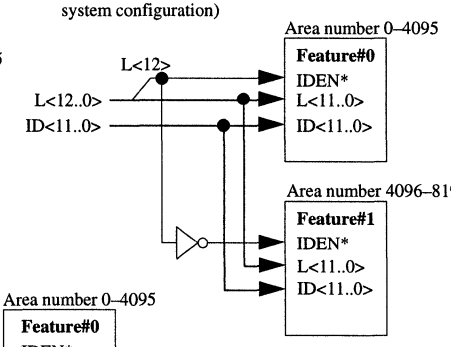
This section provides several sample configurations to show how multiple Feature chips can be used for real-time processing in expanded systems. These examples are intended as conceptual sketches only: thoroughly check timing, processing constraints, system requirements, and environmental factors before adapting any of these concepts to actual applications.

4.3.1 Sample System: Expanding the Number of Areas for Cumulative Processing by Area

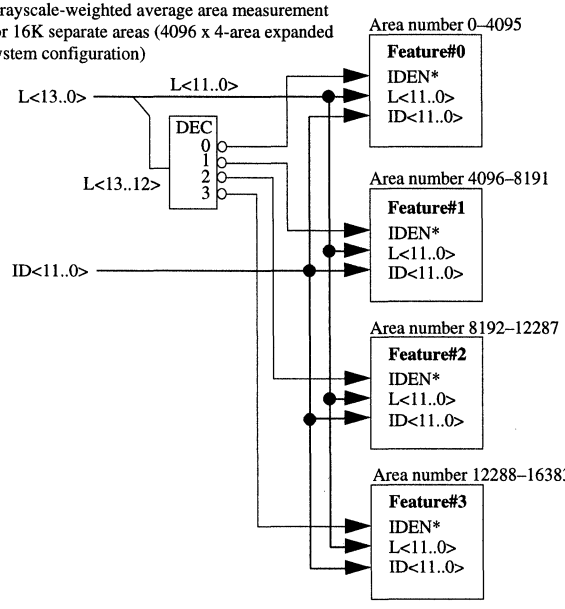
(A) Grayscale-weighted average area measurement for 4096 separate areas (basic system configuration)



(B) Grayscale-weighted average area measurement for 8192 separate areas (4096 x 2-area expanded system configuration)

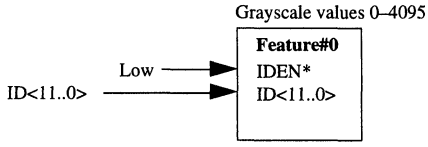


(C) Grayscale-weighted average area measurement for 16K separate areas (4096 x 4-area expanded system configuration)

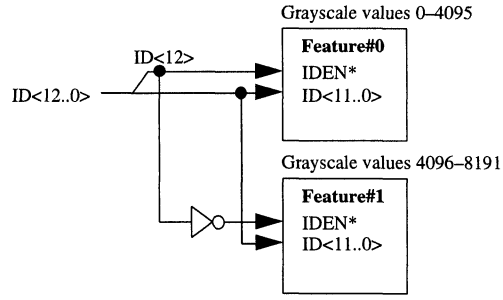


4.3.2 Sample System: Expanding the Number of Grayscale Pixel Gradients for Grayscale Histogram Processing

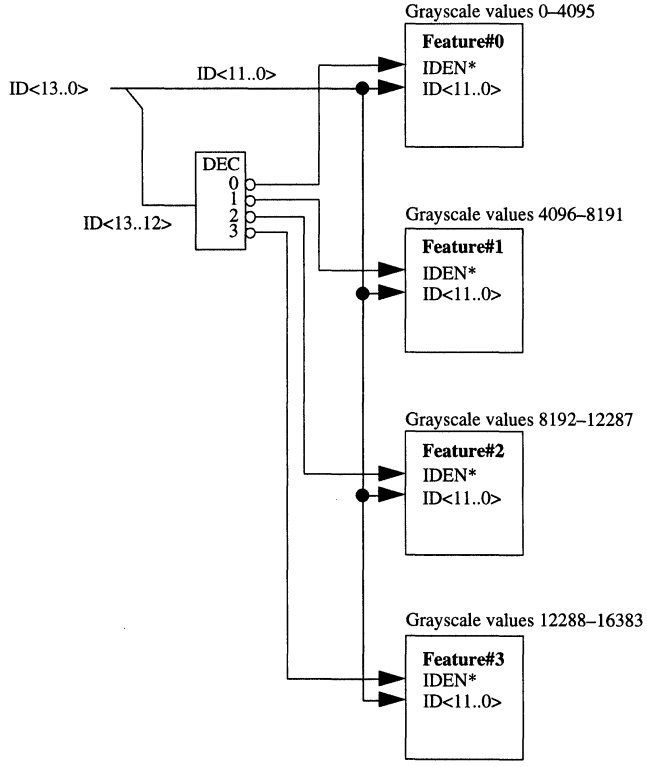
(A) Histogram measurement of 4096 grayscale gradient values (basic system configuration)



(B) Histogram measurement of 8192 grayscale gradient values (4096 x 2-gradient expanded system configuration, 13-bit histogram processing)

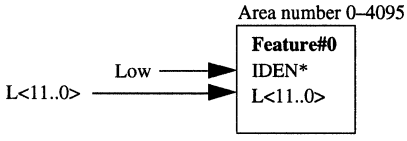


(C) Histogram measurement of 16K grayscale gradient values (4096 x 4-gradient expanded system configuration, 14-bit histogram processing)

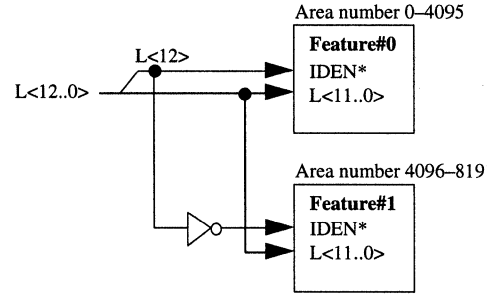


4.3.3 Sample System: Expanding the Number of Areas for Surface Area Measurement

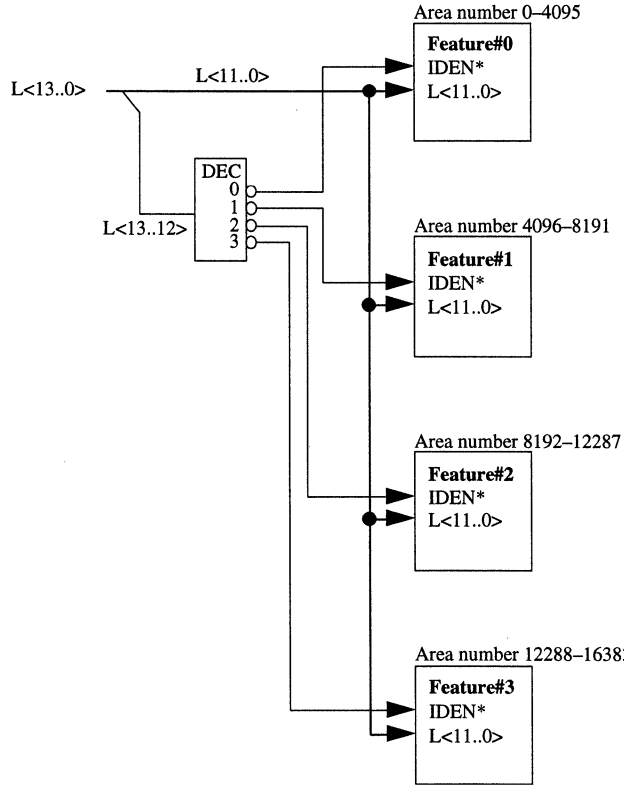
(A) Surface area measurement for 4096 separate areas (basic system configuration)



(B) Surface area measurement for 8192 separate areas (4096 x 2-area system configuration)

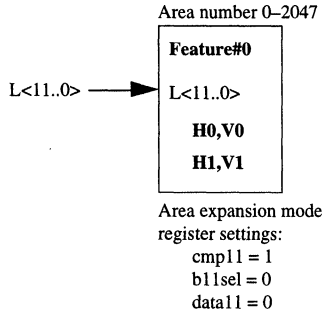


(C) Surface area measurement for 16K separate areas (4096 x 4-area system configuration)

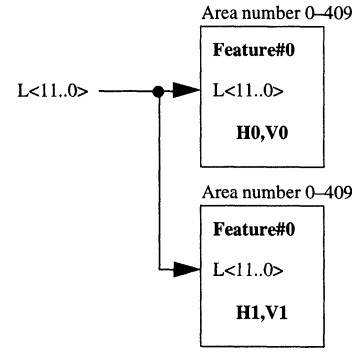


4.3.4 Sample System: Expanding the Number of Areas for Diagonal Coordinate Measurement of Circumscribed Rectangles

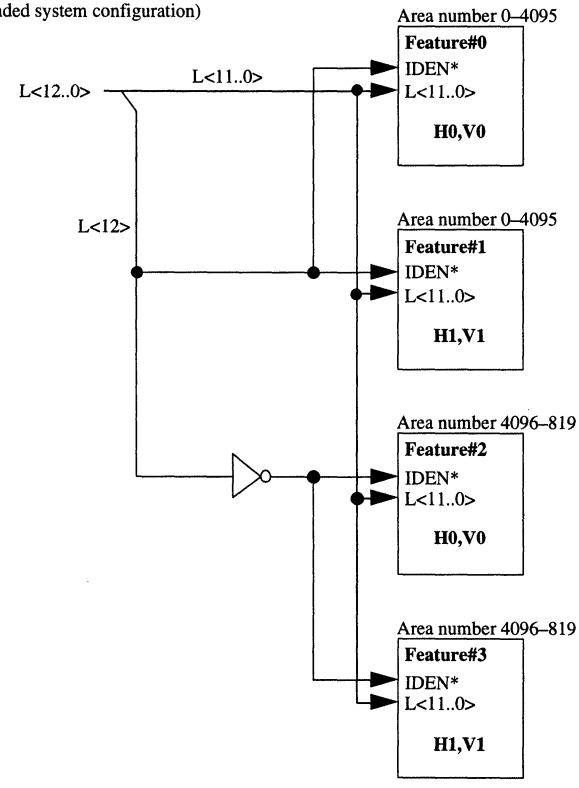
(A) Diagonal coordinate measurement for 2048 circumscribed rectangular areas (basic system configuration)



(B) Diagonal coordinate measurement for 4096 circumscribed rectangular areas (4096-area expanded system configuration)

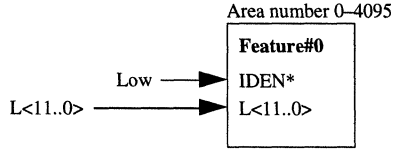


(C) Diagonal coordinate measurement for 8192 circumscribed rectangular areas (4096 x 2-area expanded system configuration)

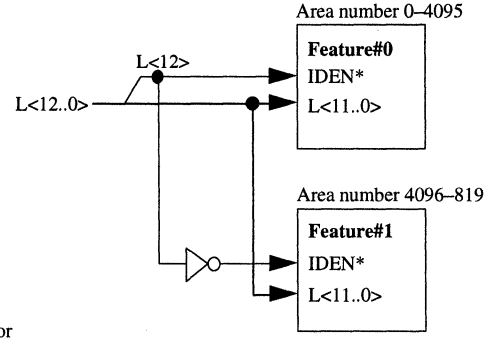


4.3.5 Sample System: Expanding the Number of Areas for One-Point Boundary Coordinate Measurement

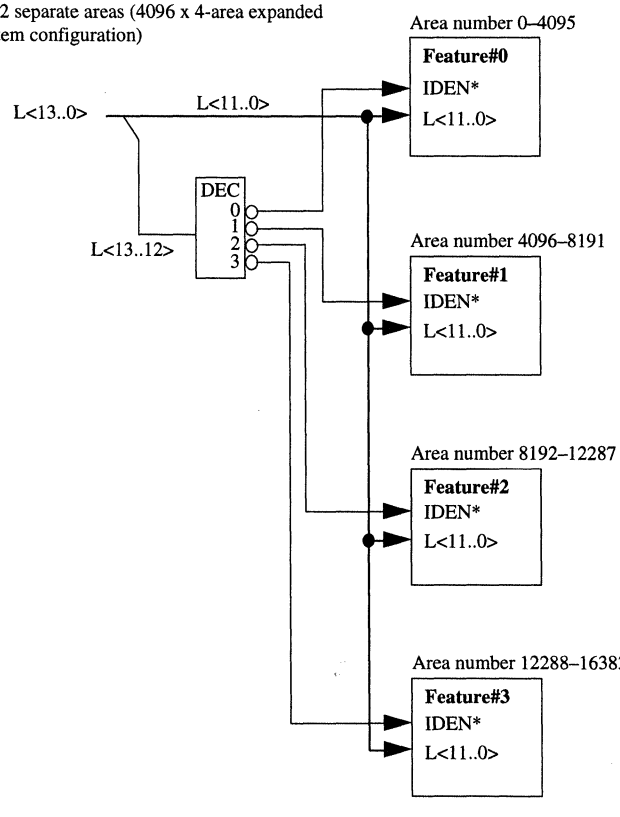
(A) One-point boundary coordinate measurement for 4096 separate areas (basic system configuration)



(B) One-point boundary coordinate measurement for 8192 separate areas (4096 x 2-area expanded system configuration)

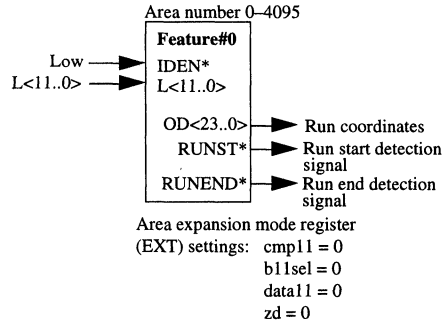


(C) One-point boundary coordinate measurement for 8192 separate areas (4096 x 4-area expanded system configuration)

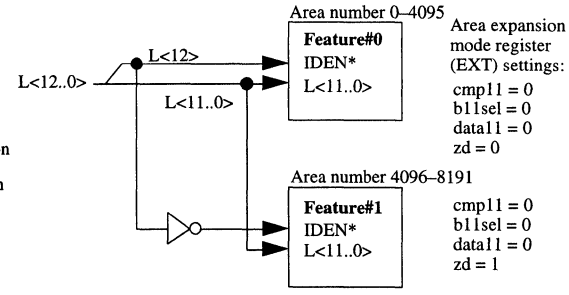


4.3.6 Sample System: Expanding the Number of Areas for Run Coordinate Detection by Area

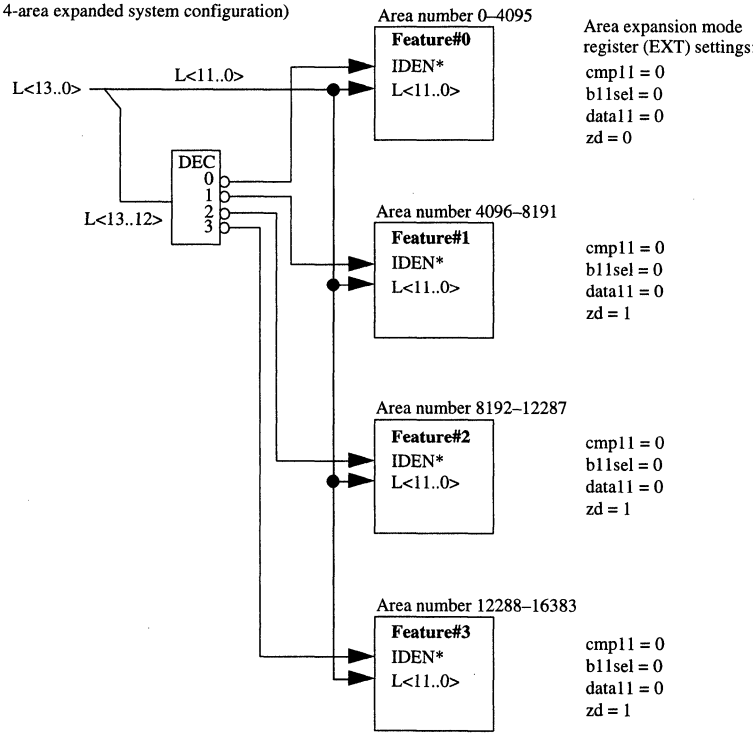
(A) Run coordinate detection for 4096 separate areas (basic system configuration)



(B) Run coordinate detection for 8192 separate areas (4096 x 2-area expanded system configuration)

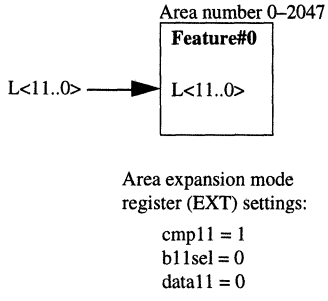


(C) Run coordinate detection for 16K separate areas (4096 x 4-area expanded system configuration)

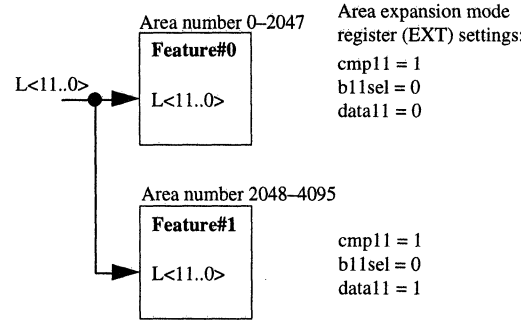


4.3.7 Sample System: Expanding the Number of Areas for Primary Moment Measurement by Area

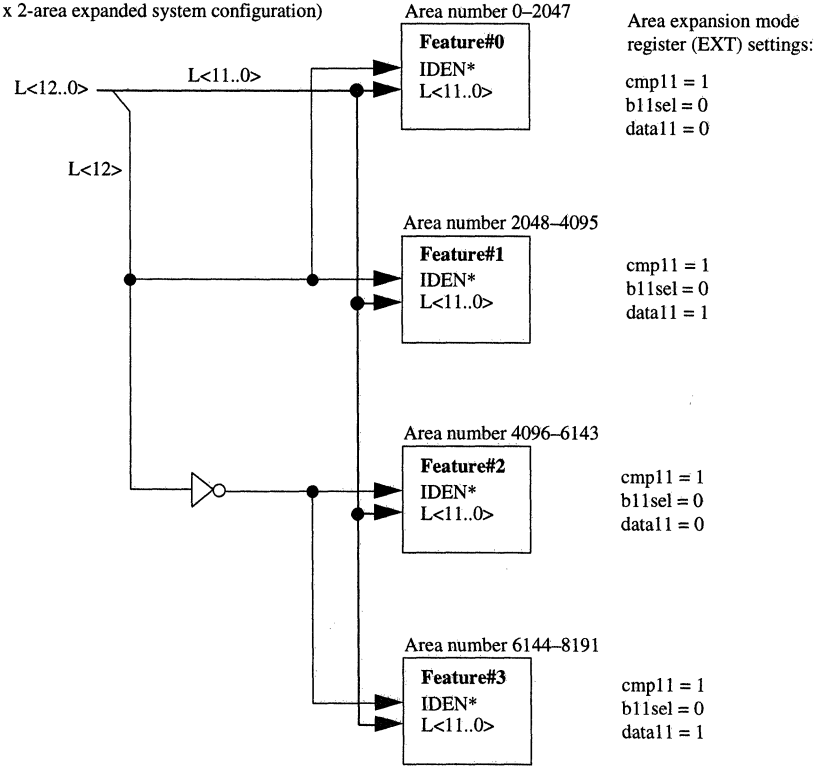
(A) Primary moment measurement for 2048 separate areas (basic system configuration)



(B) Primary moment measurement for 4096 separate areas (4096-area expanded system configuration)

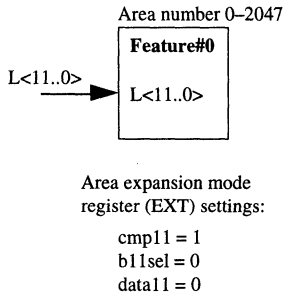


(C) Primary moment measurement for 8192 separate areas (4096 x 2-area expanded system configuration)

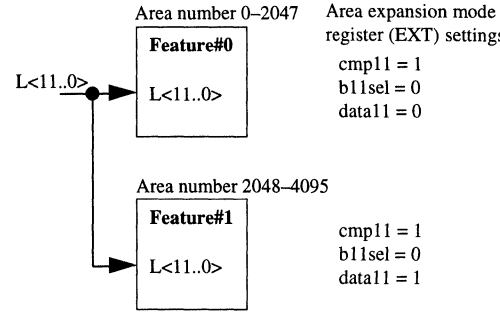


4.3.8 Sample System: Expanding the Number of Areas for Secondary Moment Measurement by Area

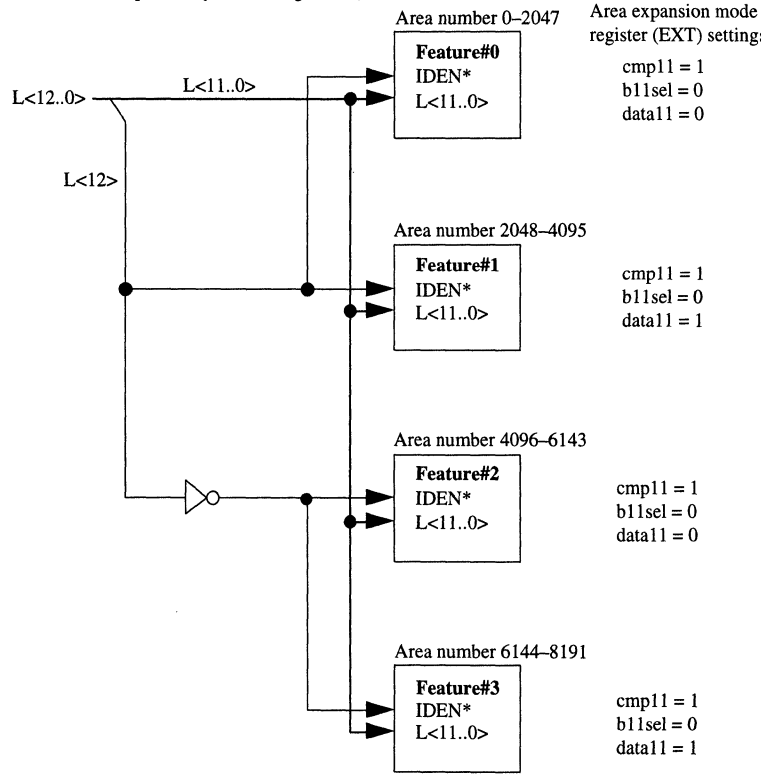
(A) Secondary moment measurement for 2048 separate areas (basic system configuration)



(B) Secondary moment measurement for 4096 separate areas (4096-area expanded system configuration)

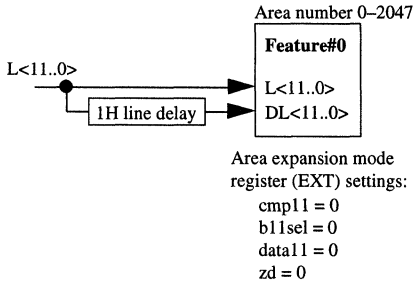


(C) Secondary moment measurement for 8192 separate areas (4096 x 2-area expanded system configuration)

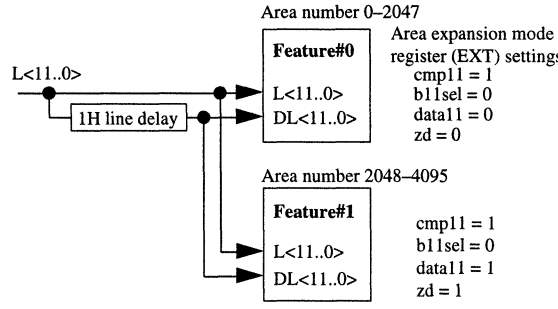


4.3.9 Sample System: Expanding the Number of Areas for Three-Way Simultaneous Perimeter Measurement

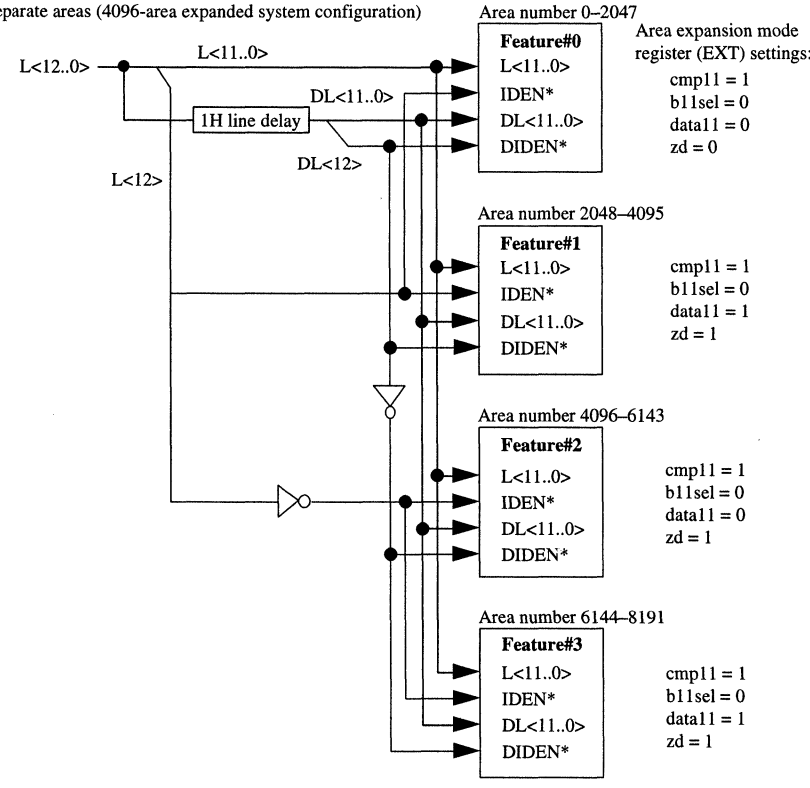
(A) Three-way simultaneous perimeter measurement for 2048 separate areas (basic system configuration)



(B) Three-way simultaneous perimeter measurement for 4096 separate areas (4096-area expanded system configuration)



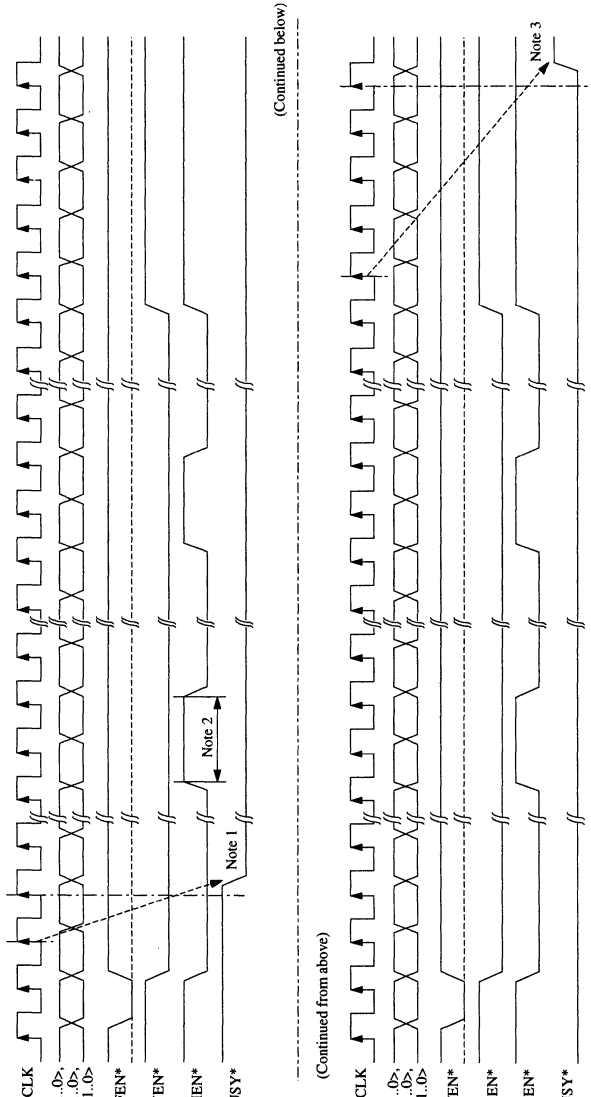
(C) Three-way simultaneous perimeter measurement for 4096 separate areas (4096-area expanded system configuration)



4.4 Operating Timing Charts

4.4.1 Timing with the FEN*, VEN*, and HEN* Signals Enabled

The following timing chart shows how measurement processing operates with the mode register area of interest disable flag (disAOI) set to 0. Note that the timing for the results memory dump output operation differs, as shown on the next page.

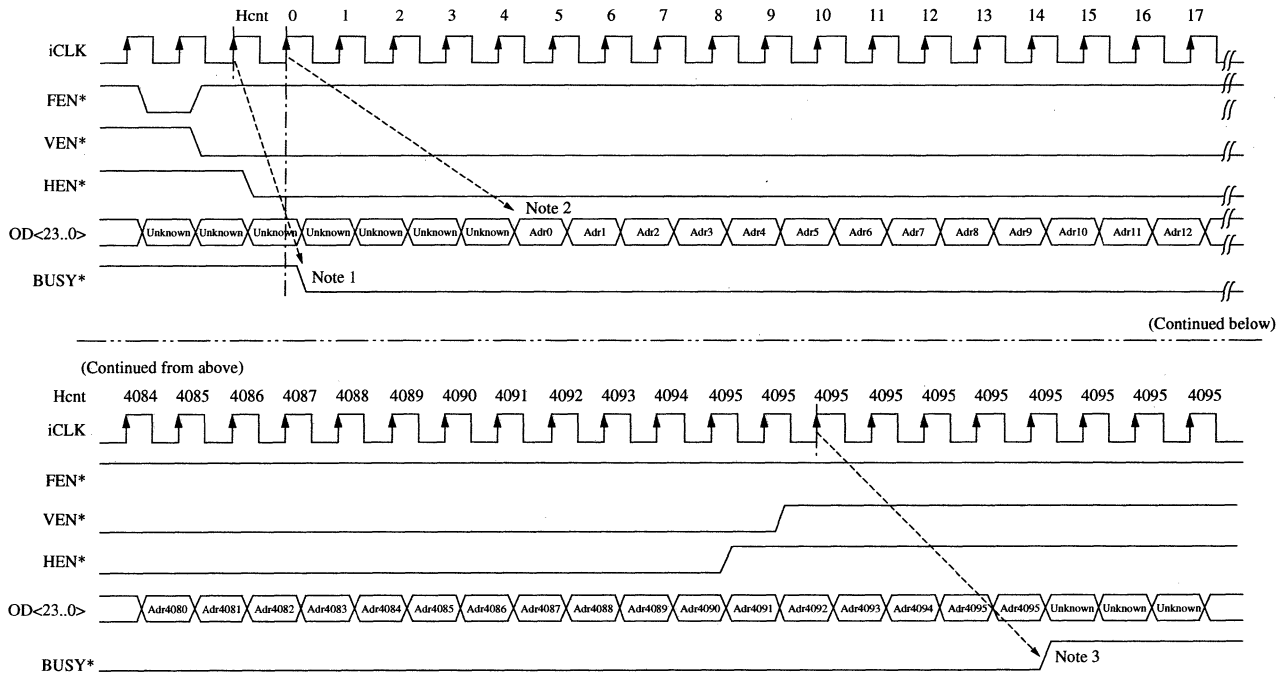


Note 1: When the exec flag is set to 1 to begin execution, it becomes internally valid two clock cycles later. After this, when a valid VEN* signal goes low (as detected at the rise of iCLK), one clock cycle elapses, then the BUSY* signal also goes low to show that internal processing has begun.

Note 2: HEN* must be inactive for at least two clock cycles

Note 3: When field count control is in effect and a value n (≥ 1) is written to the field count control register, when the n^{th} valid VEN* signal goes high (as detected at the rise of iCLK), four clock cycles elapse, then the BUSY* signal also goes high to show that internal processing has ended.

During continuous histogram processing, however, BUSY* goes high when processing ends. When processing is stopped by writing 0 to the exec flag, and the exec flag becomes internally valid two clock cycles later, BUSY* goes high to show that internal processing has ended. Once BUSY* goes high, results can be read from results memory, though results memory cannot be written to until the following cycle.



Note 1: Once the exec flag is set to 1, it becomes internally valid two clock cycles later. Then, when a rise of iCLK detects the change of the first valid VEN* signal from high to low, one clock cycle elapses, then the BUSY* signal also goes low to indicate that internal processing has begun.

Note 2: The change of HEN* from high to low is detected at the rise of iCLK, and the Hcnt coordinate counter is reset. Hcnt then counts iCLK signals as long as HEN* stays low. The Hcnt output value is taken as the address for results memory, and four clock cycles later data can be read through the OD<23..0> signal pin.

Note 3: When a value n (1) has been written to the field count control register and field count control is in effect, then beginning with the rise of iCLK that detects the change of the n^{th} valid VEN* signal from low to high, four clock cycles elapse, then BUSY* also goes high to indicate that internal processing has ended.

4.4.2 Timing with FEN*, VEN*, HEN* Signals Disabled

The following timing chart shows how measurement processing operates with the mode register area of interest disable flag (disAOI) set to 1 (used for look-up table conversion and cumulative histogram processing).

Look-Up Table Conversion Processing

Note 1: Once the exec flag is set to 1 and becomes internally valid two clock cycles later, the BUSY* signal goes low, and data input is accepted over L<11..0> or ID<11..0>. This is how data input on the cycle after BUSY* goes low becomes valid. Therefore, the value of OD<23..0> is undefined for data input up to the cycle in which BUSY* goes low.

Note 2: The delay from L<11..0> or ID<11..0> to OD<23..0> is four clock cycles.

Note 3: Once the exec flag is set to 0 and becomes internally valid two clock cycles later, BUSY* goes high, and data input is no longer accepted over L<11..0> or ID<11..0>. Therefore, the value of OD<23..0> is undefined for data beginning with the cycle in which BUSY* goes high.

Cumulative Histogram Processing

Note 1: Once the exec flag is set to 1 and becomes internally valid two clock cycles later, BUSY* goes low, and cumulative histogram processing begins.

Note 2: When cumulative histogram processing ends, BUSY* goes high. In 4K measurement mode, or 2K measurement mode with power-saving cumulative histogram processing, the BUSY* low period is approximately 4K cycles. In 2K measurement mode with normal cumulative histogram processing, the BUSY* low period is approximately 2K cycles.

4.5 Processing on the Area of Interest Boundary

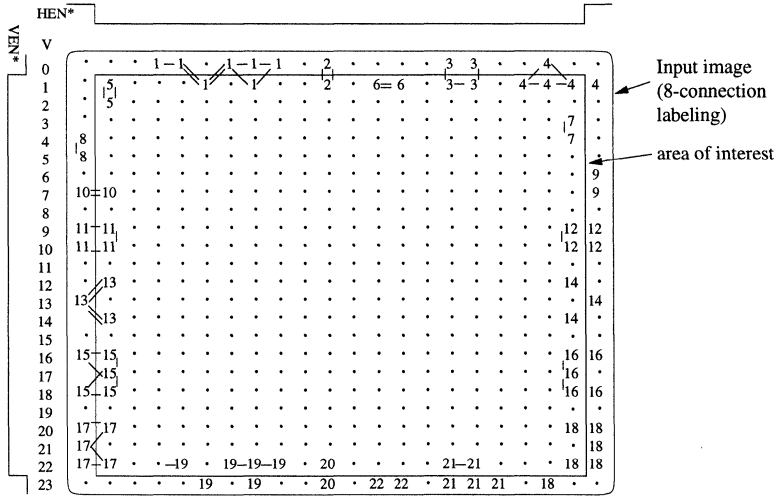
When measuring area perimeters, the Feature chip examines 2x2-pixel local areas to find the boundary between the area containing continuous data and the background, then counts the vertical, horizontal, and diagonal perimeter lengths independently. Also, in run coordinate detection, the Feature chip looks at 2x1-pixel local areas to find the start and run points of the run. Thus, in processing these two-dimensional local areas, the boundary of the AOI itself occasionally appears within a local area. This section describes processing on AOI boundaries.

4.5.1 Perimeter Length Measurement on the Area of Interest Boundary

When measuring area perimeters, the area number (label number) is input through L<11..0>, and the area number of the previous line is input through DL<11..0>. If HEN* and VEN* are both low when data is input through L<11..0>, a 2x2-pixel local area is created from the data values input through L<11..0> and DL<11..0> and from the data values input through L<11..0> and DL<11..0> on the previous clock cycle. This 2x2-pixel local area is then used for perimeter measurement. However, if HEN* or VEN* is high when data is input through L<11..0>, perimeter measurement is not performed on the 2x2-pixel local area created from the same data.

The following diagram shows an example using a labeled input image, and the resulting measurements. The following pages show the timing charts for input through L<11..0> of label value data for lines 1, 22, and 23 in the labeled image.

Sample Labeled Input Image

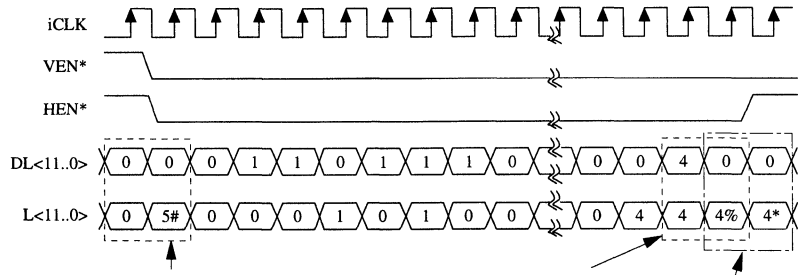


Note: Numerals indicate non-background pixel label values, and dots (•) indicate background pixels (label value 0).

Measurement Results

Label value	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Hperi	1	0	1	2	0	2	0	0	0	2	2	0	0	0	2	0	1	0	3	0	1	0
Vperi	0	2	2	0	2	0	1	1	0	0	1	1	0	0	2	2	0	0	0	0	0	0
Dperi	6	0	0	2	0	0	0	0	0	0	0	0	4	0	2	0	2	0	0	0	0	0

(A) Line V = 1

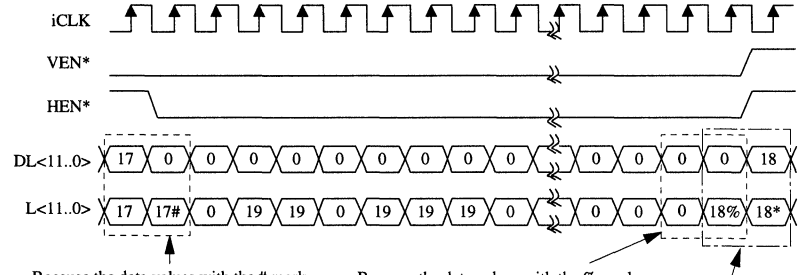


Because the data values with the # mark lie inside the AOI, perimeter measurement is performed on the local area bounded by the dotted lines.

Because the data values with the % mark lie inside the AOI, perimeter measurement is performed on the local area bounded by the dotted lines.

Because the data values with the * mark lie outside the AOI, perimeter measurement is not performed on the local area bounded by the dashed lines.

(B) Line V = 22

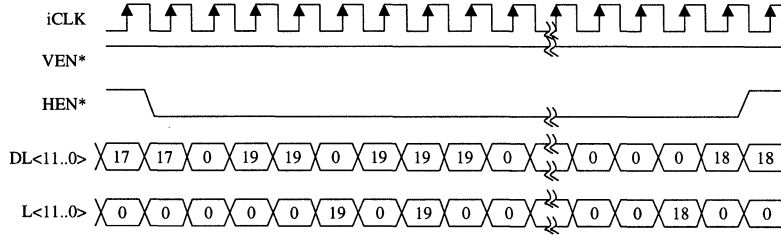


Because the data values with the # mark lie inside the AOI, perimeter measurement is performed on the local area bounded by the dotted lines.

Because the data values with the % mark lie inside the AOI, perimeter measurement is performed on the local area bounded by the dotted lines.

Because the data values with the * mark lie outside the AOI, perimeter measurement is not performed on the local area bounded by the dashed lines.

(C) Line V = 23

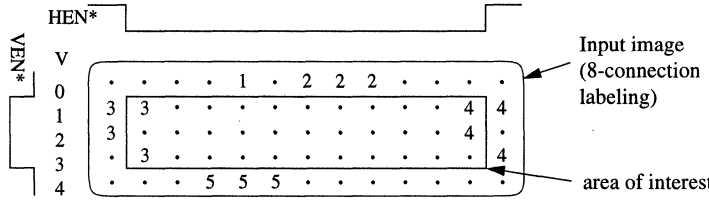


Because the L<11..0> input data for this line lies outside the AOI area (VEN* = high), no perimeter length measurement is performed.

4.5.2 Run Coordinate Detection on the AOI Boundary

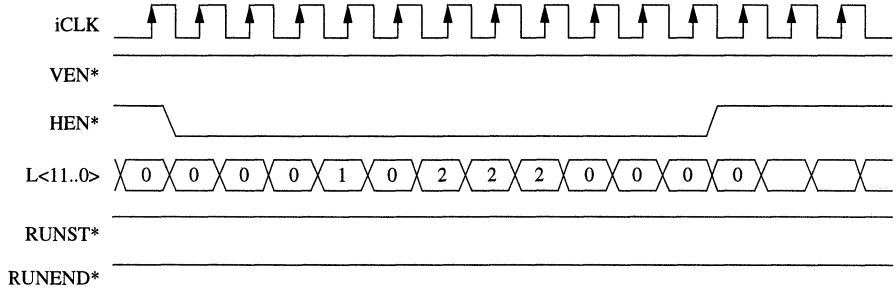
When performing run coordinate detection, area numbers outside the AOI boundary (input through L<11..0>) are treated as 0, the same as background values. The following diagram shows an example using a labeled input image and the resulting measurements, including timing charts.

Sample Labeled Input Image



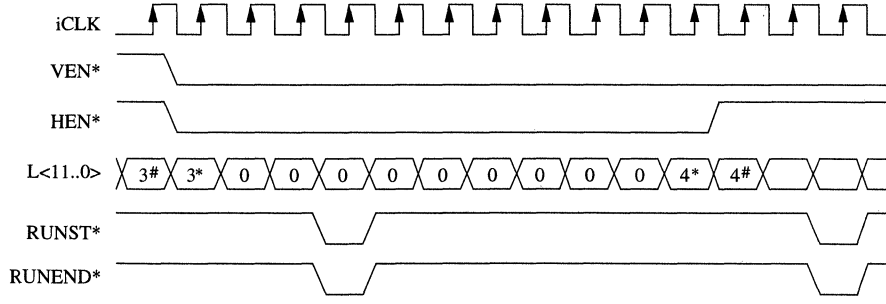
Note: Numerals indicate non-background pixel label values, and dots (•) indicate background pixels (label value 0).

(A) Line V = 0



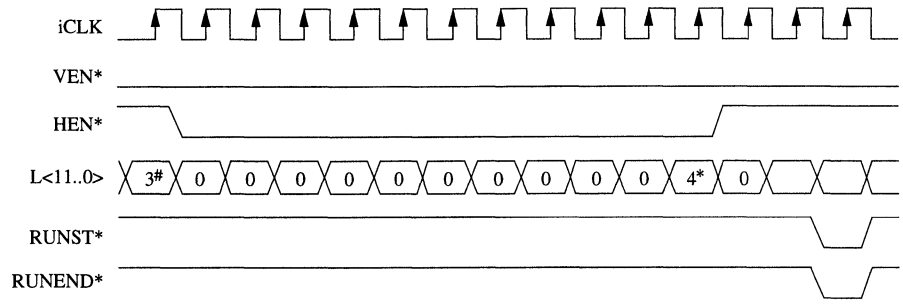
Note: Because the L<11..0> input data for this line lies outside the AOI (VEN* = high), no run coordinate detection is performed.

(B) Line V = 1



Note: Because the areas labeled 3# and 4# lie outside the AOI, they are processed as background values, and RUNST* and RUNEND* signals are output in relation to the areas labeled 3* and 4*.

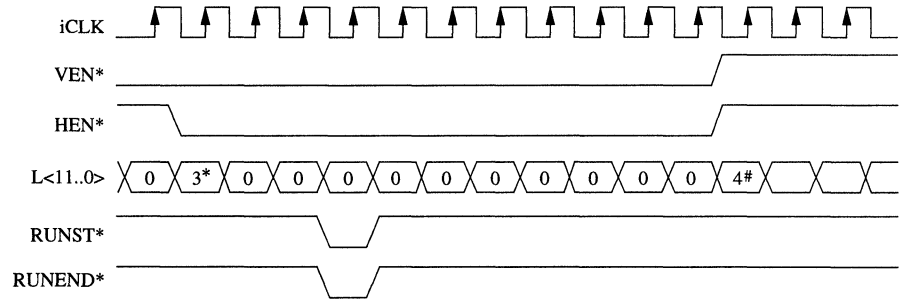
(C) Line V = 2



Note: Because the area labeled 3# lies outside the AOI, it is processed as a background value, and RUNST* and RUNEND* signals are not output.

Note: RUNST* and RUNEND* signals are output in relation to the area labeled 4*, which lies inside the AOI.

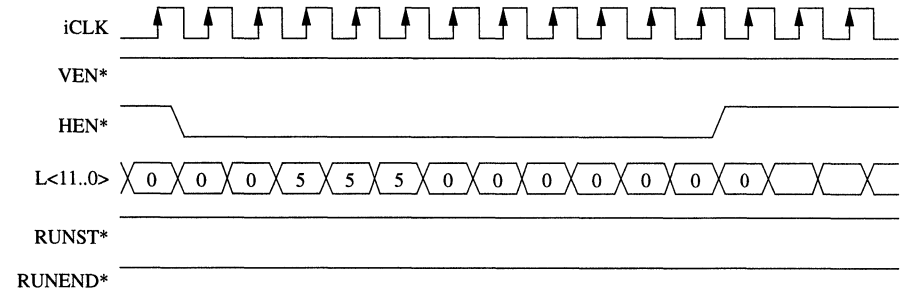
(D) Line V = 3



Note: RUNST* and RUNEND* signals are output in relation to the area labeled 3*, which lies inside the AOI.

Note: Because the area labeled 4# lies outside the AOI, it is processed as a background value, and RUNST* and RUNEND* signals are not output.

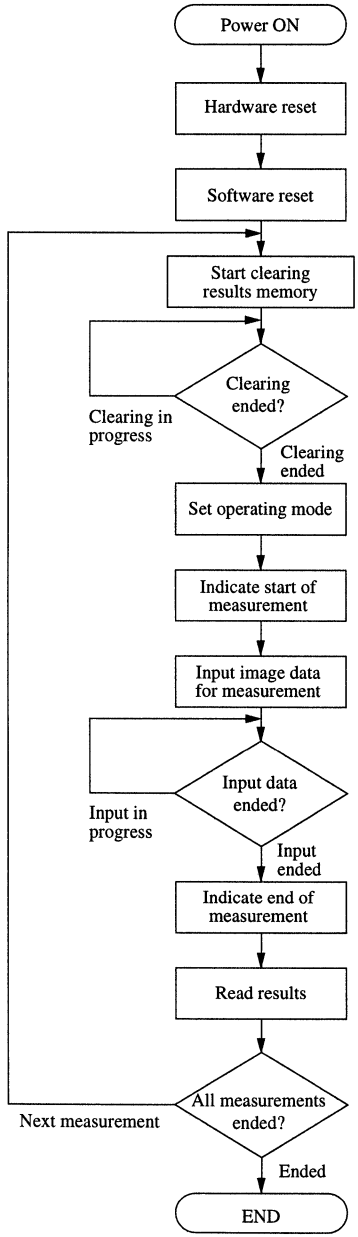
(E) Line V = 4



Note: Because the L<11..0> input data for this line lies outside the AOI (VEN* = high), no run coordinate detection is performed.

4.6 Setup Sequences

4.6.1 Setup for Continuous Extraction of Quantitative Image Characteristics



RST* signal input set low, then returns to high and remains high.

Use reset register rst flag to initiate software reset:
- 1: to execute reset
- 0: to cancel reset

Use results memory to clear results memory register (all values to 0).
Write the number of clear cycles to the lower 6 bits, then set the clear flag (bit 7) to 1 to clear for the designated number of cycles.

Verify the end of clearing through the BUSY* signal or busy flag.

Set registers for operating mode according to type of measurement desired:
For sample settings for various operating modes using the memory address selection register, memory data setting register, adder setting register, etc., see Section 5.3.

Write 1 to the exec flag of the execute register to indicate the start of measurement processing.

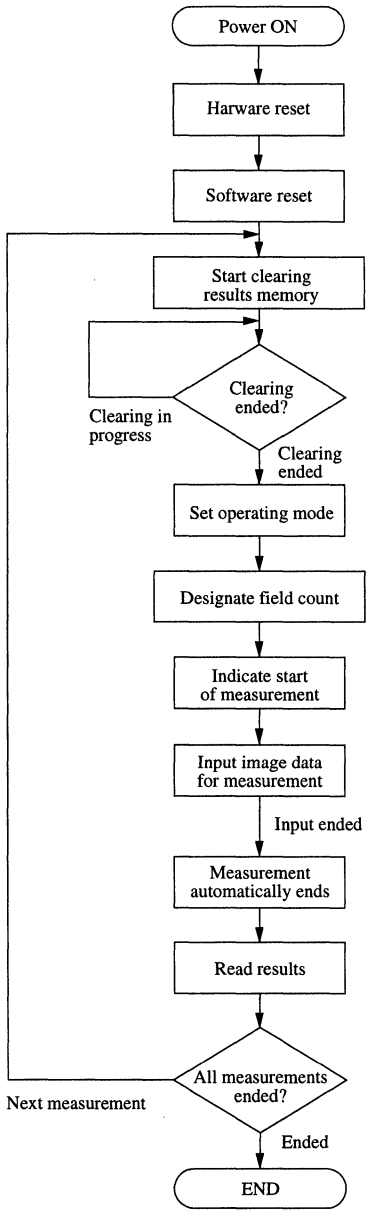
Perform measurement of areas designated by the IDEN*, VEN*, and HEN* signals.

After data input for measurement has ended, verify the end of internal processing by the BUSY* signal or busy flag.

Write 0 to the exec flag to indicate the end of measurement.
This initiates switching to allow the control bus to access results memory.

Access results memory from the control bus, and read the results of measurement.

4.6.2 Setup for Extraction of Quantitative Image Characteristics from a Designated Number of Fields



RST* signal input set low, then returns to high and remains high.

Use reset register rst flag to initiate software reset.
- 1: to execute reset
- 0: to cancel reset

Use results memory to clear contents or results memory register (all values to 0).

Write the number of clear cycles to the lower 6 bits, then set the clear flag (bit 7) to 1 to clear for the designated number of cycles.

Verify end of clearing through the BUSY* signal or busy flag.

Set registers for the operating mode according to type of measurement desired:

For sample settings for various operating modes using various registers, see Section 5.3, "Sample Register Setting for Operating Modes."

Designate the number of fields to be processed using the field count control register.

Write 1 to the exec flag of the execute register to indicate the start of measurement processing.

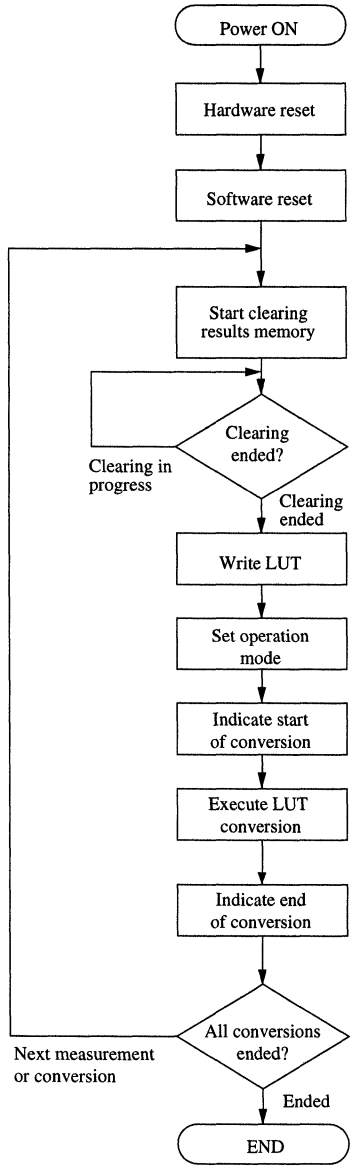
Perform measurement of areas designated by the IDEN*, VEN*, and HEN* signals.

If field count control is used, the exec flag is automatically reset to 0 when the internal processing ends.

This initiates switching to allow the control bus to access results memory.

Access results memory from the control bus, and read the results of measurement

4.6.3 Setup for LUT Processing



RST* signal input set low, then returns to high and remains high.

Use the reset register rst flag to initiate software reset:

- 1: to execute reset
- 0: to cancel reset

Use results memory to clear the contents or results memory register (all values to 0). (It is not necessary to clear results memory if data has been entered to all addresses.)

Write the number of clear cycles to the lower 6 bits, then set the clear flag (bit 7) to 1 to clear for the designated number of cycles.

Verify the end of clearing through the BUSY* signal or busy flag.

Write the contents of the conversion table (LUT) to internal memory.

Set registers for operating mode, according to type of LUT conversion processing desired.

For sample settings for LUT conversion modes, see Section 5.3, "Sample Register Settings for Operating Modes."

Write 1 to the exec flag of the execute register to indicate the start of measurement processing.

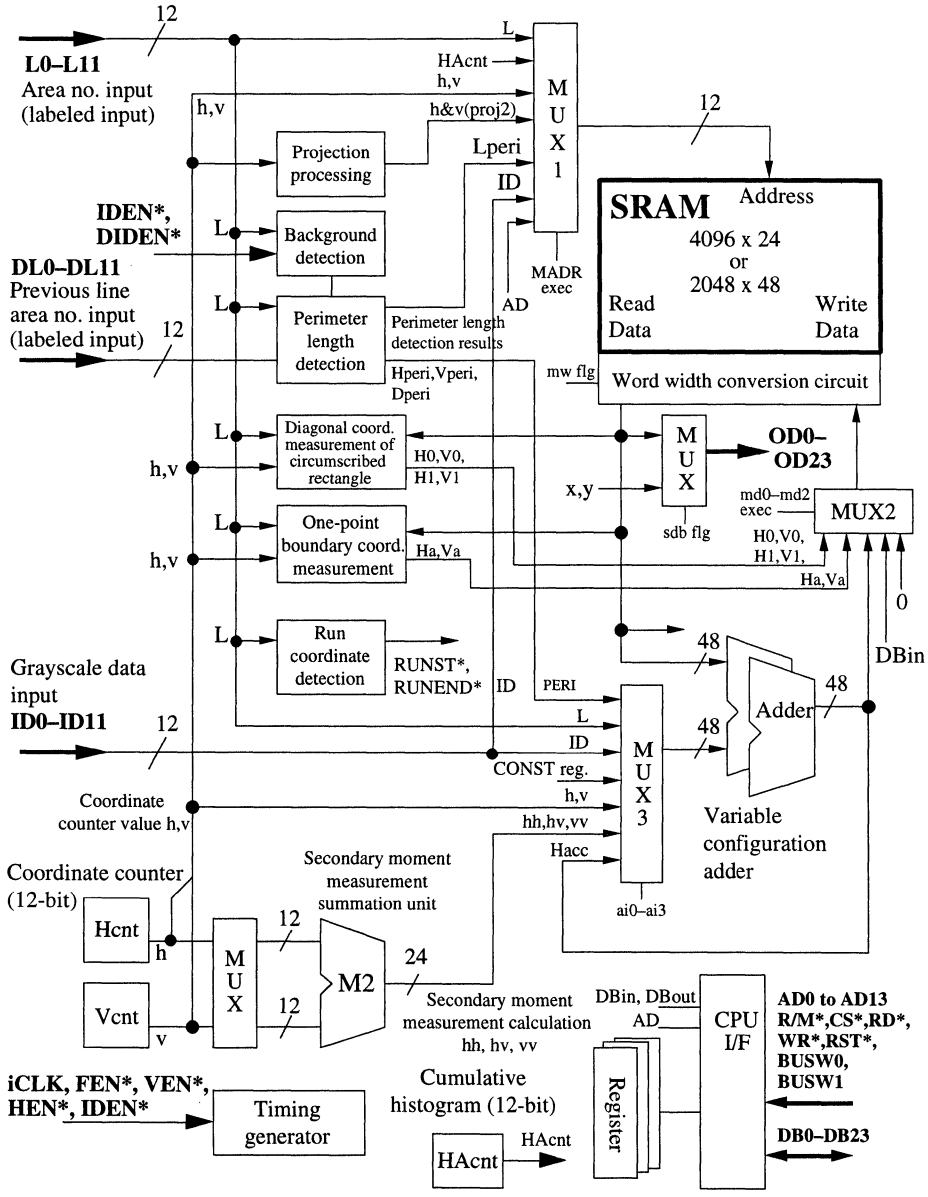
Using the results memory as a conversion table, execute LUT conversion using the memory address input as ID (or L) input and memory data output as OD output.

Write 0 to the exec flag to indicate the end of conversion.

This initiates switching to allow the control bus to access results memory.

Section 5: Internal Configuration

5.1 Internal Block Diagram



Note: This diagram is only intended to describe the general functions of the IP90C18 Feature chip. Not all functions and operations are shown. For full details, refer to the related sections.

5.2 External Signal Connections and LSI Internal Connections for Each Type of Processing

5.2.1 LSI External Pin Signal Connections

L0 to L11

- Weighted surface area measurement: Area number (label value)
- Grayscale histogram processing: 0 (not used)/grayscale value (area weighting coefficient) register setting
- Labeled image area measurement: Area number (label value)/0 (not used) register setting
- Circumscribed rectangle coordinate measurement: Area number (label value)
- Perimeter measurement: Area number (label value)
- One-point boundary coordinate measurement: Area number (label value)
- Run coordinate detection: Area number (label value)
- Primary/secondary moment measurement: Area number (label value)
- Grayscale projection processing: 0 (not used)/grayscale value
- Table conversion: Area number/grayscale value/not used
- Cumulative histogram processing: 0 (not used)
- Results memory dump: 0 (not used)

ID0 to ID11

- Weighted surface area measurement: Grayscale value (area weighting coefficient)
- Grayscale histogram processing: Grayscale value (area weighting coefficient)/ 0 (not used) register setting
- Labeled image area measurement: 0 (not used)/area number (label value) register setting
- Circumscribed rectangle coordinate measurement: 0 (not used)
- Perimeter measurement: 0 (not used)
- One-point boundary coordinate measurement: 0 (not used)
- Run coordinate detection: 0 (not used)
- Primary, secondary moment measurement: 0 (not used)
- Grayscale projection processing: Grayscale value/0 (not used)
- Table conversion: Grayscale value/area number/0 (not used)
- Cumulative histogram processing: 0 (not used)
- Results memory dump: 0 (not used)

Signal pins DL0–DL11 only require area number (label value) data (including a one-line delay) from L0–L11 when measuring perimeters. During other measurements, set these pins high or low as appropriate. For table conversion functions, conversion table address input can be switched to any signal pin Ln (n = 0 to 11) or IDn (n = 0 to 11) by using a register setting. In the same way, address input for grayscale histogram processing and labeled image area measurement can be switched to any signal pin Ln or IDn by using a register setting. Avoid leaving the input signal pins Ln or IDn in unused or floating (unconnected) status. Floating signal pins can cause abnormal current flows in the device.

5.2.2 LSI Internal Memory Signal Connections

SRAM Address

• Weighted surface area measurement:	Area number (label value)	Ln pins
• Grayscale histogram processing:	Grayscale value	IDn pins / Ln pins
• Labeled image area measurement:	Area number (label value)	Ln pins / IDn pins
• Circumscribed rectangle coordinate measurement:	Area number (label value)	Ln pins
• Perimeter measurement:	2 x2 local area number (label value)	Internal output signal Lperi
• One-point boundary coordinate measurement:	Area number (label value)	Ln pins
• Run coordinate detection:	0 (not used)	
• Primary, secondary moment measurement:	Area number (label value)	Ln pins
• Grayscale projection processing:	Coordinate counter output	Hcnt, Vcnt output
• Table conversion:	Grayscale value, or area number, etc.	IDn pins, Ln pins
• Cumulative histogram processing:	Accumulation counter output	HAcnt output
• Results memory dump:	Coordinate counter output	Hcnt output

SRAM Data

• Weighted surface area measurement:	Weighted surface area value for each area Data input: 24, 48-bit adder output
• Grayscale histogram processing:	Pixel count for each grayscale value Data input: 24, 48-bit adder output
• Labeled image area measurement:	Pixel count (surface area) for each label value (area) Data input: 24, 48-bit adder output
• Circumscribed rectangle coordinate measurement:	Diagonal coordinates [(H0, V0), (H1, V1)] for each area Data input: (H0, V0), (H1, V1)
• Perimeter measurement:	Vertical, horizontal, diagonal perimeter count for each area Data input: 16-bit adder output x 3/24, 48-bit adder output
• One-point boundary coordinate measurement:	Boundary point coordinates (Ha, Va) for each area Data input: (Ha, Va)
• Run coordinate detection:	0 (not used)
• Primary/secondary moment measurement:	Moment for each area Data input: 24, 48-bit adder output
• Grayscale projection processing:	Cumulative sum of grayscale values for each row or column Data input: 24, 48-bit adder output
• Table conversion:	Conversion value output Data output: OD0–OD23
• Cumulative histogram processing:	Cumulative frequency distribution Data input: 24, 48-bit adder output
• Results memory dump:	Results memory dump output Data output: OD0–OD23

5.3 Sample Register Settings for Each Measurement

This section presents sample register settings for each type of measurement that uses the IP90C18 Feature chip. Each example assumes that the overflow flag bit generator flag (ovm) in the adder indicator register (FAIN) and all flags in the overflow flag mask register (OVFMASK) are set to 0. The examples also assume that in measurements for which the number of fields is controlled, the field count control register (FLD) is set to 01h to allow processing of only one field. For details about individual registers, see Section 6.

5.3.1 Cumulative Addition Processing by Image Area

5.3.1.1 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	01h
1Ch	I/O control register	IOCTRL	80h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.1.2 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	01h
1Ch	I/O control register	IOCTRL	80h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.
Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.2 Grayscale Histogram Processing

5.3.2.1 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	02h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000001h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.2.2 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	03h (Note 1)
10h	Results memory address select register	MADR	02h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000001h

Note 1: Grayscale values are valid in the range 0–2047.

Note 2: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.3 Labeled Image Surface Area Measurement

5.3.3.1 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000001h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.3.2 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000001h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.4 Circumscribed Rectangle Diagonal Coordinate Measurement

5.3.4.1 Simultaneous Measurement of Lower Left and Upper Right Coordinates; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C3h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.4.2 Lower Left Coordinate Measurement; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	44h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.4.3 Upper Right Coordinate Measurement; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	45h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.5 Perimeter Length Measurement

5.3.5.1 Simultaneous Measurement of All Three Directional Components in an 8-Connection Labeled Image; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	03h (Note 2)
14h	Results memory data indicator register	MDATA	C2h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.2 Measurement of the Horizontal Component in an 8-Connection Labeled Image; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	03h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	08h
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.3 Measurement of the Vertical Component in an 8-Connection Labeled Image; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	03h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	09h
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.4 Measurement of Diagonal Component in an 8-Connection Labeled Image; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	03h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	0Ah
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.5 Simultaneous Measurement of All Three Directional Components in a 4-Connection Labeled Image; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	02h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	03h (Note 2)
14h	Results memory data indicator register	MDATA	C2h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.6 Measurement of the Horizontal Component in a 4-Connection Labeled Image; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	02h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	03h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	08h
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.7 Measurement of the Vertical Component in a 4-Connection Labeled Image; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	02h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	03h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	09h
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.5.8 Measurement of the Diagonal Component in a 4-Connection Labeled Image; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	02h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	03h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	0Ah
1Ch	I/O control register	IOCTRL	20h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. Input area numbers from previous (delayed) line to pins DL0–DL11. ID0–ID11 are not used.

5.3.6 Measurement of One-point Boundary Coordinates

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	46h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.7 Run Coordinate Detection

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	20h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	00h
14h	Results memory data indicator register	MDATA	00h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	BFh
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.8 Primary Moment Measurement

5.3.8.1 Mh; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	03h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.8.2 Mv; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	04h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.8.3 Mh; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	03h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.8.4 Mv; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	04h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.9 Secondary Moment Measurement

5.3.9.1 Mhh; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	05h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.9.2 Mhv; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	06h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.9.3 Mvv; 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	01h (Note 1)
10h	Results memory address select register	MADR	01h (Note 2)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	07h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Area numbers are valid in the range 0–2047.

Note 2: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.9.4 Mhh; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	05h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.9.5 Mhv; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	06h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.9.6 Mvv; 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	07h
1Ch	I/O control register	IOCTRL	A0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.10 Grayscale Projection Processing

5.3.10.1 Two-Axis Simultaneous Projection Processing

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	06h
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	01h (Note 1)
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.10.2 Single-Axis Projection Processing on H-Axis: 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	04h
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	01h (Note 1)
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.10.3 Single-Axis Projection Processing on V-Axis: 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	05h
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	01h (Note 1)
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.10.4 Single-Axis Projection Processing on H-Axis: 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	04h
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	01h (Note 1)
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.10.5 Single-Axis Projection Processing on V-Axis: 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	05h
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	01h (Note 1)
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.11 Look-Up Table Conversion Processing

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	01h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	01h (Note 1)
14h	Results memory data indicator register	MDATA	00h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	AFh
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	00h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input area numbers to pins L0–L11. ID0–ID11 are not used.

5.3.12 Cumulative Histogram Processing

5.3.12.1 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	05h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	07h
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	0Bh
1Ch	I/O control register	IOCTRL	E0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	00h
30h, 34h	Constant register	CONST	000000000000h

5.3.12.2 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	0Dh (Note 1)
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	07h
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	0Bh
1Ch	I/O control register	IOCTRL	E0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	00h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Set the low power cumulative histogram processing mode flag to 1.

5.3.13 Continuous Cumulative Histogram Processing

5.3.13.1 4K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	10h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	02h (Note 1)
14h	Results memory data indicator register	MDATA	41h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Note 1: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.13.2 2K Area Measurement Mode

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	18h (Note 1)
0Ch	Area extension mode register	EXT	05h (Note 2)
10h	Results memory address select register	MADR	02h (Note 3)
14h	Results memory data indicator register	MDATA	C1h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	C0h
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000001h

Note 1: Set the low power cumulative histogram processing mode flag to 1.

Note 2: Grayscale values are valid in the range 0–2047

Note 3: Input grayscale values to pins ID0–ID11. L0–L11 are not used.

5.3.14 Results Memory Dump

Address (hex)	Register Name	Abbreviation	Setting (hex)
04h	Reset register	RST	00h
08h	Mode register	MOD	00h
0Ch	Area extension mode register	EXT	00h
10h	Results memory address select register	MADR	04h
14h	Results memory data indicator register	MDATA	00h
18h	Adder indicator register	FAIN	00h
1Ch	I/O control register	IOCTRL	EFh
20h	Overflow status register	OVFSTAT	00h
24h	Overflow flag mask register	OVFMASK	00h
28h	Results memory clear register	CLRmem	00h
2Ch	Field count control register	FLD	01h
30h, 34h	Constant register	CONST	000000000000h

Section 6: Address Maps and Registers

6.1 Address Map and Register List

The following table is an address map for the results memory and register addresses as seen by the control bus. Byte address format is used by the address map, so that one address is allocated to each 8-bit section of address space. Thus, changing the byte width of the data bus does not affect the address map. The results memory and control registers are stored in separate address spaces, and either can be accessed by switching the value of the memory/register select signal pin (R/M*). For example, to access results memory through the control bus, set the R/M* signal low; to enter register settings or read register status, set R/M* high.

The switching signal from the R/M* pin is interpreted as the equivalent of the high byte of other address signals. This allows results memory and the registers to be mapped in the same space by processing the R/M* signal on AD14, the MSB, or the address signal (AD0 to AD13), and mapping results memory in the first half and the registers in the second half of that space.

For details about address mapping within results memory, see Section 6.17, "Results Memory Address Map." Results memory has a configuration of 4K words x 24-bits/word, but is mapped as if the configuration were 4K words x 32-bits/word. One of every four bytes read represents an empty column, and has the value 0.

Address (hex)	Abbreviation	Width (bits)	Function (overview)
0000–3FFF	MEM	24/32	Results memory 4K words x 24-bit (24 effective of 32 bits)

Register space is mapped as shown on the following page. Following a reset, all register values are 0. Registers are mapped every four bytes, starting with the lowest addresses. Address values are based on byte access (8-bit units). The lowest 8 bits of each address are allocated to AD1, 0 = 00B, the middle 8 bits to AD1, 0 = 01B, and the upper 8 bits to AD1, 0 = 10B. AD2–AD5 is shared with the memory address, and used for register selection (decoding). In double-byte access (16-bit units), the lower 8 bits are allocated to AD1 = 0, the upper 8 bits are allocated to AD1 = 1, and AD2–AD5 are decoded. AD0 is not used. In word access (24-bit units), AD0 and AD1 are not used, and AD2–AD5 are decoded.

During register access (R/M* = high), address signals AD6–AD13 are not used.

All registers are cleared to 0 when reset.

Address (hex)	Abbreviation	Width (bit)	Function (overview)
00h	EXEC	8	Execute register
04h	RST	8	Reset register
08h	MOD	8	Mode register
0Ch	EXT	8	Area extension mode register
10h	MADR	8	Results memory address select register
14h	MDATA	8	Results memory data indicator register
18h	FAIN	8	Adder indicator register
1Ch	IOCTRL	8	I/O control register
20h	OVFSTAT	8	Overflow status register
24h	OVFMASK	8	Overflow flag mask register
28h	CLRmem	8	Results memory clear register
2Ch	FLD	8	Field count control register
30h	CONST	48	Constant register
34h			
38h	LBL00	12	Origin area number register
3C	LSI-TEST	8	LSI internal test register

6.2 Execute Register

The execute register starts and stops measurement processing.

When a 1 is written to the execute flag (exec), measurement processing begins following the input of the first vertical data enable (VEN*) signal controlled by the FEN* signal. In actual operation, after the software writes 1 to the exec flag, the Feature chip's internal synchronous differentiation circuit detects that the external VEN* signal has gone low: this marks the start of a field (and clears the vertical and horizontal coordinate counters Vcnt and Hcnt). The chip then processes the raster scan image pixels that are input while the area enable signals VEN* and HEN* are low. Note, however, that the disAOI flag in the mode register can be used to ignore the area enable signals (HEN*, VEN*, and FEN*).

When 0 is written to the exec flag to stop processing, two clock cycles elapse before the signal becomes internally valid and processing actually stops.

When the exec flag is set to 1, results memory cannot be accessed through the control bus. To access results memory, set the exec flag to 0.

EXEC Register address 00h

Execute register Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	0	busy	exec

busy: Execute status flag (read only)

This flag gives execution status as does the external BUSY* pin, except that the BUSY* pin uses inverse logic, whereas the busy flag uses positive logic with the following values:

busy = 1: Processing under way

busy = 0: Stop/standby status

After a reset, when the exec and busy flags are set to 0. After 1 is written to exec and the chip waits for the assertion of the external vertical area signals (FEN*, VEN*), the busy flag is still set to 0 (standby status). When the vertical area signals (FEN*, VEN*) are asserted, the busy flag changes to 1 (processing under way). When processing ends and the exec flag changes to 0, the busy flag also changes to 0 (stop status). When processing is interrupted by writing 0 to the exec flag, the busy flag returns to 0. However, if the disAOI flag in the mode register is set to 1, the area enable signals (HEN*, VEN*, and FEN*) are ignored, the exec flag is set to 1, and after two clock cycles the busy flag is also set to 1.

The busy flag is also set to 1 while the results memory is being cleared. Therefore, the combination of busy = 1 and exec = 0 occurs after a start-execution setting while awaiting a valid area, and also while clearing results memory.

exec: Execute flag (read/write)

exec = 1: Start measurement processing

exec = 0: Stop measurement processing

Execution begins when 1 is written to the exec flag. If field count controls are in effect, processing stops automatically after the designated number of fields are processed, and the exec flag is then cleared to 0. If field count controls are not in effect, then processing stops when 0 is written to the exec flag. Execution continues until 0 is written to the exec flag, a reset signal is input at the RST* pin, or software reset is initiated from the reset register RST.

Writing to registers is asynchronous with respect to the image clock. However, to prevent metastable conditions, values written to the exec flag must pass through the shift register before becoming valid. This asynchronous process imposes a delay of two clock cycles.

6.3 Reset Register

The reset register executes software resets. This process is AND-linked with the hardware reset, which is initiated from the reset signal pin (RST*). All registers in the IP90C18 are reset by this process and cleared to 0, except for results memory. To execute a software reset, write 1 to the register (the reset is executed within three iCLK cycles), then write 0 to cancel the reset condition (the reset cancel is completed within three iCLK cycles). Set all unused bits to 0. Hardware resets set the reset register to 00h.

Use the results memory clear register to clear results memory. Results memory cannot be cleared from the reset signal pin RST* or through a software reset by this reset flag.

RST Register address 04h

Reset register Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	0	0	rst

rst: Reset flag (read/write)

rst = 1: Reset

rst = 0: Cancel reset

6.4 Mode Register

The mode register selects the IP90C18's execution mode. All necessary settings must be entered in this register before setting the execute flag in the execute register.

All values in this register are reset to 00h.

MOD Register address 08h (all bits read/write enabled)
Mode register Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	disVcnt	runM	rH	HaccLP	HaccM	peri4/8*	disAOI

disVcnt: Vertical coordinate counter Vcnt reset stop flag

When this flag is set to 0, the one-clock-cycle pulse VENpls is generated internally by synchronous differentiation when the VEN* signal goes low as detected at the rise of iCLK, and becomes the Vcnt reset signal for all fields. (For details see Section 4.2.2, "VEN* and HEN* Signals and Spatial Coordinates.")

However, when this flag is set to 1, the VENpls pulse of the first field becomes the Vcnt reset signal, and the VENpls pulses of all subsequent fields are reset by the Vcnt signal. Therefore, when this flag is set to 1, and the processing of multiple fields has been enabled, the system does not count HEN* signals within the AOI of each field, but rather counts all HEN* signals in the AOIs of valid fields. (For examples of how to use of this flag in processing, see Sections 6.19.4.2 and 6.19.4.5.)

disVcnt = 1: Vcnt reset signal disabled

disVcnt = 0: Vcnt reset signal enabled

runM: Run coordinate detection mode flag

This flag controls the run coordinate detection circuit, which outputs the run-start signal RUNST* and the run-end signal RUNEND*. To execute run coordinate detection, set this flag to 1.

runM = 1: Enable run coordinate detection circuit

runM = 0: Disable run coordinate detection circuit. The external signals RUNST* and RUNEND* are deasserted (high level).

rH: Continuous cumulative histogram processing mode flag

In this mode, histogram processing (cumulative processing within each area) is followed automatically by cumulative histogram processing (continuous cumulative histogram processing mode).

rH = 1: After the end of cumulative histogram processing by area or histogram measurement, the system automatically proceeds to cumulative histogram processing. Within the related circuits, the values of the disAOI and HaccM flags are set to 1 to start execution after image data input measurement processing ends: this sets the processing field count for histogram processing. However, the actual values of disAOI and HaccM in the mode register are ignored but left unchanged. At the end of cumulative histogram processing, disAOI and HaccM are enabled again.

rH = 0: Normal processing mode. There is no link to cumulative histogram processing.

HaccLP: Low power cumulative histogram processing flag

At every clock cycle, cumulative histogram processing reads from memory and performs addition and writing operations, which requires heavy use of internal operating circuits. Because of this, histogram measurement in 2K area measurement mode at high operating frequencies generates considerable heat, which can affect the long-term reliability of the device. To avoid this problem, set this flag to 1 when using cumulative histogram processing in 2K area measurement mode at clock frequencies greater than 25 MHz. This will reduce processing speed by half, and also reduce the power consumed in cumulative histogram processing by approximately one half, thus lowering heat generation and preventing loss of reliability.

HaccP = 1: Low-power cumulative histogram processing mode. Processing is performed at half speed.

HaccP = 0: Normal cumulative histogram processing mode.

HaccM: Cumulative histogram processing mode flag

HaccM = 1: Cumulative histogram processing mode.

HaccM = 0: Normal processing mode. Cumulative histogram processing is not performed.

Cumulative histogram processing mode performs these tasks:

- Adder input processing is changed to cumulative addition, though the values in the adder selection register FAIN are not changed.
- The values in the results memory selection register MADR are forcibly set to 07h, and results memory addresses are connected to the cumulative histogram processing counter HAcnt. In this case values in the MADR register are ignored. The values in the MADR register become valid when HaccM = 0.

peri4/8*: Perimeter length measurement label selection flag (read/write enabled)

This flag determines the directions in which label linking is recognized for perimeter length measurement.

peri4/8* = 1: Perimeter length measurement applied to 4-connection labeled images.

peri4/8* = 0: Perimeter length measurement applied to 8-connection labeled images.

disAOI: Area of interest signal (HEN*, VEN*, FEN*) disable flag

This flag enables execution whenever the exec flag is set to 1, regardless of the activity of the area of interest signals HEN*, VEN*, and FEN*. Cumulative histogram processing and look-up table processing differ from other measurement processes in that they do not require field start point information. By allowing the chip to execute at any time, this flag causes the system to effectively ignore changes in the VEN* signal, and lets processing be executed by setting the exec flag.

disAOI = 1: Disable area of interest signals

disAOI = 0: Enable area of interest signals

6.5 Area Extension Mode Register

The area extension mode register contains area extension mode settings when operating in 2K area measurement mode. Note that the cmp11, b11sel, and data11 flags have no function in 4K area measurement mode.

All values in this register are reset to 00h.

EXT — Register address 0Ch (all bits read/write enabled)
Area extension mode register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
zd	0	0	0	0	data11	b11sel	cmp11

cmp11: High bit (ID11/L11) comparison mode flag for simplified extension

2K area measurement mode does not use the high bit (ID11/L11) of grayscale image data input or area number input. This highest bit can therefore be used as a simple extender to extend 2K area measurement processing of grayscale image data input or area number input over multiple chips. For details, see Section 4.3, "Sample Configuration: Expanded System." The cmp11 bit determines whether the highest bit of either the grayscale image data input or area number input is used as a simple extender.

When cmp11 is set to 1, the value of the ID11 or L11 signal is compared to the value of data11: If the values are equal, measurement processing is executed with mapping between the values 0 and 2047. If the values are not equal, measurement processing is stopped just as when IDEN* is high or treated as background input. When cmp11 is reset to 0, the high bit not used in 2K area measurement mode is ignored (as in normal processing). Normal processing does not use the values of the upper bit select flag for simplified extension (b11sel) or the upper bit comparison value flag for simplified extension (data11).

cmp11 = 1: Simplified extension ID11/L11 comparison mode

cmp11 = 0: Normal processing mode ID11/L11 ignored

b11sel: High byte (ID11/L11) select flag for simplified extension

2K area processing mode does not use the highest bit (ID11/L11) of grayscale value image data input and area number input is not needed. This bit therefore provides an easy way to expand processing through the use of multiple chips while operating in 2K area processing mode. For details, see Section 4.3, "Sample Configuration: Expanded System." The b11sel bit determines which bit is used for simplified extension: the highest bit of the grayscale image data input signal, or the highest bit of the area number input signal.

b11sel = 1: ID11 comparison mode

b11sel = 0: L11 comparison mode

data11: High byte (ID11/L11) comparison flag for simplified extension

data11 = 1 Measure when ID11/L11 = 1

data11 = 0 Measure when ID11/L11 = 0

zd: Background value pixel processing flag

This bit enables extended processing areas through the use of multiple chips or multiple measurement cycles. When extending the area for processes (such as run coordinate measurement or perimeter measurement) in which the background value 000h has special significance, this bit controls whether the input value 000h is processed as background, or as the lowest of the extended area numbers through the use of the background value identification signal IDEN* (normal measurement processing).

In other words, to view the input value 000h as background in 4K area measurement mode (using the values 0–4095), set zd to 0. In an extended configuration using the values 4096–8191, for example, the input value 000h must be processed as pixel value 4096 rather than as background, so zd would be set to 1. In this case, the extension chip recognizes background pixels by using the area decode signal input through the IDEN* signal pin. (For details, see Section 4.3 "Sample Configuration: Expanded System."

zd = 1: Process area number input data 000h as normal data

zd = 0: Process area number input data 000h as background data

6.6 Results Memory Address Selection Register

This register controls the input of address data to results memory. The IP90C18 can switch among many feature extraction processes by using register settings (including this one) to choose various combinations of the chip's three multiplexers.

Set all undefined bits to 0. At reset, all values become 00h.

MADR — Register address 10h (all bits read/write enabled)

Results Memory Address Selection Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	0	0	0	0	madr2	madr1	madr0

madr2, madr1, madr0: Results memory address selection flags

Value (madr2–madr0)	Symbol	Parameter
0 (000)	CONST	Constant register value CONST
1 (001)	L	Area number input
2 (010)	ID	Grayscale value input
3 (011)	Lperi	2 x 2 local area number for perimeter measurement
4 (100)	Hcnt	H coordinate value, horizontal coordinate counter
5 (101)	Vcnt	V coordinate value, vertical coordinate counter
6 (110)	proj2	H, V-axis simultaneous projection processing mode
7 (111)	HAcnt	Counter for Hacc

Note: The exec flag controls switching between bus access from the CPU/external devices and memory access during processing. During access from the control bus, the MADR register has no effect. Before accessing results memory from the control bus during image measurement processing, write 0 to the exec flag to stop processing (access to image results memory is disabled during processing).

6.7 Results Memory Data Selection Register

This register controls the input of data to results memory. The IP90C18 can switch among many feature extraction processes by using register settings (including this one) to choose various combinations of the chip's three multiplexers.

Set all undefined bits to 0. At reset, all values become 00h.

MDATA — Register address 14h (all bits read/write enabled)

Results Memory Data Selection Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
mw	we	0	0	0	md2	md1	md0

mw: Memory storage capacity selection flag

- mw = 0: 4096 word (24-bit width) in 4K area measurement mode
- mw = 1: 2048 word (48-bit width) in 2K area measurement mode

we: Memory write enable flag

- we = 0: Results memory is read-only, write disabled (LUT etc.)
- we = 1: Results memory is write-enabled (normal processing)

md2, md1, md0: Memory write data selection flags

Value (md2–md0)	Symbol	Parameter
000	zero	Input value fixed at 0
001	FA1	24, 48-bit adder output
010	FA2	16-bit adder output x 3 (note 1)
011	HV01	Circumscribed rectangle coordinates (H0, V0) (H1, V1)
100	H0V0	Circumscribed rectangle coordinates (H0, V0)
101	H1V1	Circumscribed rectangle coordinates (H1, V1)
110	HaVa	Area boundary 1-point coordinates (Ha, Va)
111	CONST	Constant register value CONST

Note 1: Used for three-direction simultaneous perimeter measurement. Executes three types of addition processes: Hperi, Vperi, and Dperi. For perimeter measurement in one direction only, select either 24-bit or 48-bit adder output.

Note 2: The exec flag controls switching between bus access from the CPU/external devices and memory access during image processing. During access from the control bus, the mw, we, and md flags have no effect. Before accessing results memory from the control bus during image measurement processing, write 0 to the exec flag to stop processing (access to image results memory is disabled during processing).

6.8 Adder Selection Register

This register selects the input and configuration of the variable configuration adder. The IP90C18 can switch among many feature extraction processes by using register settings (including this one) to choose various combinations of the chip's three multiplexers. This is done by selecting one of the adder inputs from among the signals listed in the following table. The other input is connected to the data output from results memory.

Set all undefined bits to 0. At reset, all values become 00h.

FAIN — Register address 18h (all bits read/write enabled)

Adder Selection Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
ovm	0	0	0	ai3	ai2	ai1	ai0

ovm: Overflow flag bit generator flag

The MSB of results memory data (the 24th or 48th bit, depending on the setting of the memory data width flag mw in the results memory data selection register MDATA) is used as an overflow flag, and contains the overflow data generated from cumulative addition at each address. The effective data width of results memory is therefore reduced by one bit, and so the maximum value that can be processed (maximum cumulative addition result) is also reduced by one bit.

ovm = 0: Overflow flags are common to all areas, and are not stored in results memory

ovm = 1: Overflow flags are handled as the MSB of data stored in results memory, and store the overflow status of each separate area

ai3, ai2, ai1, ai0: Adder input selection flag

Value (ai3, ai2, ai1, ai0)	Symbol	Parameter
0000	CONST	Constant register value CONST
0001	ID	Grayscale value input
0010	L	Area number input
0011	H	H-coordinate value, for measuring primary moment
0100	V	V-coordinate value, for measuring primary moment
0101	HH	Square of H-coordinate value, for measuring secondary moment
0110	HV	Product of H and V coordinate values, for measuring secondary moment
0111	VV	Square of V-coordinate value, for measuring secondary moment
1000	Hperi	Horizontal perimeter measurement,
1001	Vperi	Vertical perimeter measurement,
1010	Dperi	Diagonal perimeter measurement,
1011	Hacc	Cumulative histogram value, used in cumulative histogram measurement

6.9 Input/Output Control Register

The input/output control register controls the image data output pins OD<23..0>, as well as data input through image data input pins L<11..0>, DL<11..0>, and ID<11..0>.

Set all undefined bits to zero. At reset, all values become 00h.

IOCTRL — Register address 1Ch (all bits read/write enabled)
Input/Output Control Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
dl	l	id	sod	oden3	oden2	oden1	oden0

dl: Delay area number input reset flag

- dl = 1: Reset delay area number input data to 0 regardless of the delay area number input pins DL<11..0> (Note 1)
- dl = 0: Input and process delay area number input data from delay area number input pins DL<11..0>

l: Area number input reset flag

- l = 1: Reset area number input data to 0 regardless of the area number input pins L<11..0> (Note 1)
- l = 0: Input and process area number data from area number input pins L<11..0>

id: Grayscale value input reset flag

- id = 1: Reset grayscale input data values to 0 regardless of the grayscale input pins ID<11..0> (Note 1)
- id = 0: Input and process grayscale data from grayscale input pins ID<11..0>

sod: Image data output select flag

This flag selects which type of data is output from image data output pins OD<23..0>:

- sod = 1: Output coordinate data from coordinate counters
- sod = 0: Output read data from results memory

oden0, oden1, oden2, oden3: Results output enable flags

These flags determine whether the output from the image data output signal pins OD<23..0> is fixed at high level or enabled. However, the drive or hi-Z signal from the ODEN* signal pin takes priority over this setting. The oden0 flag controls the 8-bit signal OD<7..0>, oden1 controls the 4 bits OD<11..8>, oden2 controls the 4 bits OD<15..12>, and oden 3 controls the 8 bits OD<23..16>.

$oden_n$ = 1: Image data output is enabled

$oden_n$ = 0: Image data output pins OD<23..0> are fixed at high level output (note 1)

Note 1: These settings are used in power-saving mode to reduce the power consumed by the chip.

Each of these selections creates a logical gate on input pin routes to eliminate unnecessary switching and thereby reduce power consumption. This is also effective in controlling noise on output pins, so when a particular output pin is not used, leave the pin open or use the appropriate flags to prevent switching.

Because input voltage to each input pin is not clipped at that pin's connection level, note that similar prohibitions apply to pin status that are not allowed in CMOS logic, such as input pin signals in intermediate (indeterminate) or floating state.

6.10 Overflow Status Register

The overflow status register reads the status of overflow conditions that occur during processing. When an overflow occurs during processing, the bit corresponding to that type of overflow is set to 1. However, if the corresponding mask flag in the overflow flag mask register has been set to 1 (masked), the overflow signal OVF* is not asserted, but remains high. Whenever any of the bits in the overflow status register is set to 1, and the corresponding mask flag is 0 (unmasked), the OVF* signal is asserted (low-level output). The overflow status register must by forcibly cleared by writing 0 to it, except for hardware resets initiated by the RST* signal or software resets by the reset register.

OVFSTAT — Register address 20h (all bits read/write enabled)

Overflow Status Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
ovfdp	ovfvp	ovfhp	ovfad	ovfvw4K	ovfvw2K	ovfhw4K	ovfhw2K

ovfdp: Perimeter diagonal component 16-bit adder output overflow flag

ovfdp = 1: Overflow

ovfdp = 0: No overflow

ovfvp: Perimeter vertical component 16-bit adder output overflow flag

ovfvp = 1: Overflow

ovfvp = 0: No overflow

ovfhp: Perimeter horizontal component 16-bit adder output overflow flag
- ovfhp = 1: Overflow
- ovfhp = 0: No overflow

ovfad: Adder output overflow flag
- ovfad = 1: Overflow
- ovfad = 0: No overflow

ovfvw4K: Vcnt overflow flag for vertical processing screen size in 4K area mode
- ovfhw4K = 1: Overflow: set when Vcnt exceeds FFFh
- ovfhw4K = 0: No overflow

ovfvw2K: Vcnt overflow flag for vertical processing screen size in 2K area mode
- ovfhw2K = 1: Overflow: set when Vcnt exceeds 7FFh
- ovfhw2K = 0: No overflow

ovfhw4K: Hcnt overflow flag for horizontal processing screen size in 4K area mode
- ovfhw4K = 1: Overflow: set when Hcnt exceeds FFFh
- ovfhw4K = 0: No overflow

ovfhw2K: Hcnt overflow flag for horizontal processing screen size in 2K area mode
- ovfhw2K = 1: Overflow: set when Hcnt exceeds 7FFh
- ovfhw2K = 0: No overflow

6.11 Overflow Flag Mask Register

The overflow mask register contains bits that can be set to 1 to mask the overflow signal OVF* in the event of overflows of particular types. Thus, when an overflow occurs of a type that corresponds to a given bit in the overflow flag mask register and that bit is set to 1 (masked), the corresponding flag in the overflow status register is set to 1 to indicate the overflow, but the overflow signal OVF* is not asserted and remains high.

Set all undefined bits to zero. At reset, all values become 00h.

OVFMASK — Register address 24h (all bits read/write enabled)
Overflow Flag Mask Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
movfdp	movfvp	movfhp	movfad	movfvw4K	movfvw2K	movfhw4K	movfhw2K

movfdp: Perimeter diagonal component 16-bit adder output overflow mask flag
- movfdp = 1: Masked
- movfdp = 0: Unmasked

movfvp: Perimeter vertical component 16-bit adder output overflow mask flag

movfvp = 1: Masked

movfvp = 0: Unmasked

movfhp: Perimeter horizontal component 16-bit adder output overflow mask flag

movfhp = 1: Masked

movfhp = 0: Unmasked

movfad: Adder output overflow mask flag

movfad = 1: Masked

movfad = 0: Unmasked

movfvw4K: Vcnt overflow mask flag for vertical processing screen size in 4K area mode

movfhw4K = 1: Masked

movfhw4K = 0: Unmasked

movfvw2K: Vcnt overflow mask flag for vertical processing screen size in 2K area mode

movfvw2K = 1: Masked

movfvw2K = 0: Unmasked

movfhw4K: Hcnt overflow mask flag for horizontal processing screen size in 4K area mode

movfhw4K = 1: Masked

movfhw4K = 0: Unmasked

movfhw2K: Hcnt overflow mask flag for horizontal processing screen size in 2K area mode

movfhw2K = 1: Masked

movfhw2K = 0: Unmasked

6.12 Results Memory Clear Register

The results memory clear register clears the contents of results memory. This differs from hardware resets initiated by the reset signal (RST*) or software resets from the reset register, in that no registers other than results memory are cleared.

Set all undefined bits to zero. At reset, all values become 00h.

CLRmem — Register address 28h (all bits read/write enabled)

Results Memory Clear Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
CLRexec	0	CLR5	CLR4	CLR3	CLR2	CLR1	CLR0

CLRexec: Results memory clear flag

This flag clears the contents of results memory. After CLRexec is set, results memory is cleared for the number of clock cycles specified in bits CLR0–CLR5. After results memory is cleared, CLRexec is automatically reset to 0. Clearing can be stopped while in process by writing 0 to the results memory clear flag, but this can affect the accuracy of the contents of results memory. The results memory clear flag differs from the soft reset flag rst in that it does not clear other registers.

CLRexec = 1: Clear results memory

CLRexec = 0: Cancel clearing of results memory, or end automatically

CLR0–CLR5: Results memory clear cycle selection flags

Clearing results memory takes one microsecond, so this register must designate the number of image clock cycles that elapse in a microsecond. After the designated number of clock cycles is counted (and thus a microsecond elapses), the results memory clear flag CLRexec is reset to 0 and the clearing process canceled. Note, however, that using a results memory clearing function with the value 0 in all bits CLR0–CLR5 represents a special case, and causes the clearing process to be executed for 64 clock cycles, just as if the register had been set to the decimal value 64.

For example, at 40 MHz, the Feature chip's maximum operating frequency, 40 clock cycles are required to provide the minimum of one microsecond needed to clear the register. Therefore, the bits CLR0–CLR5 would be set to the equivalent of the decimal value 40 (28h). Even with CLR0–CLR5 at their reset values of 00h, the memory clear time is 1.6 microseconds. Also, if the operating frequency is 12.5 MHz, memory clears in 5.12 microseconds at the default reset value of 00h. To set the minimum clearing time, write the equivalent of the decimal value 13 (0Dh) to bits CLR0–CLR5. In the absence of critical factors such as clock delays from line sensors or scanners, or stopping of clock counts, it is generally best to use the default setting of 00h.

6.13 Field Count Control Register

The field count control register establishes controls over execution based on the number of fields processed. If this register is used, the necessary settings must be entered before setting the exec bit.

After a reset, all values are 00h. Note that because labeled image data is the input for several processes (including circumscribed rectangle diagonal coordinate measurement, area boundary one-point coordinate measurement, primary moment measurement, secondary moment measurement, and perimeter length measurement), these processes deal with non-interlaced images: therefore two-field processing settings have no meaning.

FLD — Register address 2Ch (all bits read/write enabled)

Field Count Control Register — Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
fld7	fld6	fld5	fld4	fld3	fld2	fld1	fld0

fld0–fld7: Field count flag set

The field control register consists of flags that designate a number of fields that are counted using the processed field count control circuit. The MSB is fld7 and the LSB is fld0, and together the eight flags control the field count (from 01h to FFh). Note that if the fld flags are set to 00h, processed field count controls are not applied.

When any bit in the field count control register ($n \neq 0$) is set to 1 and the exec flag is set to 1 execute processing, the chip performs feature extraction processing for the duration of the designated number (n) of fields only, that is, for n periods in which the VEN* signal remains active (at low level). When these n fields are processed, the exec flag automatically clears to 0.

When 0 is written to the field count control register and then 1 is written to the exec flag to execute processing, processing field controls are not applied, and processing continues until 0 is written to the exec flag.

6.14 Constant Register

The constant register contains 48 bits and is used to represent the value of an external signal as a constant value. For example, using constants from the constant register to weight grayscale values for cumulative weighted grayscale measurement by area makes it possible to measure the number of pixels in each area. In this case, using the value 1 from the constant register produces normal histogram processing. In cumulative histogram processing, the fields near the end are sometimes given increased weightings. This technique can be used to create a time axis in a histogram of multiple fields, by adding a smaller weight from the constant register to the fields that are input first, then using increasingly larger weightings up to the last field to be input. This can also be used in measuring moving images to weight the most recent measurements.

The illustration below shows the address mapping of the constant register, in which the lowest 24 bits (C0–C23) are in address 30h–32h, and the highest 24 bits (C24–C47) are divided between address 34h–36h. This mapping arrangement allows for the writing of values from a 24-bit bus; however, in normal image measurement the higher 24 bits are seldom written, so there is no difficulty in using a 16-bit control bus.

CONST (all bits read/write enabled)

Constant Register (48-bits) Reset: 000000000000h

	MSB							LSB
address	37h	36h	35h	34h	33h	32h	31h	30h
	00	C47..40	C39..32	C31..24	00	C23..16	C15..8	C7..0

6.15 Origin Area Number Register

The origin area number register stores the number of the area that contains the origin (0, 0) located at the upper left corner of the image being measured. Use the area number assigned to the origin when extending processing areas for software processing, as well as for identifying the area assigned to the origins for area boundary one-point coordinate measurement. In area boundary one-point coordinate measurement, the one-point coordinate result in results memory can be 0 when the area containing the origin is measured. The origin area number register provides a means of distinguishing a processing result of 0 from a 0 remaining from the last memory clear cycle.

Values in this register are reset to 00h.

LBL00 (all bits read-only)

Origin Area Number Register Reset: 00h

address	39h				38h							
	MSB											LSB
	bit 11	bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
	lb11	lb10	lb9	lb8	lb7	lb6	lb5	lb4	lb3	lb2	lb1	lb0

lb (lb0–lb11): Origin area number value

This field holds the area number of the area containing the first pixel (0, 0) in the effective input image.

6.16 LSI Internal Test Registers

These registers are used to test the Feature chip before shipment, and should not be used for normal operation. Set all bits to 0, or leave them with 00h after a reset. If a 1 is written to any bits in this register, the chip enters test mode and may not function normally.

TESTM1 (all bits read/write enabled)
LSI Internal Test Register 1 Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
T7	T6	T5	T4	T3	T2	T1	T0

Register address: 3Ch

TESTM2 (all bits read/write enabled)
LSI Internal Test Register 2 Reset: 00h

MSB							LSB
bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
T7	T6	T5	T4	T3	T2	T1	T0

Register address: 3Dh

6.17 Results Memory Address Maps

In 2K or 4K area measurement mode, the configuration of results memory as seen from the CPU interface is 4096 words x 24-bits.

AD13–AD2	AD1–AD0 = 10	AD1–AD0 = 01	AD1–AD0 = 00
	b23 b16	b15 b8	b7 b0
00 0000 0000 00	00 0000 0000 00 10	00 0000 0000 00 01	00 0000 0000 00 00
00 0000 0000 01	00 0000 0000 01 10	00 0000 0000 01 01	00 0000 0000 01 00
00 0000 0000 10	00 0000 0000 10 10	00 0000 0000 10 01	00 0000 0000 10 00
00 0000 0000 11	00 0000 0000 11 10	00 0000 0000 11 01	00 0000 0000 11 00
00 0000 0001 00	00 0000 0001 00 10	00 0000 0001 00 01	00 0000 0001 00 00
. . .	. . .	. . .	. . .
11 1111 1111 10	11 1111 1111 10 10	11 1111 1111 10 01	11 1111 1111 10 00
11 1111 1111 11	11 1111 1111 11 10	11 1111 1111 11 01	11 1111 1111 11 00
	High 8 bits	Middle 8 bits	Low 8 bits

8-bit access (BUSW1 = 1, BUSW0 = 0)

Control address bus		Data bus		
AD1	AD0	DB23–DB16	DB15–DB8	DB7–DB0
0	0	High-Z (Note 1)	High-Z (Note 1)	Low 8 bit
0	1	High-Z (Note 1)	High-Z (Note 1)	Middle 8 bit
1	0	High-Z (Note 1)	High-Z (Note 1)	High 8 bit
1	1	High-Z (Note 1)	High-Z (Note 1)	00h constant

Note 1: DB23–DB8 should be pulled up or pulled down.

16-bit access (BUSW1 = 0, BUSW0 = 1)

Control address bus		Data bus		
AD1	AD0	DB23–DB16	DB15–DB8	DB7–DB0
0	*	High-Z (Note 2)	Middle 8 bit	Low 8 bit
1	*	High-Z (Note 2)	00h	High 8 bit

*: Ignored

Note 2: DB23–DB16 should be pulled up or pulled down.

24-bit access (BUSW1 = 0, BUSW0 = 0)

Control address bus		Data bus		
AD1	AD0	DB23–DB16	DB15–DB8	DB7–DB0
*	*	High 8-bit	Middle 8 bit	Low 8 bit

*: Ignored

6.18 Results Data Storage Formats

This section shows the formats used to store various types of image-processing results in results memory.

6.18.1 Circumscribed Rectangle Diagonal Coordinate Measurement Results

6.18.1.1 Results of Lower-Left, Upper-Right Simultaneous Coordinate Measurement (2K area measurement mode)

Word	WD_{n+1}			WD_n		
bit	47 – 36	35 – 24		23 – 12	11 – 0	
Measurement results	V1	H1		V0	H0	
Address	AD_{n+6}	AD_{n+5}	AD_{n+4}	AD_{n+2}	AD_{n+1}	AD_n

6.18.1.2 Results of Measurement of Lower-Left Coordinate Only (4K area measurement mode)

Word	WD_n		
bit	23 – 12	11 – 0	
Measurement results	V0	H0	
Address	AD_{n+2}	AD_{n+1}	AD_n

6.18.1.3 Results of Measurement of Upper-Right Coordinate Only (4K area measurement mode)

Word	WD_n		
bit	23 – 12	11 – 0	
Measurement results	V1	H1	
Address	AD_{n+2}	AD_{n+1}	AD_n

6.18.2 Area Boundary One-Point Coordinate Measurement Results

Word	WD_n		
bit	23 – 12	11 – 0	
Measurement results	Va	Ha	
Address	AD_{n+2}	AD_{n+1}	AD_n

6.18.3 Perimeter Length (Three-Direction Simultaneous) Measurement Results

Word	WD_{n+1}			WD_n		
	47 – 24			23 – 0		
bit	47 – 32	31 – 16		15 – 0		
Measurement results	Dperi	Vperi		Hperi		
Address	AD_{n+6}	AD_{n+5}	AD_{n+4}	AD_{n+2}	AD_{n+1}	AD_n

6.19 Detailed Sample Address Maps for Measurement Results Data

6.19.1 Circumscribed Rectangle Diagonal Coordinate Measurement Results

6.19.1.1 Results of Lower-Left, Upper-Right Simultaneous Coordinate Measurement (2K area measurement mode)

	AD1 to AD0				
	10	01		00	
AD13 to AD2	b23 - - - - - - b16	b15 - - - - - b12	b11 - - - - - - b8	b7 - - - - - - b0	Area number
00 0000 0000 00	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	000h
00 0000 0000 01	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	
00 0000 0000 10	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	001h
00 0000 0000 11	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	⋮
⋮	⋮	⋮	⋮	⋮	⋮
01 1111 1111 11	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	3FFh
10 0000 0000 00	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	400h
10 0000 0000 01	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	
⋮	⋮	⋮	⋮	⋮	⋮
11 1111 1111 10	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	7FFh
11 1111 1111 11	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	

6.19.1.2 Results of Measurement of Lower-Left Coordinate Only (4K area measurement mode)

	AD1 to AD0				
	10	01		00	
AD13 to AD2	b23 - - - - - - b16	b15 - - - - - b12	b11 - - - - - - b8	b7 - - - - - - b0	Area number
00 0000 0000 00	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	000h
00 0000 0000 01	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	001h
00 0000 0000 10	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	002h
00 0000 0000 11	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	003h
⋮	⋮	⋮	⋮	⋮	⋮
01 1111 1111 11	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	7FFh
10 0000 0000 00	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	800h
10 0000 0000 01	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	801h
⋮	⋮	⋮	⋮	⋮	⋮
11 1111 1111 10	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	FFEh
11 1111 1111 11	V0 high 8 bits	V0 low 4 bits	H0 high 4 bits	H0 low 8 bits	FFFh

6.19.1.3 Results of Measurement of Upper-Right Coordinate Only (4K area measurement mode)

	AD1 to AD0				
	10	01		00	
AD13 to AD2	b23 - - - - - - b16	b15 - - - - - b12	b11 - - - - - - b8	b7 - - - - - - b0	Area number
00 0000 0000 00	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	000h
00 0000 0000 01	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	001h
00 0000 0000 10	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	002h
00 0000 0000 11	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	003h
. . . .	. . . .	. . . .	. . . .	. . . .	. . . .
01 1111 1111 11	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	7FFh
10 0000 0000 00	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	800h
10 0000 0000 01	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	801h
. . . .	. . . .	. . . .	. . . .	. . . .	. . . .
11 1111 1111 10	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	FFEh
11 1111 1111 11	V1 high 8 bits	V1 low 4 bits	H1 high 4 bits	H1 low 8 bits	FFFh

6.19.2 Area Boundary One-Point Coordinate Measurement Results (4K area measurement mode)

	AD1 to AD0				
	10	01		00	
AD13 to AD2	b23 - - - - - - b16	b15 - - - - - b12	b11 - - - - - - b8	b7 - - - - - - b0	Area number
00 0000 0000 00	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	000h
00 0000 0000 01	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	001h
00 0000 0000 10	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	002h
00 0000 0000 11	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	003h
. . . .	. . . .	. . . .	. . . .	. . . .	. . . .
01 1111 1111 11	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	7FFh
10 0000 0000 00	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	800h
10 0000 0000 01	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	801h
. . . .	. . . .	. . . .	. . . .	. . . .	. . . .
11 1111 1111 10	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	FFEh
11 1111 1111 11	Va high 8 bits	Va low 4 bits	Ha high 4 bits	Ha low 8 bits	FFFh

6.19.3 Perimeter Length (Three-Direction Simultaneous) Measurement Results (2K area measurement mode)

AD13 to AD2	AD1 to AD0			
	10	01	00	
	b23 - - - - - - - - b16	b15 - - - - - - - - - b8	b7 - - - - - - - - - - b0	Area number
00 0000 0000 00	Vperi low 8 bits	Hperi high 8 bits	Hperi low 8 bits	000h
00 0000 0000 01	Dperi high 8 bits	Dperi low 8 bits	Vperi high 8 bits	
00 0000 0000 10	Vperi low 8 bits	Hperi high 8 bits	Hperi low 8 bits	001h
00 0000 0000 11	Dperi high 8 bits	Dperi low 8 bits	Vperi high 8 bits	
. . . .	. . . .	. . . .	. . . .	. . . .
01 1111 1111 11	Dperi high 8 bits	Dperi low 8 bits	Vperi high 8 bits	3FFh
10 0000 0000 00	Vperi low 8 bits	Hperi high 8 bits	Hperi low 8 bits	400h
10 0000 0000 01	Dperi high 8 bits	Dperi low 8 bits	Vperi high 8 bits	
. . . .	. . . .	. . . .	. . . .	. . . .
11 1111 1111 10	Vperi low 8 bits	Hperi high 8 bits	Hperi low 8 bits	7FFh
11 1111 1111 11	Dperi high 8 bits	Dperi low 8 bits	Vperi high 8 bits	

6.19.4 Grayscale Projection Processing Results

6.19.4.1 Two-Axis Simultaneous Projection Measurement Results (Normal measurement)

The address map below applies to results data from two-axis simultaneous projection processing when the disVcnt flag in the mode register is set to 0. The results of vertical projection processing (H-axis projection) are stored in addresses 0000h–1FFFh, and the results of horizontal projection processing (V-axis projection) are stored in addresses 2000h–3FFFh. When processing multiple fields using processing field count controls, each address stores the cumulative sum of grayscale values from the corresponding row/column of the number of fields processed.

	AD1 to AD0				
	10	01	00		
AD13 to AD2	b23 - - - - - - - - b16	b15 - - - - - - - - - b8	b7 - - - - - - - - - - b0	Coordinate value	
00 0000 0000 00				h = 000h	H-axis projection results
00 0000 0000 01				h = 001h	
00 0000 0000 10				h = 002h	
00 0000 0000 11				h = 003h	
⋮	⋮	⋮	⋮	⋮	
01 1111 1111 11				h = 7FFh	
10 0000 0000 00				v = 000h	V-axis projection results
10 0000 0000 01				v = 001h	
⋮	⋮	⋮	⋮	⋮	
11 1111 1111 10				v = 7FEh	
11 1111 1111 11				v = 7FFh	

6.19.4.2 Two-Axis Simultaneous Projection Measurement Results (Field-By-Field Measurement)

The address map shown below applies to results data from two-axis simultaneous projection processing when the disVcnt flag in the mode register is set to 1. The results of vertical projection processing (H-axis projection) are stored in addresses 0000h–1FFFh, and the results of horizontal projection processing (V-axis projection) are stored in addresses 2000h–3FFFh. When processing multiple fields using processing field count controls, the H-axis projection results contain the cumulative sum of grayscale values for each column of the number of fields processed. The V-axis projection results, however, are stored by individual field, as shown below. This is because the start points of the second and all subsequent fields are not reset by the vertical coordinate counter Vcnt when the disVcnt flag is set to 1.

	AD1 to AD0					
	10	01	00			
AD13 to AD2	b23 - - - - - - - - b16	b15 - - - - - - - - - b8	b7 - - - - - - - - - - b0	Coordinate value		
00 0000 0000 00				h = 000h	H-axis projection results	V-axis projection results
00 0000 0000 01				h = 001h		
00 0000 0000 10				h = 002h		
00 0000 0000 11				h = 003h		
⋮	⋮	⋮	⋮	⋮		
01 1111 1111 11				h = 7FFh		
10 0000 0000 00				v = 000h	1st field	
10 0000 0000 01				v = 001h		
⋮	⋮	⋮	⋮	⋮		
				v = xxxh		
				v = 000h	2nd field	
				v = 001h		
⋮	⋮	⋮	⋮	⋮		
				v = xxxh		
				v = 000h	3rd field	
				v = 001h		
⋮	⋮	⋮	⋮	⋮		
11 1111 1111 11						

6.19.4.3 Vertical One-Axis Projection Processing Results (4K area measurement mode)

AD13 to AD2	AD1 to AD0			Area number
	10	01	00	
	b23 - - - - - - - - b16	b15 - - - - - - - - b8	b7 - - - - - - - - - - b0	
00 0000 0000 00				h = 000h
00 0000 0000 01				h = 001h
00 0000 0000 10				h = 002h
00 0000 0000 11				h = 003h
⋮	⋮	⋮	⋮	⋮
01 1111 1111 11				h = 7FFh
10 0000 0000 00				h = 800h
10 0000 0000 01				h = 801h
⋮	⋮	⋮	⋮	⋮
11 1111 1111 10				h = FFEh
11 1111 1111 11				h = FFFh

6.19.4.4 Horizontal One-Axis Projection Processing Results (Normal measurement, 4K area measurement mode)

The address map below applies to results data from one-axis projection processing in the horizontal direction only (V-axis projection only) when the disVcnt flag in the mode register is set to 0. When processing multiple fields using processing field count controls, each address contains the cumulative sum of grayscale values from the corresponding row of the number of fields processed. This is because the start points of the second and all subsequent fields are reset by the vertical coordinate counter Vcnt when the disVcnt flag is set to 0.

AD13 to AD2	AD1 to AD0			Area number
	10	01	00	
	b23 - - - - - - - - b16	b15 - - - - - - - - b8	b7 - - - - - - - - - - b0	
00 0000 0000 00				v = 000h
00 0000 0000 01				v = 001h
00 0000 0000 10				v = 002h
00 0000 0000 11				v = 003h
⋮	⋮	⋮	⋮	⋮
01 1111 1111 11				v = 7FFh
10 0000 0000 00				v = 800h
10 0000 0000 01				v = 801h
⋮	⋮	⋮	⋮	⋮
11 1111 1111 10				v = FFEh
11 1111 1111 11				v = FFFh

6.19.4.5 Horizontal One-Axis Projection Processing Results (Field-By-Field Measurement, 4K Area Measurement Mode)

The address map below applies to results data from one-axis simultaneous projection processing in the horizontal direction only (V-axis projection) when the disVcnt flag in the mode register is set to 1. When processing multiple fields using processing field count controls, the results are stored by field as shown below. This is because the start points of the second and all subsequent fields are not reset by the vertical coordinate counter Vcnt when the disVcnt flag is set to 1.

	AD1 to AD0				
	10	01	00		
AD13 to AD2	b23 - - - - - - - - b16	b15 - - - - - - - - - b8	b7 - - - - - - - - - - b0	Coordinate value	
00 0000 0000 00				v = 000h	1st field
00 0000 0000 01				v = 001h	
⋮	⋮	⋮	⋮	⋮	
				v = xxxh	
				v = 000h	2nd field
				v = 001h	
⋮	⋮	⋮	⋮	⋮	
				v = xxxh	
				v = 000h	3rd field
				v = 001h	
⋮	⋮	⋮	⋮	⋮	
				v = xxxh	
⋮	⋮	⋮	⋮	⋮	⋮
11 1111 1111 11					

6.19.5 Measurement Results Using the 24/48-bit Adder

Measurement processes using the 24/48-bit adder (in which output from the 24/48-bit adder is designated as data input to results memory) include cumulative grayscale value processing by area, histogram processing, primary moment measurement, secondary moment measurement, perimeter measurement in one-direction mode, and grayscale projection processing. The address maps for measurement data from these processes are shown below, for 4K area measurement mode and 2K area measurement mode respectively. (For grayscale projection processing, see also Section 6.19.4, "Grayscale Projection Processing Results.")

6.19.5.1 4K Area Measurement Mode

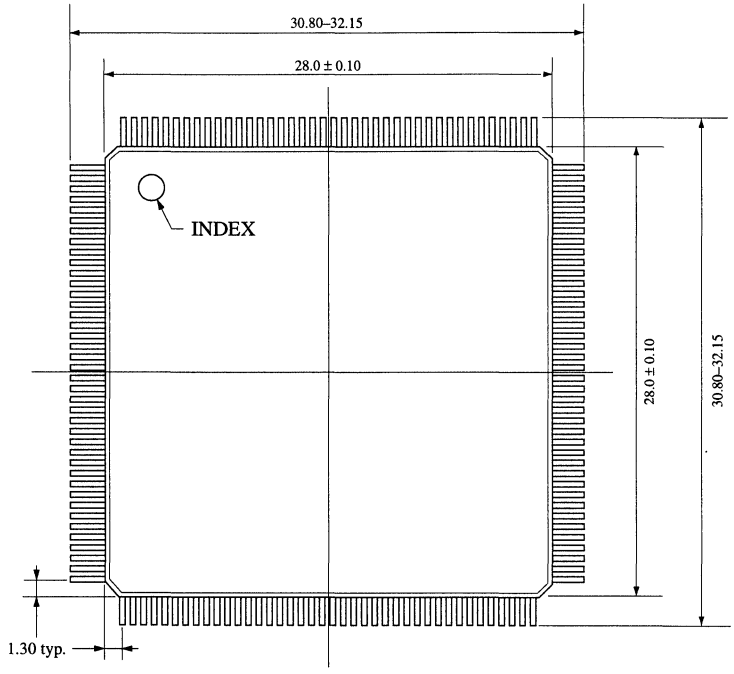
AD13 to AD2	AD1 to AD0 = 10 b23 - - - - - - - - b16	AD1 to AD0 = 01 b15 - - - - - - - - - b8	AD1 to AD0 = 00 b7 - - - - - - - - - - b0	Area number
00 0000 0000 00				000h
00 0000 0000 01				001h
00 0000 0000 10				002h
00 0000 0000 11				003h
⋮	⋮	⋮	⋮	⋮
01 1111 1111 11				7FFh
10 0000 0000 00				800h
10 0000 0000 01				801h
⋮	⋮	⋮	⋮	⋮
11 1111 1111 10				FFEh
11 1111 1111 11				FFFh

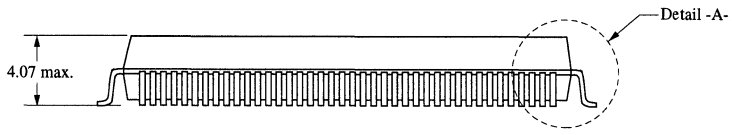
6.19.5.2 2K Area Measurement Mode

AD13 to AD2	AD1 to AD0 = 10 b23 - - - - - - - - b16	AD1 to AD0 = 01 b15 - - - - - - - - - b8	AD1 to AD0 = 00 b7 - - - - - - - - - - b0	Area number
00 0000 0000 00		Low 24-bits		000h
00 0000 0000 01		High 24-bits		
00 0000 0000 10		Low 24-bits		001h
00 0000 0000 11		High 24-bits		
⋮	⋮	⋮	⋮	
01 1111 1111 11		High 24-bits		3FFh
10 0000 0000 00		Low 24-bits		400h
10 0000 0000 01		High 24-bits		
⋮	⋮	⋮	⋮	
11 1111 1111 10		Low 24-bits		7FFh
11 1111 1111 11		High 24-bits		

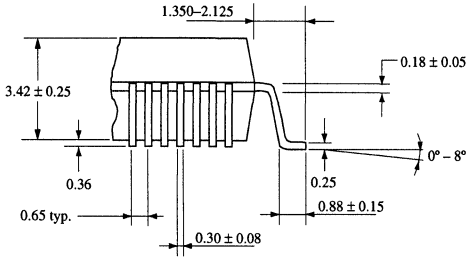
Section 7: External Dimensions

7.1 160-pin Plastic QFP Package





Units: mm



Detail -A-

Section 8: Electrical Characteristics

8.1 Absolute Maximum Ratings

Parameter	Symbol	Rating	Unit
Supply voltage	Vdd	-0.3 to 6.0	V
Input voltage	Vi	-0.3 to Vdd + 0.3	V
Input current	Ii	±10	mA
Output current	Io	10	mA
Operating temperature	Topt	0 to 70	°C
Storage temperature	Tstg	-10 to 80	°C

8.2 Recommended Operating Conditions

(GND = 0 V, Ta = 0 to 70 °C)

Parameter	Symbol	Condition		Min.	Typ.	Max.	Unit
Power-supply voltage	Vdd			4.75	5.0	5.25	V
High-level input voltage	Vih	TTL level, normal input		2.0	—	Vdd	V
Low-level input voltage	Vil			0	—	0.8	V
High-level threshold voltage	Vih+	Schmitt input (Note 1)	Vdd = 4.75 V	2.5		3.2	V
			Vdd = 5V, Ta = 25 °C	—	3.1	—	V
			Vdd = 5.25 V	3.0	—	3.7	V
Low-level threshold voltage	Vil-		Vdd = 4.75 V	1.2	—	1.9	V
			Vdd = 5V, Ta = 25 °C	—	1.7	—	V
			Vdd = 5.25 V	1.5	—	2.3	V
Input rise time	Tri	TTL level, normal input		0	—	100	ns
Input fall time	Tfi			0	—	100	ns
Input rise time	Tris	Schmitt input (Note 1)		0	—	1000	ns
Input fall time	Tfis			0	—	1000	ns

Note 1: The Schmitt input pins are the RST* and TEST* pins.

8.3 Input/Output Pin Capacitance

(Vdd = Vi = 0 V)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Input pins	Cin	f = 1 MHz	—	10	—	pF
Output pins	Cout	f = 1 MHz	—	10	—	pF
I/O pins	Cin	f = 1 MHz	—	10	—	pF

8.4 DC Characteristics

Vdd = 5 V ± 5%
Ta = 0 to 70 °C
GND = 0V

Parameter		Symbol	Condition	Min.	Typ.	Max.	Unit
Static power consumption (Note 1)		Il	Vi = Vdd or GND	—	—	200	µA
Output short circuit current (all output and I/O pins) (Note 2)		Ios	Vdd = Max, Vo = Vdd	15	50	200	mA
			Vdd = Max, Vo = 0 V	-5	-25	-100	mA
Low level input leak current							
	Normal I/O pins	Iil	Vi = GND	-10	±1	10	µA
	Pins with pull-up resistance (Note 3)	Iipl	Vi = GND	-140	—	-410	µA
High-level input leak current (all input and I/O pins)		Iih	Vi = Vdd	-10	±1	10	µA
Low-level output voltage		Vol	Iol = 8 mA	—	—	0.4	V
High-level output voltage		Voh	Ioh = 8 mA	2.4	—	—	V
Schmitt hysteresis voltage (Note 4)		Vsch	Vdd = 4.75 V	0.9	—	—	V
			Vdd = 5 V, Ta = 25°C	—	1.4	—	V
			Vdd = 5.25 V	1.5	—	—	V

Note 1: Excluding static current dissipation to pull-up resistors.

Note 2: Output short current is for one second or less, applied only to one LSI pin at a time.

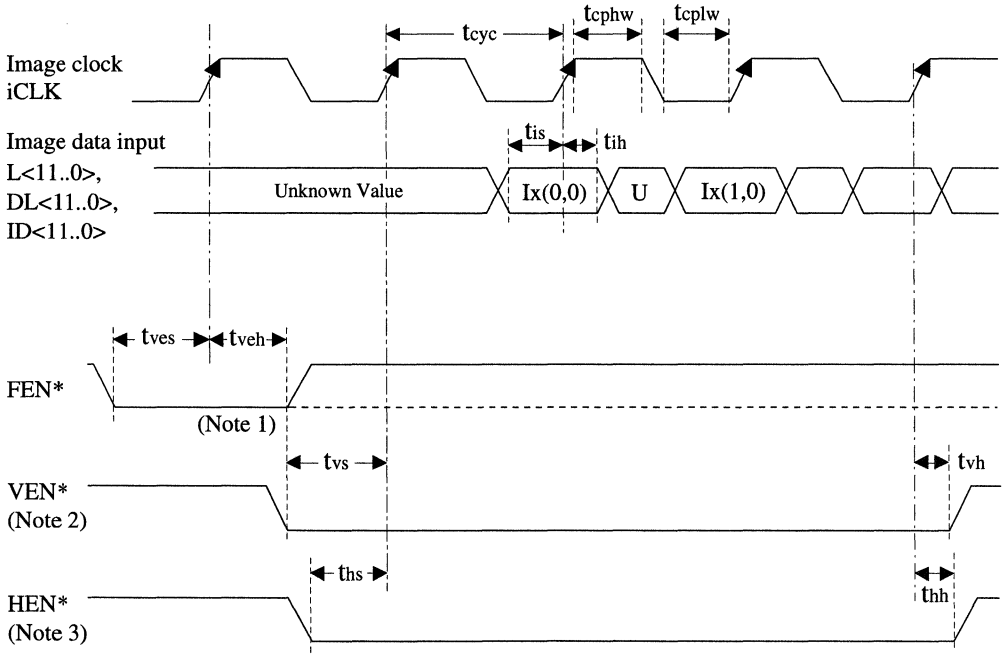
Note 3: Pins with pull-up resistance are the RST*, TEST*, and DL<11..0> pins.

Note 4: The Schmitt input pins are the RST* and TEST* pins.

8.5 AC Characteristics

All specifications in this section assume a load capacity of 30 pF on all output pins.

a) Data timing at frame start point



Note 1: The start of a field is recognized when the change (assertion) of the VEN* signal from high to low is detected at the rise of iCLK. Each field defined by a VEN* signal can be enabled or disabled by the FEN* signal. If FEN* is low one clock cycle before the assertion of the VEN* signal is detected, the field defined during that assertion of VEN* is enabled for execution and processing. If, on the other hand, FEN* is high one clock cycle before the assertion of VEN* is detected, that VEN* assertion is disabled, and execution must wait for the start of the next enabled field.

During a disabled VEN* assertion cycle, FEN* can be high or low, but will not be referred to again until immediately before the entry of the next field. In this sense, the field enable/disable decision is not made while data for the field is being input.

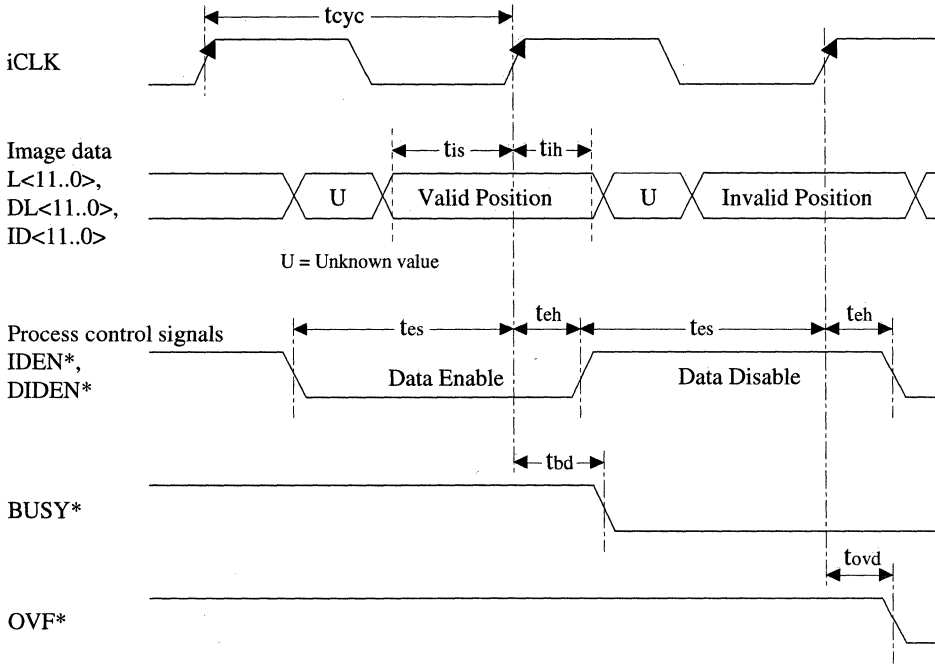
Note 2: VEN* must remain high for at least five clock cycles before the next synchronous input.

Note 3: HEN* must remain high for at least two clock cycles before the next synchronous input.

units: ns

Item	Symbol	IP90C18-HS			IP90C18		
		Min.	Typ.	Max.	Min.	Typ.	Max.
iCLK signal cycle time	t_{cyc}	25.0	—	—	50.0	—	—
iCLK signal high time	t_{cphw}	9.0	—	—	20.0	—	—
iCLK signal low time	t_{cplw}	9.0	—	—	20.0	—	—
Image data L<11..0>, DL<11..0>, ID<11..0> setup time	t_{is}	8.0	—	—	10.0	—	—
Image data L<11..0>, DL<11..0>, ID<11..0> hold time	t_{ih}	3.0	—	—	3.0	—	—
FEN* signal setup time	t_{ves}	8.0	—	—	10.0	—	—
FEN* signal hold time	t_{veh}	3.0	—	—	3.0	—	—
VEN* signal setup time	t_{vs}	8.0	—	—	10.0	—	—
VEN* signal hold time	t_{vh}	3.0	—	—	3.0	—	—
HEN* signal setup time	t_{hs}	8.0	—	—	10.0	—	—
HEN* signal hold time	t_{hh}	3.0	—	—	3.0	—	—

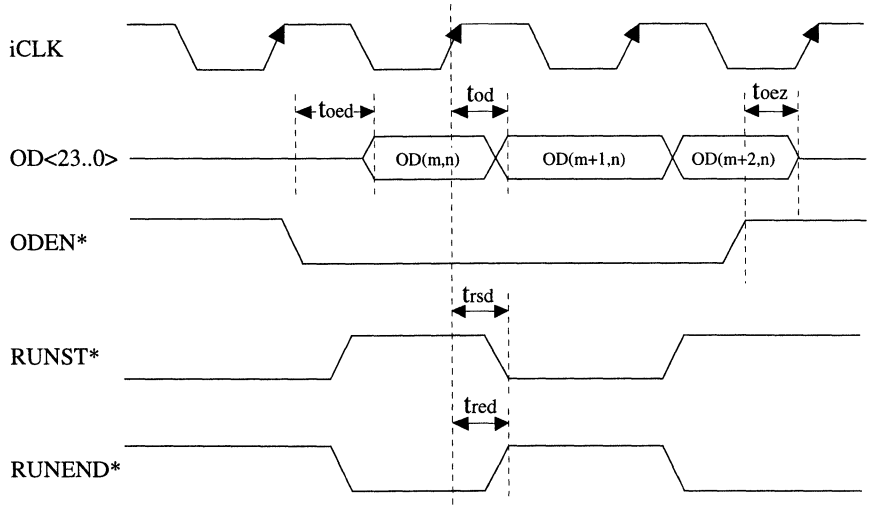
b) Image data input control system timing



units: ns

Item	Symbol	IP90C18-HS			IP90C18		
		Min.	Typ.	Max.	Min.	Typ.	Max.
iCLK signal cycle time	t_{cyc}	25.0	—	—	50.0	—	—
Image data L<11..0>, DL<11..0>, ID<11..0> setup time	t_{is}	8.0	—	—	10.0	—	—
Image data L<11..0>, DL<11..0>, ID<11..0> hold time	t_{ih}	3.0	—	—	3.0	—	—
IDEN*, DIDEN* signal setup time	t_{es}	8.0	—	—	10.0	—	—
IDEN*, DIDEN* signal hold time	t_{eh}	3.0	—	—	3.0	—	—
BUSY* signal delay time	t_{bd}	3.0	—	20.0	3.0	—	25.0
OVF* signal delay time	t_{ovd}	3.0	—	20.0	3.0	—	25.0

c) Measurement results output control system timing

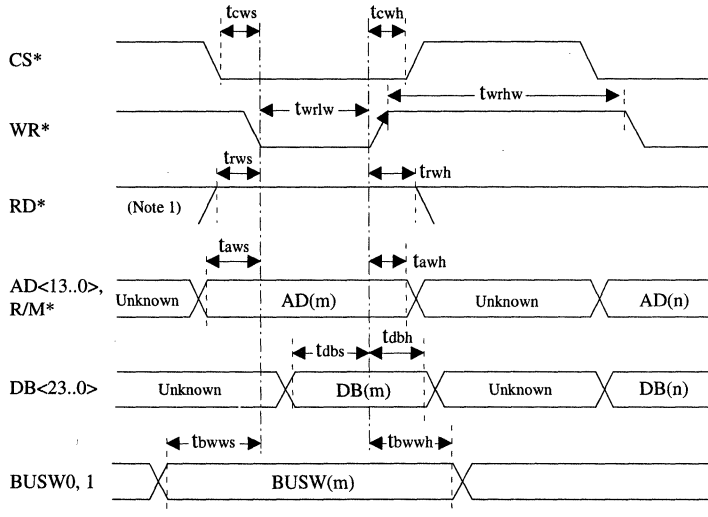


units: ns

Item	Symbol	IP90C18-HS			IP90C18		
		Min.	Typ.	Max.	Min.	Typ.	Max.
Delay from ODEN* signal fall to image data output OD<23..0> enable	t_{oed}	—	—	15.0	—	—	20.0
Delay from ODEN* signal rise to image data output OD<23..0> disable	t_{oez}	—	—	15.0	—	—	20.0
OD<23..0> delay	t_{od}	3.0	—	20.0	3.0	—	25.0
RUNST* signal delay	t_{rsd}	3.0	—	20.0	3.0	—	25.0
RUNEND* signal delay	t_{red}	3.0	—	20.0	3.0	—	25.0

d) CPU interface system timing

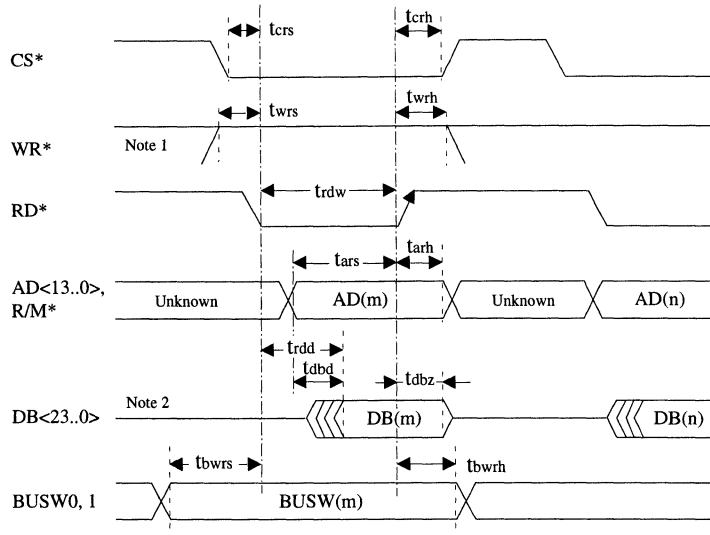
Write Cycle



units: ns

Item	Symbol	Min.	Typ.	Max.
CS* signal setup time (from WR* fall)	t_{cws}	20.0	—	—
CS* signal hold time (from WR* rise)	t_{cwh}	3.0	—	—
RD* signal setup time (from WR* fall)	t_{rws}	3.0	—	—
RD* signal hold time (from WR* rise)	t_{rwh}	3.0	—	—
WR* signal low pulse width	t_{wrh}	20.0	—	—
WR* signal high pulse width	t_{wrw}	15.0	—	—
AD<13..0>, R/M* setup time (from WR* fall)	t_{aws}	20.0	—	—
AD<13..0>, R/M* hold time (from WR* rise)	t_{awh}	3.0	—	—
DB<23..0> setup time (from WR* rise)	t_{dbs}	24.0	—	—
DB<23..0> hold time (from WR* rise)	t_{dbh}	3.0	—	—
BUSW0, 1 setup time (from WR* fall)	t_{bwws}	50.0	—	—
BUSW0, 1 hold time (from WR* rise)	t_{bwwh}	20.0	—	—

Read Cycle



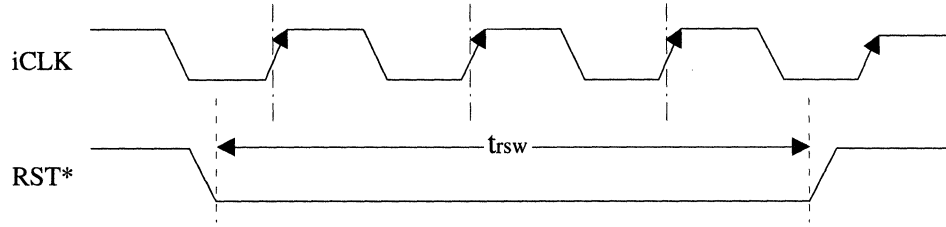
Note 1: While the WR* signal is low, the RD* signal should be high. Make sure both signals are never low at the same time. Similarly, when the RD* signal is low, the WR* signal should be high.

Note 2: The above illustrations are for 24-bit access (BUSW0 = BUSW1 = 0). For 16-bit access (BUSW0 = 1, BUSW1 = 0), signal pins DB<15..0> are in output status, and BS<23..16> are in high impedance status and should be internally pulled up or pulled down. For 8-bit access (BUSW0 = 0, BUSW1 = 1), signal pins DB<7..0> are in output status, and DB<23..8> are in high impedance status and should be internally pulled up or pulled down.

units: ns

Item	Symbol	Min.	Typ.	Max.
CS* signal setup time (from RD* fall)	t_{crs}	20.0	—	—
CS* signal hold time (from RD* rise)	t_{crh}	3.0	—	—
WR* signal setup time (from RD* fall)	t_{wrs}	3.0	—	—
WR* signal hold time (from RD* rise)	t_{wrh}	3.0	—	—
RD* signal low pulse width	t_{rdw}	25.0	—	—
AD<13..0>, R/M* setup time (from RD* rise)	t_{ars}	20.0	—	—
AD<13..0>, R/M* hold time (from RD* rise)	t_{arh}	3.0	—	—
Delay until DB<23..0> enabled (from AD<13..0>, R/M* signal)	t_{dbd}	—	—	45.0
Delay until DB<23..0> enabled (from RD* fall)	t_{rdd}	—	—	16.0
Delay until DB<23..0> high-Z state (from RD* rise)	t_{dbz}	—	—	16.0
BUSW0, 1 setup time (from RD* fall)	t_{bwrs}	50.0	—	—
BUSW0, 1 hold time (from RD* rise)	t_{bwrh}	20.0	—	—

e) Reset system timing



Note: The reset signal input cycle must be at least three clock cycles long. The internal circuits are reset when a low-level RST* signal is detected at the rising edge of two successive clock cycles.

The reset condition is canceled three clock cycles after the rise of RST* is detected at the rise of iCLK.

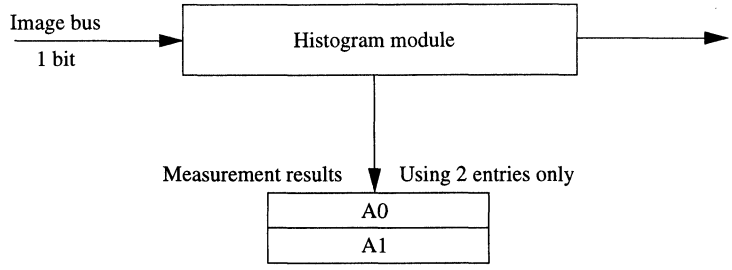
units: clock cycles

Item	Symbol	Min.	Typ.	Max.
RST* signal low pulse width	t_{rsw}	3.0	—	—

Note: The system reset function is enabled when RST* is low (asserted) for two clock cycles. The reset condition is released three clock cycles after the rise of RST* is detected at the rise of iCLK.

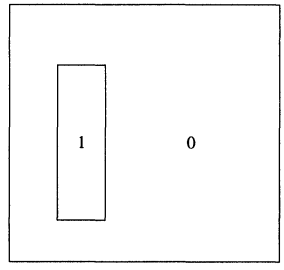
Section 9: Sample Applications and Reference Materials

9.1 Detecting Defective or Missing Objects Using Binary Image Area Measurement

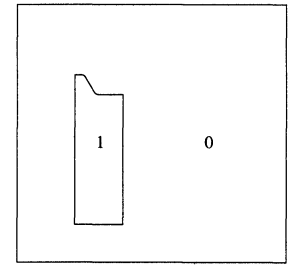


Sample Applications

Defects in shape, surface flaws, and missing items or pieces can be detected by using surface area measurement and comparing the results with a standard.



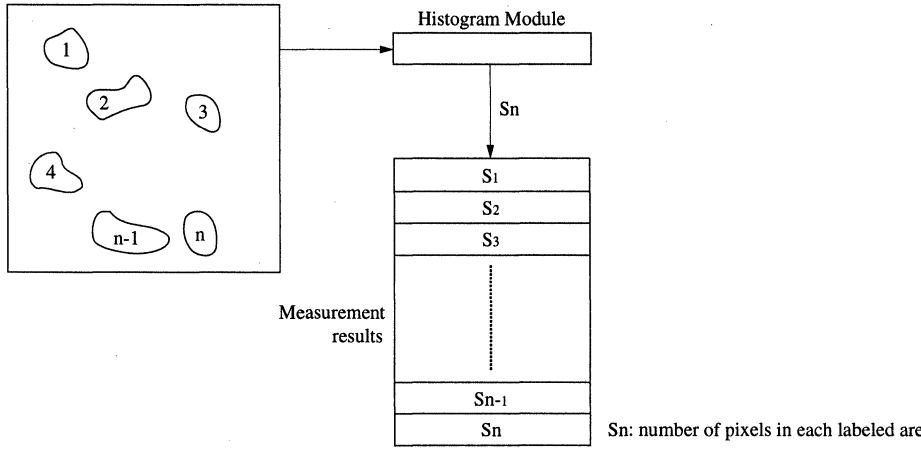
(Acceptable item)



(Defective or missing item)
1 count: n-m
The 1 count is decreased by the surface area of the defective or missing piece.

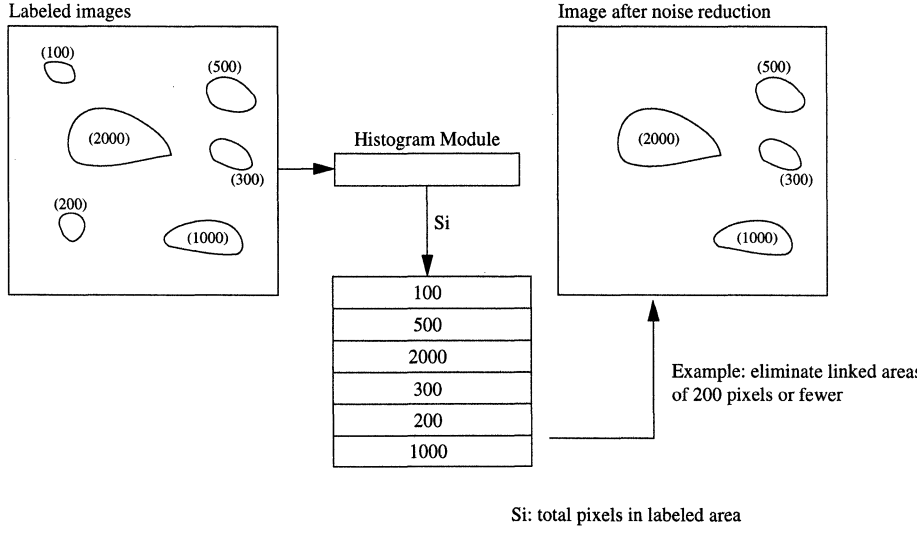
9.2 Surface Area Measurement of Labeled Images

The surface areas of multiple labeled images can be obtained by using histogram processing to find how often each label value occurs in the whole image (by performing a pixel count for each area).



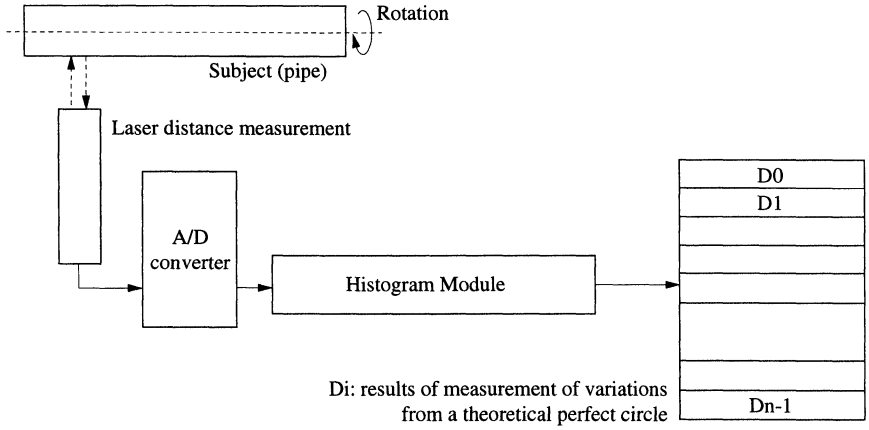
Sample Applications

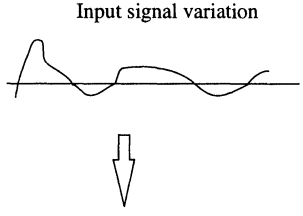
This procedure can be used for analyzing metal composition or particles in liquid solution, counting spherical objects, or eliminating noise by using a threshold area value, as shown below.



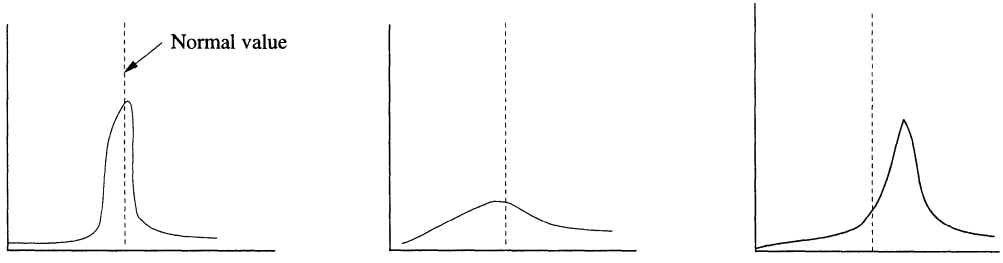
9.3 Data Analysis by Histogram (Histogram Measurement of Non-Image Data)

Histogram measurement is a general statistical calculation process that has many applications other than image processing. In the example below, laser distance measurement is used to examine the external diameter of a cylindrical object (such as a pipe). The resulting data passes through an A/D converter, and is input in digital form to a histogram measurement system that produces a statistical analysis of variability in the pipe diameter.

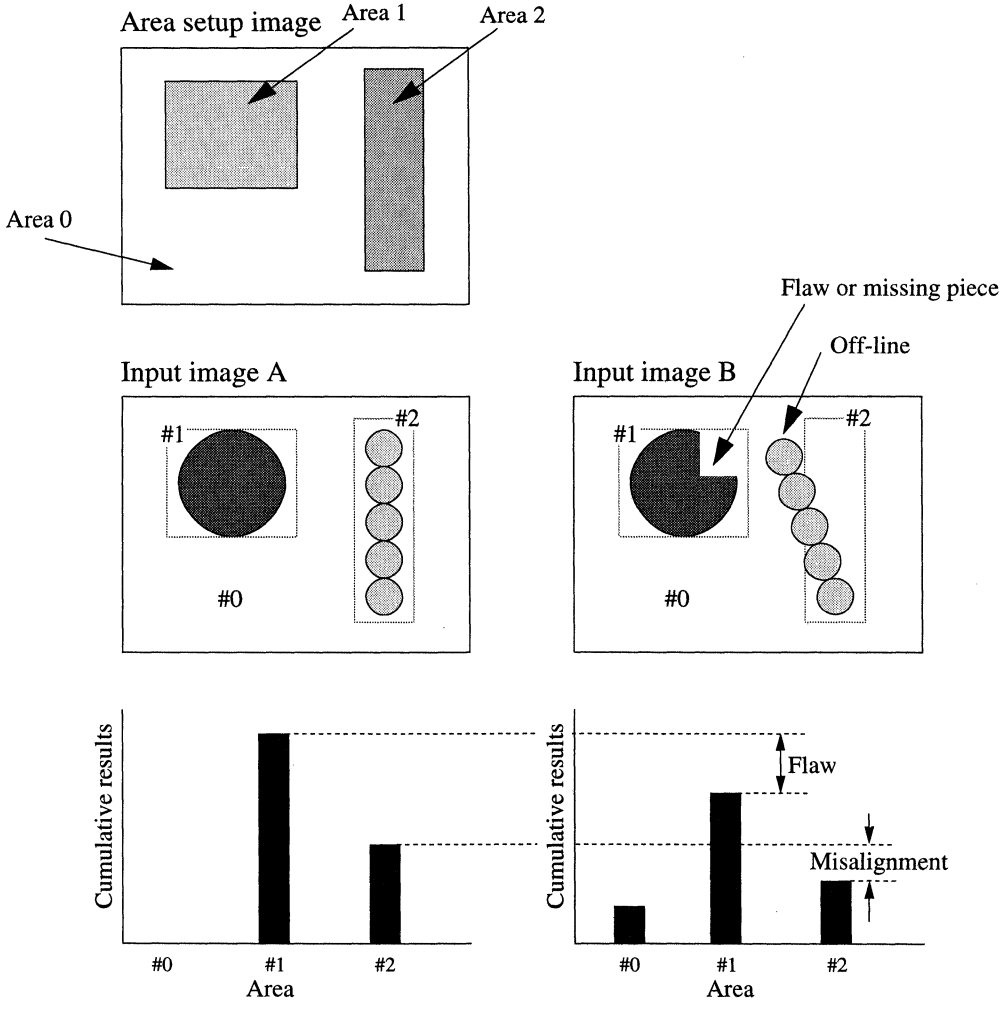




*Measurement results: frequency of occurence display



9.4 Detecting Missing or Misaligned Objects by Cumulative Addition Processing of Separate Areas



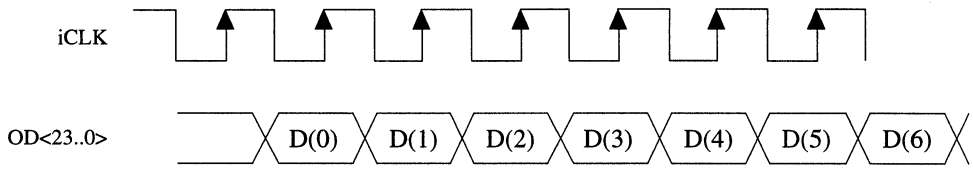
Cumulative results for input image A
PASS

Cumulative results for input image B
FAIL

9.5 Dumping the Contents of Results Memory

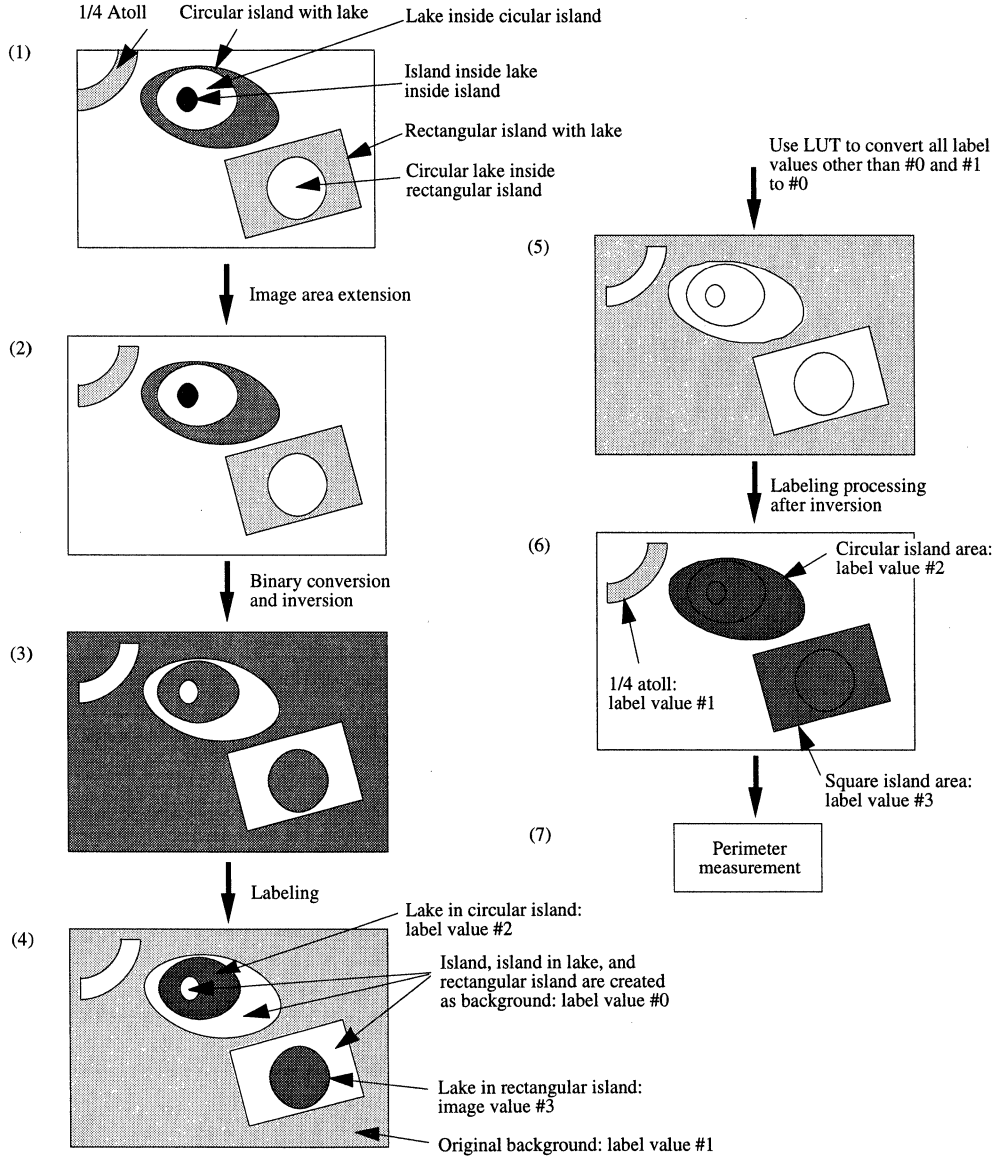
After processing is complete, the values stored in results memory can be output over the image data output pins (OD<23..0>) by selecting the value of the horizontal coordinate counter as the results memory address input value. For specific examples of register settings, see Section 5.3.14, "Results Memory Dump." For timing charts, see Section 4.4, "Operating Timing Charts."

Memory address	Measurement results
0	D(0)
1	D(1)
2	D(2)
3	D(3)
4	D(4)
	⋮
i	D(i)
	⋮
4095	D(4095)



9.6 Perimeter Measurements

Perimeter measurements are executed after the enclosed area is filled in. In Figure (1) below, the white areas are treated as background



9.7 Measuring Roundness

The characteristics of the external shape of each area in an image can be discovered by making individual surface area and perimeter measurements. When an object's surface area remains constant but its perimeter increases, the object can be considered to have increasing degrees of deformation. Conversely, among objects with the same area, the one with the smallest perimeter is closest to a perfect circle, so that the length of an area's perimeter can be considered inversely proportional to the area's roundness.

In the example below, areas 5 and 9 have the same surface area, but area 9 has a longer perimeter, and area 9 therefore has a greater degree of deformation. When measuring roundness, the higher the degree of roundness, the less the area's shape differs from a perfect circle (which has a roundness of 1). In digital images, however, the use of discrete values introduces an element of error. This example produces a value greater than the theoretical limit of 1 because the area of measurement is very small, which increases the magnitude of the contradiction between definitions of perimeter length and area: perimeter length is measured by considering each pixel as a point without size, while area is measured by considering each pixel as a unit of area.

Example: Screen with input area numbers (labeled image):

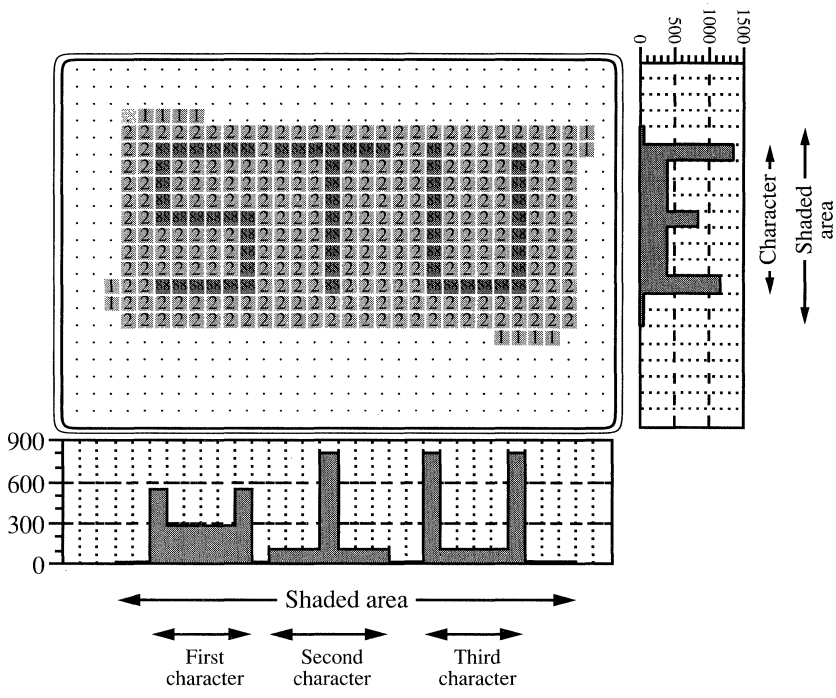
0	0	0	**5**	0	0	0	0	0	0
0	0	**5**	**5**	**5**	0	0	**9**	0	0
0	0	0	**5**	0	0	0	**9**	0	0
0	0	0	0	0	0	**9**	0	0	0
0	0	0	0	0	0	**9**	**9**	0	0
0	0	0	0	0	0	0	0	0	0

Area no.	Measurement results				Calculation results	
	Pixel count (area)	Perimeter			Perimeter	Roundness
		Vertical	Horizontal	Diagonal	V + H + D÷2	$4\pi S/L^2$
5	5	0	0	4	4÷2 ≈ 5.7	1.96
9	5	4	1	3	5 + 3÷2 ≈ 9.2	0.74

9.8 Grayscale Projection Processing

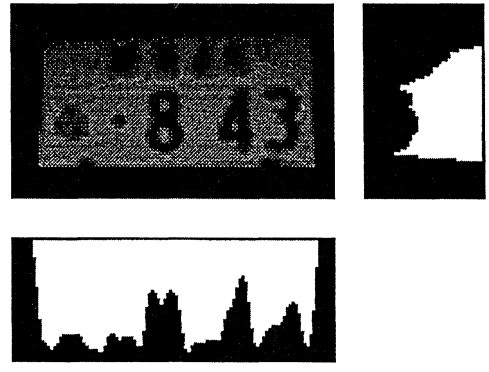
9.8.1 Character Recognition

One use of projection processing is recognizing simple character areas and symbols.



9.8.2 License Plate Number Recognition

The example below shows the use of projection processing on an automobile license plate.



9.9 Compensation of Moment Characteristics for Screen Partition or Origin Movement

The Feature chip can process measurements of moment characteristics on large input screens. In cases where the screen is partitioned into two or more smaller screens, or when the origin moves, it is still practical to use the chip's high-speed processing capability to make repeat measurements, though the movement of coordinate points can also be compensated through simple calculations.

Consider using the following compensation techniques when using parallel processing with multiple chips for large-area images obtained from scanners or line sensors, or when handling images that have already been partitioned during labeling or other processes. Although the examples shown here are taken from studies by Sumitomo, keep in mind that Sumitomo cannot assume responsibility for the consequences of any mistakes in the formulas presented. Always thoroughly verify the accuracy of all compensating formulas before using them.

9.9.1 Zero-Degree Moment Calculation (Area)

The zero-degree moment expresses surface area. Because this value is not related to image coordinates, no compensation is required when the origin is moved.

In a partitioned screen, some object areas may extend into two or more partitioned sections. It is then necessary to deduce which parts of these object areas belong to the same object, by using the values of pixels on the partition boundaries of the labeled image. This requires a process that searches for and matches portions of areas that straddle the partitions in the image.

This procedure is also applied in grayscale-weighted area measurement, so that area measurement results can be totaled and weighted for objects that extend into two or more areas in partitioned screens.

9.9.2 Primary Moment Calculation

If the calculation of primary moment characteristics about the origin (0, 0) is represented by m_{10}, m_{01}, then from the definitions of zero-degree moment and primary moment, the primary moment m'_{10}, m'_{01} about the origin after it has moved some distance (Dx, Dy) can be determined by equations (9-1) and (9-2):

$$m'_{10} = m_{10} - \Delta x m_{00} \qquad (9\text{-}1)$$

$$m'_{01} = m_{01} - \Delta y m_{00} \qquad (9\text{-}2)$$

This shows that for each object area contained on the screen, compensation for the primary moment after movement of the origin can be derived by simple calculations from the zero-degree moment and the primary moment before moving the origin, plus the coordinates of the new origin. Object areas that straddle two or more boundaries in a partitioned screen require a process that searches for and matches portions of areas, as is done in zero-degree moment calculations.

9.9.3 Secondary Moment Calculation

If the calculation of secondary moment characteristics about the origin (0, 0) is represented by (m_{20}, m_{11}, m_{02}), then from the definitions of zero-degree moment, primary moment and secondary moment, the secondary moment (m'_{20}, m'_{11}, m'_{02}) about the origin after it has moved the distance (Dx, Dy) can be determined by equations (9-3), (9-4), and (9-5):

$$m'_{20} = m_{20} - 2 \cdot \Delta x \cdot m_{10} + \Delta x^2 \cdot m_{00} \qquad (9\text{-}3)$$

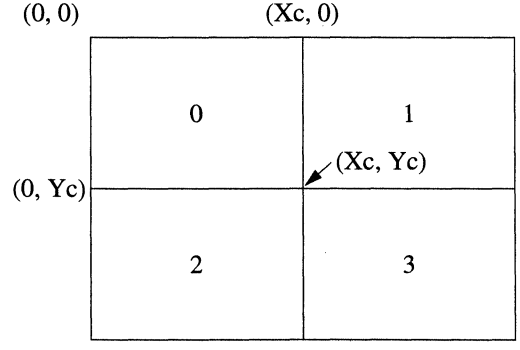
$$m'_{02} = m_{02} - 2 \cdot \Delta y \cdot m_{01} + \Delta y^2 \cdot m_{00} \qquad (9\text{-}4)$$

$$m'_{11} = m_{11} - \Delta y \cdot m_{10} - \Delta x \cdot m_{01} + \Delta x \cdot \Delta y \cdot m_{00} \qquad (9\text{-}5)$$

These equations show that for each object area contained on the screen, compensation for the secondary moment after movement of the origin can be derived by simple calculations from the zero-degree moment, primary moment, and secondary moment before moving the origin, plus the coordinates of the new origin. Object areas that straddle two or more boundaries in a partitioned screen require a process that searches for and matches portions of areas, as described for zero-degree and primary moment calculations.

9.9.4 Partitioning the Screen into Four Areas

The following procedure measures moment characteristics with the screen partitioned into four areas.



In this example, consider a screen partitioned into four areas around a central point (Xc, Yc) for calculating moment characteristics in which the upper left corner of each of the four screen areas is used as the origin (0, 0) of that area. As shown in the illustration, the four areas are labeled 0, 1, 2, and 3. Moment characteristics measured in each of these areas is represented by $m(0)_{xx}$, $m(1)_{xx}$, $m(2)_{xx}$, and $m(3)_{xx}$. Thus:

- $m(0)_{xx}$ represents moment characteristics measured with respect to the original origin.
- $m(1)_{xx}$ represents moment characteristics measured using (Xc, 0) as the origin in area 1.
- $m(2)_{xx}$ represents moment characteristics using (0, Yc) as the origin in area 2.
- $m(3)_{xx}$ represents moment characteristics using (Xc, Yc) as the origin in area 3.

In this case, moment characteristics with respect to the full screen can be expressed using the following equations (9-6) through (9-11):

$$m_{00} = m(0)_{00} + m(1)_{00} + m(2)_{00} + m(3)_{00} \quad (9\text{-}6)$$

$$m_{10} = m(0)_{10} + m(1)_{10} - Xc \cdot m(1)_{00} + m(2)_{10} + m(3)_{10} - Xc \cdot m(3)_{00} \quad (9\text{-}7)$$

$$m_{01} = m(0)_{01} + m(1)_{01} + m(2)_{01} - Yc \cdot m(2)_{00} + m(3)_{10} - Yc \cdot m(3)_{00} \quad (9\text{-}8)$$

$$m_{20} = m(0)_{20} + m(1)_{20} + m(2)_{20} - 2Xc \cdot m(1)_{10} + Xc^2 \cdot m(1)_{00} + m(3)_{20} - 2Xc \cdot m(3)_{10} + Xc^2 \cdot m(3)_{00} \quad (9\text{-}9)$$

$$m_{02} = m(0)_{02} + m(1)_{02} + m(2)_{02} - 2Yc \cdot m(2)_{01} + Yc^2 \cdot m(2)_{00} + m(3)_{02} - 2Yc \cdot m(3)_{01} + Yc^2 \cdot m(3)_{00} \quad (9\text{-}10)$$

$$m_{11} = m(0)_{11} + m(1)_{11} - Xc \cdot m(1)_{01} + m(2)_{11} - Yc \cdot m(2)_{10} + m(3)_{11} - Yc \cdot m(3)_{10} - Xc \cdot m(3)_{10} + Xc \cdot Yc \cdot m(3)_{00} \quad (9\text{-}11)$$

9.10 Maximum Values For Which Moment Characteristics Can Be Derived

The process of measuring moment characteristics requires the Feature chip to store intermediate results and final measurement results in its internal results memory.

Zero-degree moment characteristics (surface area) are measured in 48-bit width (4096 words). Grayscale-weighted area, as well as primary and secondary moment characteristics are measured in either 24-bit width (2048 words) or 48-bit width (4096 words). This section presents the results of studies by Sumitomo as a guide for estimating the maximum values for which moment characteristics can be calculated. In using the following formulas, be aware of the possibility of errors, for which Sumitomo cannot assume responsibility. Always thoroughly verify the accuracy of all formulas before using them.

9.10.1 Zero-Degree Moment Characteristics and Surface Area

Measurement of binary zero-degree moment characteristics is expressed in terms of a surface area (pixel count) for each object. Thus the maximum pixel count possible in 24-bit width without overflow is (2^{24} -1), or 16M pixels. Assuming a square input image, the size of a 4095 x 4095-pixel image is less than 2^{24} -1, so that even in the worst case in which the entire area of a 4095 x 4095-pixel image is a single object area, no overflow can result. Thus, even with input of a 4095 x 4095-pixel area, the surface area of 4096 separate objects can be counted. Similarly, in a grayscale histogram the horizontal axis of the histogram can be the area number or grayscale value, but the maximum pixel count (vertical axis) is the same in either case. These conclusions assume a square input image, though zero-degree moment characteristics do not depend on the aspect ratio of the input screen, and so the same conclusions apply to a screen of, for example, 16K x 1K pixels.

The case of grayscale-weighted area measurement is similar, though, because binary images are not involved, the limiting factor is not pixel count but the total of grayscale values of the pixels in each object area. 4K area measurement mode has 8 bits (or 256 grayscale gradients), and so can add a total of $(2^{24}-1)/(2^8-1) = (2^{16}+2^8+1)$ pixels. Assuming a square input screen, the total of 256 x 256 is less than $(2^{16}+2^8+1)$, so that even in the worst case (entire screen is a single object area, with all values 0FFh), no overflow can result. Because there are 12 bits (or 4096 gradients), a total of $(2^{24}-1)/(2^{12}-1) = (2^{12}+1)$ pixels can be added. Again assuming a square input image, the total of 64 x 64 is less than $(2^{12}+1)$, so that with 64 x 64 pixels, even in the worst case in which the entire screen is one object area and all values are 7FFh, no overflow can result.

2K area measurement mode limits the number of object areas to 2048, so that each area can have a value ranging from 0 to 2^{48}-1. If the number of grayscale gradient values is 12 bits or 4096, then it is possible to measure totals for up to $(2^{48}-1)/(2^{12}-1) = (2^{36}+2^{24}+2^{12}+1)$ pixels. Assuming a square input screen, the total of $2^{18} \times 2^{18}$ is less than $(2^{36}+2^{24}+2^{12}+1)$, so that with a total of $2^{18} \times 2^{18}$ pixels, no overflow can result even in the worst case in which the entire screen is a single area and all pixel values are 7FFh.

9.10.2 Primary Moment Characteristics

The maximum value of primary moment characteristics is determined by input screen size. The maximum value occurs in the case where one screen is occupied by one object area, and the vertical or horizontal size of the screen is large. If the input screen size is (Xw, Yx) and the upper left corner is the origin (0, 0), the maximum value of the primary moment around the origin can be expressed by the following formulas:

$$m_{10(\max)} = Yw \cdot \{0 + 1 + 2 + \ldots + (Xw - 1)\} = \frac{1}{2} Yw \cdot Xw \cdot (Xw - 1) \qquad (9\text{-}12)$$

$$m_{01(\max)} = Xw \cdot \{0 + 1 + 2 + \ldots + (Yw - 1)\} = \frac{1}{2} Xw \cdot Yw \cdot (Yw - 1) \qquad (9\text{-}13)$$

Example: Input screen size and maximum measurement value for primary moment characteristics:

Input screen size	Maximum value of primary moment characteristics	Maximum bit width
256 x 256 pixels	$2^8 \times 2^8 \times (2^8-1)/2 < 2^{23}$	23 bits
512 x 512 pixels	$2^9 \times 2^9 \times (2^9-1)/2 < 2^{26}$	26 bits
4096 x 4096 pixels	$2^{12} \times 2^{12} \times (2^{12}-1)/2 < 2^{35}$	35 bits

9.10.3 Secondary Moment Characteristics

The maximum value of secondary moment characteristics is determined by input screen size. The maximum value occurs in the case where one screen is occupied by one object area, and the vertical or horizontal size of the screen is large. If the input screen size is (Xw, Yw) and the upper left corner is the origin (0, 0), the maximum value of the secondary moment (m_{20}, m_{11}, m_{02}) around the origin can be expressed by the following formulas:

$$m_{20(\max)} = Yw \cdot \{0^2 + 1^2 + 2^2 + \ldots + (Xw - 1)^2\} = \frac{1}{6} Yw \cdot Xw \cdot (Xw - 1) \cdot (2Xw - 1) \qquad (9\text{-}14)$$

$$m_{02(\max)} = Xw \cdot \{0^2 + 1^2 + 2^2 + \ldots + (Yw - 1)^2\} = \frac{1}{6} Xw \cdot Yw \cdot (Yw - 1) \cdot (2Yw - 1) \qquad (9\text{-}15)$$

The maximum value of the m_{11} moment for the y^{th} line only is this:

$$m_{11(\max[y])} = y \cdot \{0 + 1 + 2 + \ldots + (Xw - 1)\} = \frac{1}{2} y \cdot Xw \cdot (Xw - 1) \qquad (9\text{-}16)$$

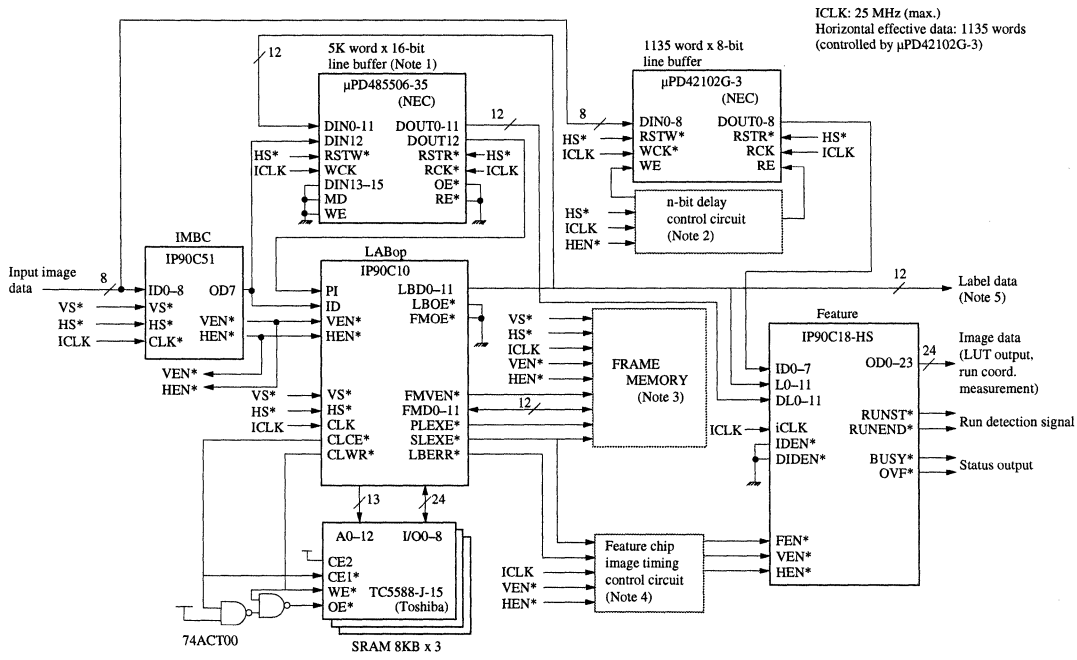
Therefore, the maximum value of the m_{11} moment for the screen as a whole is this:

$$m_{11(\max)} = \frac{1}{2} Xw \cdot (Xw - 1) \cdot \{0 + 1 + 2 + \ldots + (Yw - 1)\} = \frac{1}{4} Xw \cdot (Xw - 1) \cdot Yw \cdot (Yw - 1) \qquad (9\text{-}17)$$

Example: Input screen size and maximum measurement value for secondary moment characteristics:

Input screen size	Maximum value of secondary moment characteristics	Maximum bit width
64 x 64 pixels	$2^6 \times 2^6 \times (2^6\text{-}1) \times (2^7\text{-}1) / 6 < 2^{24}$	24 bits
128 x 128 pixels	$2^7 \times 2^7 \times (2^7\text{-}1) \times (2^8\text{-}1) / 6 < 2^{27}$	27 bits
256 x 256 pixels	$2^8 \times 2^8 \times (2^8\text{-}1) \times (2^9\text{-}1) / 6 < 2^{30}$	31 bits
512 x 512 pixels	$2^9 \times 2^9 \times (2^9\text{-}1) \times (2^{10}\text{-}1) / 6 < 2^{35}$	35 bits
4096 x 4096 pixels	$2^{12} \times 2^{12} \times (2^{12}\text{-}1) \times (2^{13}\text{-}1) / 6 < 2^{47}$	47 bits

9.11 Sample Circuit (Labeling Processing + Feature Extraction Processing) IP90C51 + IP90C10 + IP90C18



Note 1: This line buffer is used as a 1H horizontal line delay. It can also function as a line delay for the IP90C10 and IP90C18 chips.

Note 2: The frame memory configuration required by the IP90C10 generates a pixel delay in the horizontal axis. This line buffer is required to absorb that pixel delay. The delay value N is determined by the frame memory's design configuration.

Note 3: The frame memory is required by the IP90C10.

Note 4: Determines the processing area of the IP90C18. An IP90C51 can be used if necessary. Also, in some cases the circuit can be designed so that the IP90C18 operates while the IP90C10 executes two-dimensional labeling.

Note 5: The frame memory should be controlled so that the input image data and label data appear on the same horizontal line (without a 1-line delay).

Section 10: Guidelines for Avoiding Overheating

10.1 Guidelines for Avoiding Overheating

The heat generated by power consumption in an electronic device must not exceed the device's maximum allowable junction temperature. Temperatures exceeding this limit will reduce the device's reliability, sometimes dramatically, and can cause it to fail completely. The maximum junction temperature is determined by the heat relief properties of the device package; therefore, the maximum junction temperature and heat relief properties of the package determine how much power the device can use.

The IP90C18 requires no special precautions when operating at image clock frequencies of 25 MHz or less. However, if the device is used at frequencies above 25 MHz, the following precautions are necessary:

- For all image processing other than table conversion run coordinate detection and results memory dump output, set the I/O control register (IOCTL) results output enable flag (odenN) to 0, and set the output from the image data output (OD) pins to high level.
- For all image processing other than table conversion and results memory dump output, as well as for simultaneous execution of run coordinate detection (or when the I/O control register [IOCTL] results output select flag [sod] is set to 1 to output coordinate data from the coordinate counters through the image output [OD] pins), increase the load capacitance of the image output (OD) pins to 30 pF.
- When performing cumulative histogram processing in 2K area measurement mode, set the low-power cumulative histogram processing mode flag (HaccLP) in the mode register to 1.
- Grayscale projection processing in 2K area measurement mode using one-axis projection (H-axis projection only) should not be performed at frequencies over 25 MHz.

The device may not be reliable if these precautions are not followed.

Appendix A: Reference Bibliography

A.1 Reference Bibliography

The IP90C18 Feature LSI chip has been researched and developed to provide image processing and measurement through algorithms that determine the characteristics considered most important, such as area and center of gravity. The development objective has been to provide processing (1) from large-scale image screens, (2) for multiple object areas, (3) in real time simultaneously with raster-scan input, and (4) using accelerated processing of camera signals to the greatest extent possible.

Some characteristics with mathematically vague definitions (such as perimeter measurement) had to be defined for this purpose. However, virtually all characteristics used have definitions that are well accepted at the research level, and mathematically sound definitions were researched and incorporated into the final specifications. The following list includes those reference materials used in this study that are generally available. The researchers would also like to take this opportunity to express deepest appreciation for the efforts of these researchers in developing the field of image processing through research and instruction, as well as through the development of texts.

Because the IP90C18 Feature chip is intended to provide real-time processing through optimization of hardware, this chip has significantly different architecture than the examples used in these reference materials. For descriptions of the Feature chip's internal processing, as well as limiting factors and other information, see the text of this manual.

A.1.1 Digital Image Processing – General

- A. Rosenfeld and A. C. Kak, (Japanese translation by Nagao Makoto), "Digital Image Processing," Kindai Kagakusha, 1978.

Appendix B: Handling CMOS LSI Devices

B.1 Precautions

Sumitomo CMOS devices are designed and manufactured with the ability to withstand normal levels of stress during normal use and handling. However, failure to follow good usage procedures can reduce the device's reliability. To help ensure best results, always follow the precautions below in system design, handling, and storage.

(1) Ratings

Make sure the system is always within its established range of operating ratings, including operating voltage, input/output voltage, current, and temperature. Operating outside these ranges can increase failure rates quickly and dramatically, reducing reliability even though the device may appear to function normally. Also, when designing systems, keep in mind that failure rates vary according to operating voltage and temperature levels, even within the range of established ratings.

(2) Latch-Up

CMOS LSI devices are subject to a condition known as latch-up, which can occur when excessive extraneous noise is applied at the power supply or I/O signal pins, and which can eventually destroy the LSI. Because latch-up is a thyristor phenomenon, its effects remain in the device until the power supply is switched off. If latch-up occurs, shut off the power supply immediately, and do not restart the system until the cause of the problem has been eliminated.

(3) External Force

Always avoid exposing the system to vibration, shock, continued stress, or rapid temperature changes. These can damage the device's semiconductor chip, or break wire connections inside its plastic package.

(4) Static Electricity

Even though all signal pins are internally connected to anti-static protective circuits, exposure to static charges exceeding the voltage tolerance of these circuits can damage the device. Therefore, do not use insulating material in packaging and shipping containers, and make sure all materials used are electrically conductive and clearly labeled "anti-static."

(5) Temperature and Humidity

Make sure the device's physical surroundings stay within its temperature and humidity limits. Exposing the device to dust, salts, or corrosive gases such as SO_2 can cause leakage between pins, corrosion, and other problems. If the device must be used in a dirty, damp, or corrosive environment, coat the device's pins to prevent direct contact with corrosive gases or other contaminants.

(6) Frost Protection and Storage

Use and store the device in an environment that is controlled to prevent frost formation in case of rapid temperature changes. When storing the device, keep it in a tray in a horizontal position, and protect it from unnecessary loads.

SMI ASSP

Image Processing LSI Series

IP90C20

Rank Value Filter (RKFil)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.4

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Features and Functions

- The IP90C20 rank filter (RKFil) LSI chip sorts the 9 values contained in a 3x3-pixel local area, and outputs the maximum, median, and minimum values of those pixels in real time.
- The RKFil chip has a maximum processing rate (image data input rate) of 50 MHz, which enables it to be used with high-speed, high-density image data.
- Operating modes can be selected by level settings entered through the mode select signal pin on the chip, or by register settings entered through a CPU bus.
- The RKFil includes median filter processing of 1x3 or 3x1 local areas, using expanded mode settings made through a push-down register. It also provides a transparent mode that passes data through without filter processing.
- The chip also provides these features, designed to make it easy to use:
  - Low power demand, CMOS process
  - 5V single power source
  - TTL level input/output
  - Plastic QFP 44-pin package (molded area 10 mm^2, pin pitch 0.8 mm)

The following figure shows how the IP90C20 processes data:

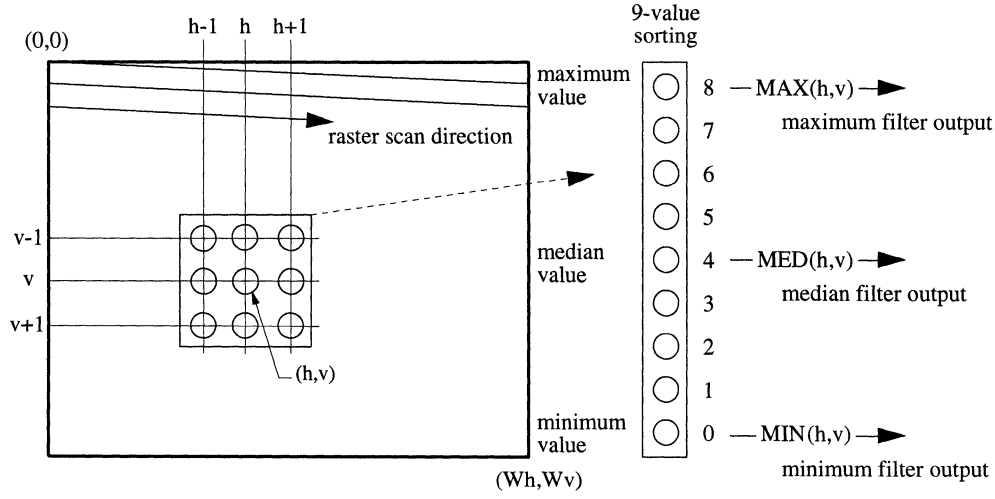
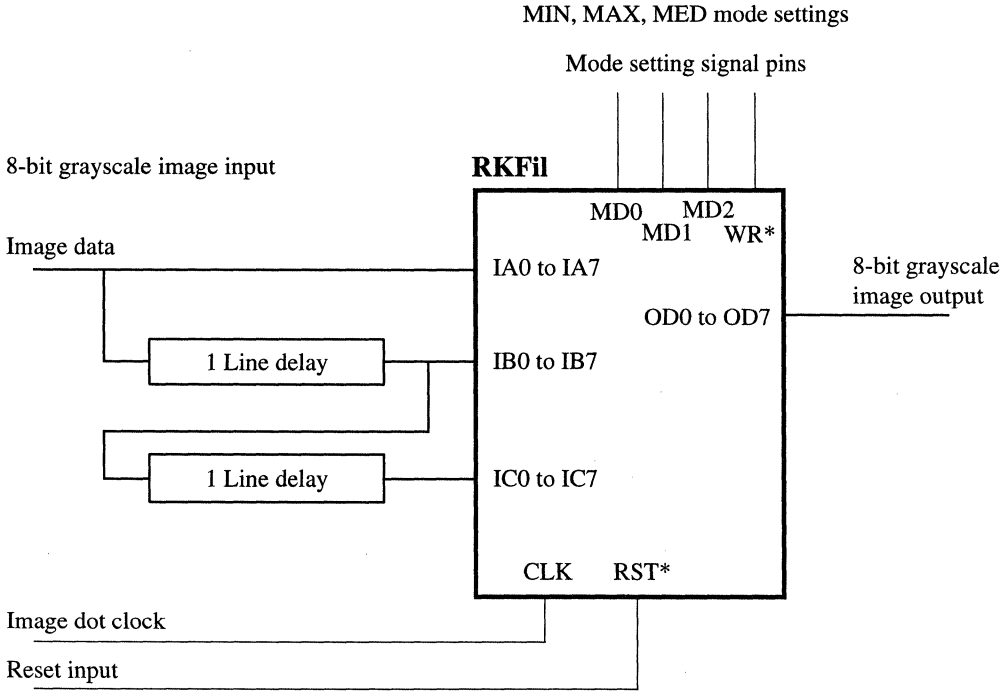


Figure 1: IP90C20 Rank Filter (RKFil) LSI: Input Data and Processing

The IP90C20 ranks the pixels in the 3x3-pixel local area in order of size, and identifies their locations using coordinates relative to the central pixel (h, v). The maximum value in the local area then becomes the maximum filter output MAX(h, v), the median value becomes the median filter output MED(h, v), and the minimum value becomes the minimum filter output MIN(h, v).

Section 2: Basic System Configuration and Operation

2.1 Basic System Configuration

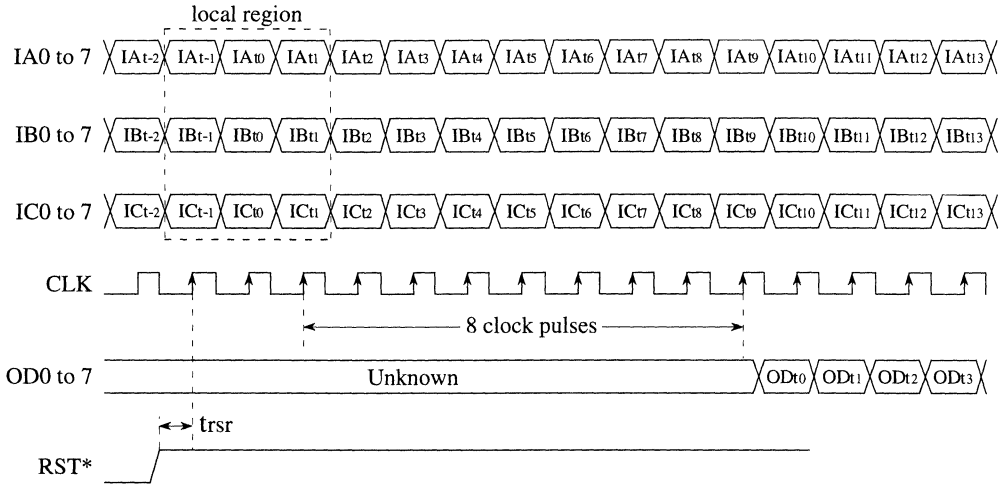


Note 1: An asterisk (*) after a signal name denotes inverse logic.

Note 2: The most and least significant bits (MSBs and LSBs) of the chip's grayscale image I/O signal pin sets are shown below:

Pin sets	MSB	LSB
IA0–IA7	IA7	IA0
IB0–IB7	IB7	IB0
IC0–IC7	IC7	IC0
OD0–OD7	OD7	OD0

2.2 Operating Description



Note 1: ↑: Image data is latched at the rising edge of the CLK signal.

Note 2: t_{rsr}: RST* signal release time is 15 ns.

The IP90C20 (RKFil) LSI chip generates a delay of 8 clock cycles between the input of 3x3 local area image data and the output of results. The RKFil chip's pipeline configuration provides real-time processing, producing the filtered output of one 3x3 local area with every clock cycle.

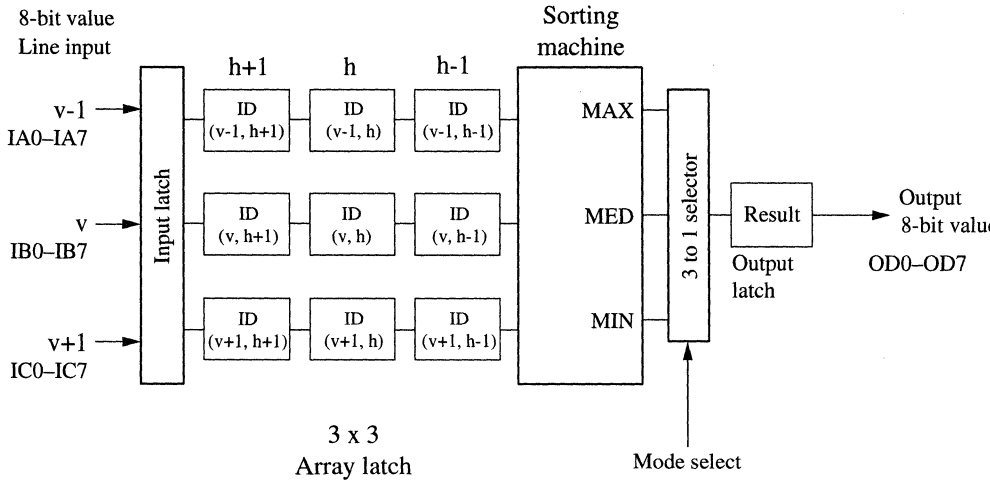
In 3x1 local area median filter mode, the RKFil chip outputs the median value of the three data points (IB_{t-1}, IB_{t0}, IB_{t1}) with the same timing as the output of OD_{t0} in 3x3 local area processing.

In 1x3 local area median filter mode, the RKFil chip outputs the median value of the three data points (IA_{t0}, IB_{t0}, IC_{t0}) with the same timing as the output of OD_{t0} in 3x3 local area processing.

In transparent data mode, the IP90C20 outputs the value of the center pixel in a 3x3 local area, corresponding to IB_{t0} in the above diagram. The latency from data input to output is the same as that shown above for max/mean/min filter mode, and the value of IB_{t0} is output with the timing of value OD_{t0}.

Section 3: Internal Configuration

3.1 Internal Configuration Diagram



Note 1: An asterisk (*) after a signal name denotes inverse logic.

Note 2: The above block diagram is intended as a functional description only, and does not indicate all the functions of the IP90C20 (RKFil) LSI chip. For descriptions of functions and timing, see the related sections of this document.

Section 4: Mode Settings

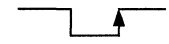
4.1 Basic Filter Mode Settings (MIN, MED, MAX, TP)

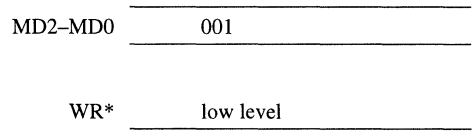
The available basic filtering modes include the three rank filter functions of the RKFil LSI, maximum filtering (MAX), median filtering (MED), and minimum filtering (MIN), plus a transparent mode (TP) in which no filter processing is performed. Select a basic filtering mode by using the signal pin input level (level setting) or by writing to registers using the write pulse signal (WR*) (register setting).

Level setting is designed for built-in image processing devices in automated equipment systems where rank filter functions are fixed. By entering high- or low-level signals at the MD1, MD0, and WR* pins as shown in the table below, the chip can be switched among MIN, MED, MAX, and TP mode as needed. Fixing WR* at low level eliminates the need to generate a write pulse, and thus also the need to connect to a CPU bus.

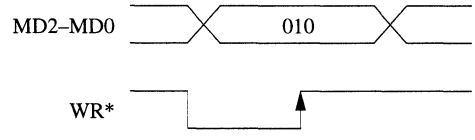
Register setting is designed for dynamic switching of general-purpose image processing devices and processing functions. In this case, the IP90C20 chip is connected to a CPU bus, and functions as a peripheral LSI with mode switching capability. The IP90C20 chip decodes address signals and uses the WR* signal input as a write pulse. The values of the MD1 and MD0 signals are latched in the mode register at the rise of the WR* signal, and can be used to switch functions according to the table below.

When using the IP90C20 chip in a basic filter mode, fix the MR2 signal low.

Filter mode	Mode Setting Signal Pins			Control pin WR*
	MD2	MD1	MD0	
MIN	Low	Low	Low	Low level or pulse 
MED	Low	Low	High	
MAX	Low	High	Low	
TP	Low	High	High	



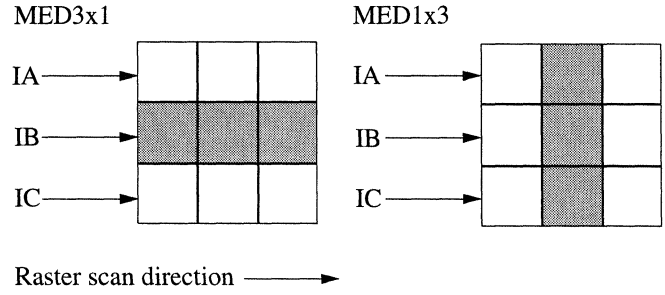
Signal Pin Level Settings for MED Filter Mode



Register Settings for MAX Filter Mode

4.2 Mode Settings for 3x1 and 1x3 Median Filter Processing

In addition to the basic mode settings described on the previous page, the IP90C20 has two expansion modes: one for filter processing to output median values from 3x1 areas (MED3X1), the other for filter processing to output median values from 1x3 areas (MED1X3).



As shown above, median filter processing of 3x1-pixel areas outputs the median value of the three pixels in a local area measuring 3 pixels horizontally by 1 pixel vertically, and median filter processing of 1x3-pixel areas outputs the median value of the three pixels in a local area measuring 1 pixel horizontally by 3 pixels vertically.

Select an expansion mode by writing into the mode register in a sequence of four cycles. The mode register is configured as a push-down shift register with a depth of four bits. In the basic modes, the push-down register values are ignored, but when the sequences shown in the following tables are used, the push-down register can be used to select the expansion modes. Glitches in WR* can result in incorrect push-down signals, resulting in abnormal operation. When this occurs, the image output signal (OD0–OD7) is not necessarily accurate. Selection settings remain valid until the next reset signal.

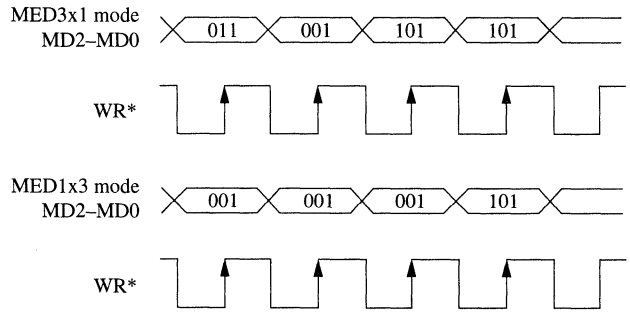
Switching between the 3x1 and 1x3 modes and among the basic filter modes (MAX, MED, MIN) is controlled by a high-level signal latched in the first level of the MD2 signal pin in the internal shift register. For this reason the MD2 signal in the fourth cycle of the write sequence for both expansion modes is a high-level signal.

3x1 Median Filter (MED3X1)

Write sequence	Mode setting signal pins		
	MD2	MD1	MD0
1st	Low	High	High
2nd	Low	Low	High
3rd	High	Low	High
4th	High	Low	High

1x3 Median Filter (MED1x3)

Write sequence	Mode setting signal pins		
	MD2	MD1	MD0
1st	Low	Low	High
2nd	Low	Low	High
3rd	Low	Low	High
4th	High	Low	High



Example: Mode Selection by Writing to Push-Down Register (four write cycles)

4.3 Sample Connections for Mode Selection

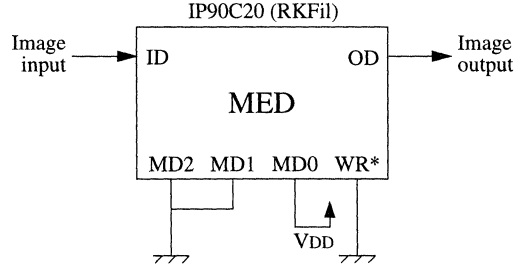
Section 4.1, "Basic Filter Mode Settings," described the two methods for switching between the MIN, MED, and MAX filter modes. The level setting method uses signals sent to the mode setting signal pins. The register setting method uses instructions sent on the write pulse signal (WR*) like a peripheral LSI chip.

Note that the examples given below are for illustration only, and are not intended to serve as specific applications. Actual applications should be designed with careful attention to the environment in which the device will operate, as well as to system specifications and timing requirements.

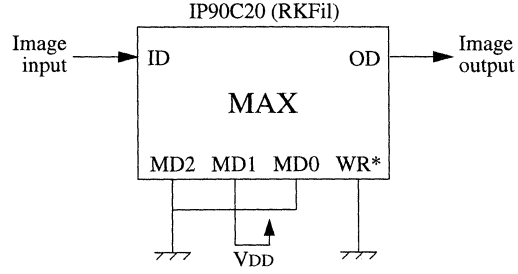
4.3.1 Selecting Basic Filter Modes Using Signal Pin Level Settings

If only median (MED) filtering is to be applied, the mode can be set relatively easily using wiring. To select maximum (MAX) filtering, change the level of the MD1 pin; to select minimum (MIN) filtering, change the level of the MD0 pin. When selecting a basic filter mode by level setting, the MD2 and WR* signals must always be held low.

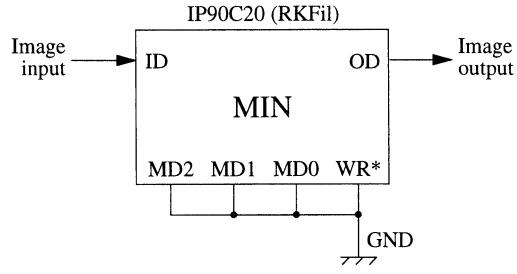
a) Median filter selected



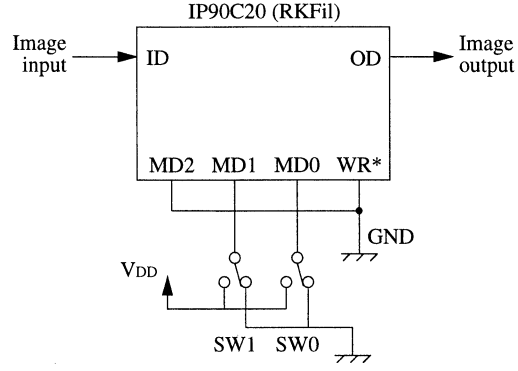
b) Maximum filter selected



c) Minimum filter selected



d) Switching, etc. selected

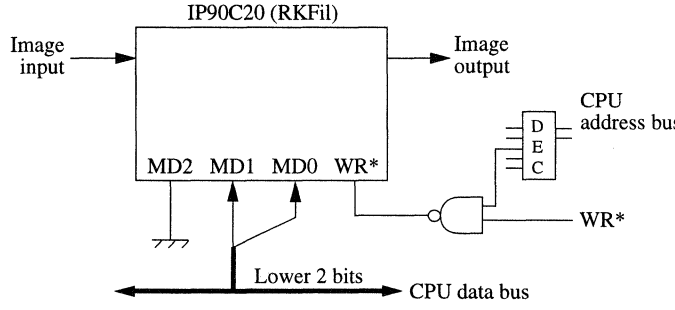


4.3.2 Selecting Basic Filter Modes Using Mode Register Settings

A basic filter mode (3x3 MAX, MED, MIN filtering) can be selected dynamically from the CPU by latching the mode signals (MD0, MD1, MD2) at the rise of the WR* signal.

This type of mode selection is maintained until the WR* signal decreases for the next mode selection, or until a reset signal is received.

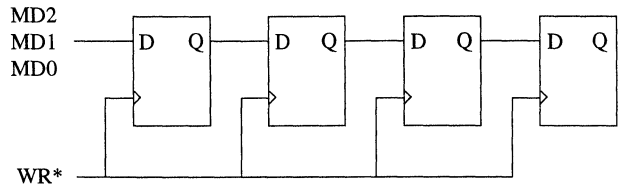
Output data becomes valid when mode switching ends. A minimum of 10 clock cycles are required for the new input data to fill the 3x3 local area, the data to be filtered, and the results to be output.



Values in the mode register are latched at the rise of the WR* pulse. Glitches in WR* can cause abnormal operation.

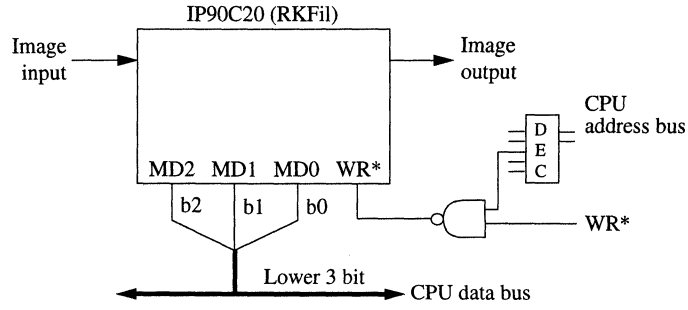
4.3.3 Sample Circuits for 3x1 and 1x3 Median Filtering Mode Settings

Select an expansion mode (3x1 median filtering (MED3x1) or 1x3 median filtering (MED1x3)) by using the mode register as a push-down shift register with a depth of four bits.



Internal Push-Down Register Configuration

The circuit connections are almost the same as for selecting basic filtering modes by register setting. However, the mode selection must use the MD2 signal pin, which classifies the setting as an expansion filtering mode. The combination of values written in the past determines the mode of operation.



Section 5: Pin Descriptions

5.1 Signal Pin Descriptions

The package is a 44-pin QFP (molded area 10 mm^2, pin pitch 0.8 mm).

Pin group	Pin	I/O	No. of Pins	Description
Image bus	IA0–IA7	I	8	Image input data (0 delay)
	IB0–IB7	I	8	1-line delay data input
	IC0–IC7	I	8	2-line delay data input
	OD0–OD7	O	8	Image output data
	CLK	I	1	Clock signal
Mode setting	MD0–MD2	I	3	Data bus 2-bit/mode input
	WR*	I	1	Register write enable (Note 1)
	RST*	I	1	System reset (Note 2)
Power supply	Vdd	PW	2	5V power supply
	GND	PW	4	GND
Total pin count		44		

Note: An asterisk (*) following a signal name indicates inverse logic.

Note 1: See the explanation of mode settings for methods of selecting filter modes (MAX, MED, MIN, TP).

Note 2: The reset signal pin (RST*) uses Schmitt trigger input with pull-up resistance.

5.2 Pin Description

Pin group	Pin	I/O	Pin No.	Description
Image input bus	CLK	I	42	
	IA7	I	31	MSB
	IA6	I	30	
	IA5	I	29	
	IA4	I	28	
	IA3	I	27	
	IA2	I	26	
	IA1	I	25	
	IA0	I	24	LSB
	IB7	I	41	MSB
	IB6	I	38	
	IB5	I	37	
	IB4	I	36	
	IB3	I	35	
	IB2	I	34	
	IB1	I	33	
	IB0	I	32	LSB
	IC7	I	6	MSB
	IC6	I	5	
	IC5	I	4	
	IC4	I	3	
	IC3	I	2	
	IC2	I	1	
	IC1	I	44	
	IC0	I	43	LSB

To prevent abnormal operation due to noise, connect all pins to the power supply or GND level.

Pin group	Pin	I/O	Pin No.	Description
Image output bus	OD7	O	12	MSB
	OD6	O	13	
	OD5	O	15	Note 1
	OD4	O	16	
	OD3	O	19	
	OD2	O	21	
	OD1	O	22	
	OD0	O	23	LSB
Mode selection	WR*	I	11	
	MD2	I	10	
	MD1	I	9	
	MD0	I	8	
	RST*	I	7	Note 2
Power supply	Vdd	PW	18	
and GND	Vdd	PW	39	
	GND	PW	14	
	GND	PW	17	
	GND	PW	20	
	GND	PW	40	

Note 1: Output current I_{ol} = 9 mA

Note 2: Schmitt-trigger input terminal with pull-up resistance 50 kW.

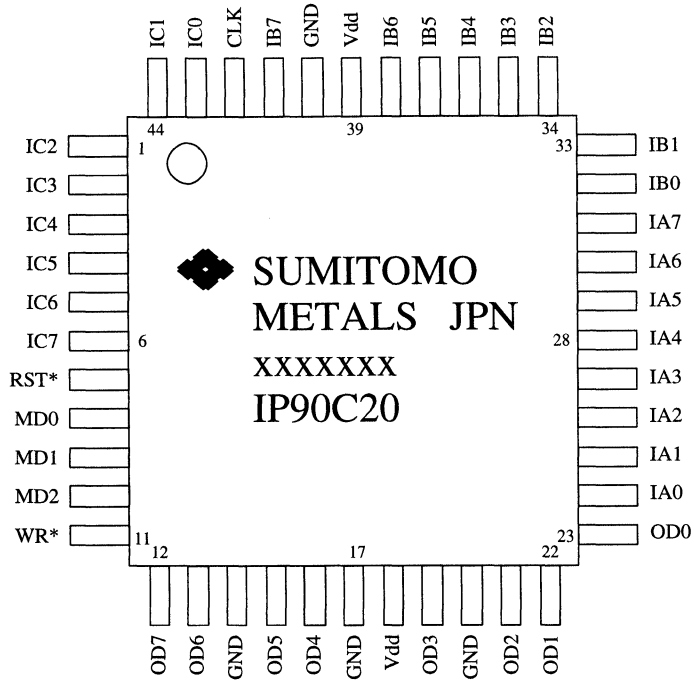
Note 3: An asterisk (*) following a signal name indicates inverse logic.

All input terminals are TTL level.

The most and least significant bits (MSBs and LSBs) of the chip's grayscale image I/O signal pin sets are shown below:

Pin sets	MSB	LSB
IA0–IA7	IA7	IA0
IB0–IB7	IB7	IB0
IC0–IC7	IC7	IC0
OD0–OD7	OD7	OD0

5.3 Pin Configuration



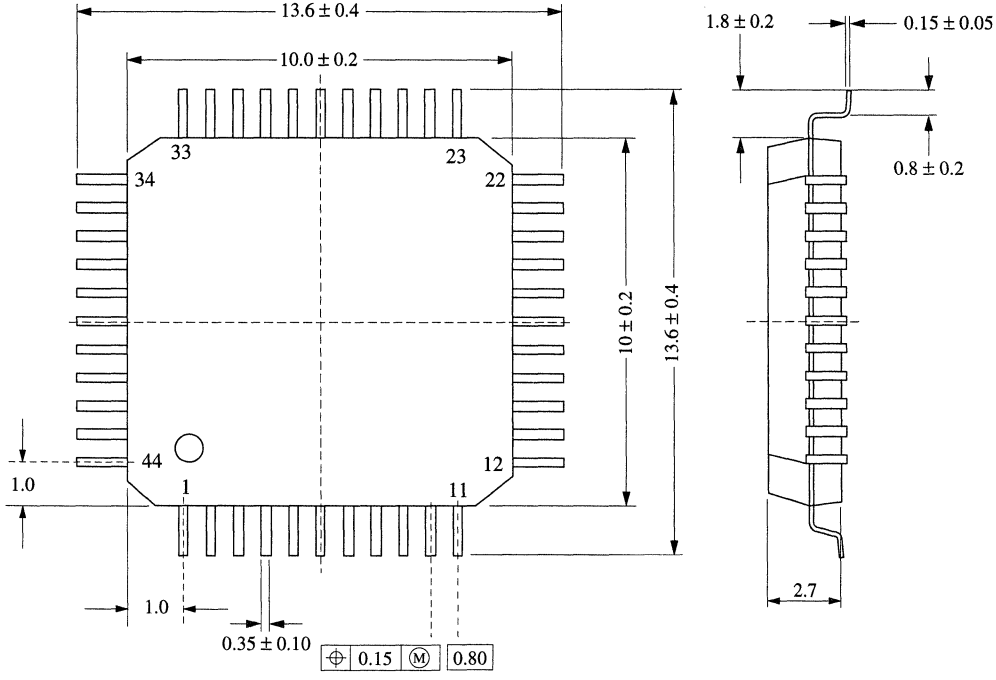
Pin Assignments

Pin	Name	Type	Pin	Name	Type	Pin	Name	Type	Pin	Name	Type
1	IC2	I	12	OD7	O	23	OD0	O	34	IB2	I
2	IC3	I	13	DO6	O	24	IA0	I	35	IB3	I
3	IC4	I	14	GND	PW	25	IA1	I	36	IB4	I
4	IC5	I	15	OD5	O	26	IA2	I	37	IB5	I
5	IC6	I	16	OD4	O	27	IA3	I	38	IB6	I
6	IC7	I	17	GND	PW	28	IA4	I	39	Vdd	PW
7	RST*	I	18	Vdd	PW	29	IA5	I	40	GND	PW
8	MD0	I	19	OD3	O	30	IA6	I	41	IB7	I
9	MD1	I	20	GND	PW	31	IA7	I	42	CLK	I
10	MD2	I	21	OD2	O	32	IB0	I	43	IC0	I
11	WR*	I	22	OD1	O	33	IB1	I	44	IC1	I

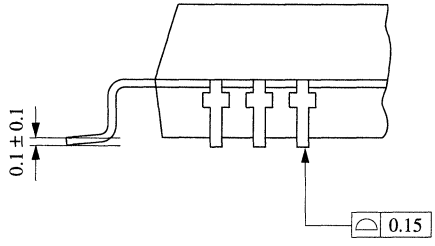
Section 6: External Dimensions

6.1 External Dimensions: 44-pin QFP

44-pin plastic QFP



Detail: Pin lead configuration



Section 7: Electrical Characteristics

7.1 Absolute Maximum Ratings

Parameter	Symbol	Rating	Unit
DC supply voltage	Vdd	-0.5 to 6.5	V
Input voltage	V_i	-0.5 to Vdd + 0.5	V
Output voltage	V_o	-0.5 to Vdd + 0.5	V
Output current	I_o	20	mA
Operating temperature	T_{opt}	0 to 70	°C
Storage temperature	T_{stg}	-10 to 80	°C

7.2 Recommended Operating Conditions

(Ta = 0 to 70°C)

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Power-supply voltage	Vdd		4.75	5.0	5.25	V
Input voltage	Vdd	TTL level	0	—	Vdd	V
Low level input voltage	V_{il}		0	—	0.8	V
High level input voltage	V_{ih}		2.2	—	Vdd	V
Positive trigger voltage	V_p	TTL level	1.2	—	2.4	V
Negative trigger voltage	V_n	Schmitt	0.6	—	1.8	V
Hysteresis voltage	V_{hys}	trigger	0.3	—	1.5	V
Input rising time	T_r	TTL level	0	—	200	ns
Input falling time	T_f		0	—	200	ns
Input rising time	T_r	Schmitt	0	—	10	ms
Input falling time	T_f	trigger	0	—	10	ms

Note 1: The reset signal pin RST* is a Schmitt-trigger input pin.

7.3 DC Characteristics

Vdd = 5 V ± 5%
Ta = 0–70°C

Parameter		Symbol	Condition	Min.	Typ.	Max.	Unit
Quiescent supply current (Note 1)		I_l	Vi = Vdd or GND		0.1	200	µA
Input ground voltage		V_{ic}	Ii = 18 mA	-1.2			V
Output OFF leakage current (Note 2)		I_{os}	Vo = 0 V			250	mA
Input leak current	Normal	I_i	Vi = Vdd or GND		10 pA	10	µA
	With pull-up resistor (50 kΩ equivalent, note 3)	I_i-P_u	Vi = GND	-45	-131	-320	µA
Output current	Low level	I_{ol}	V_{ol} = 0.4 V	9.0			mA
	High level	I_{oh}	V_{oh} = 2.4 V	-0.5			mA
Output voltage	Low level	V_{ol}	I_{ol} = 0 mA			0.1	V
	High Level	V_{oh}	I_{oh} = 0 mA	2.6	3.4		V

Note 1: Values exclude quiescent supply current to pull-up and pull-down resistance.

Note 2: Output OFF-state leakage current is the absolute value of current to one LSI terminal for 1 second or less.

Note 3: Only the reset signal pin RST* has pull-up resistance (50 kΩ equivalent).

Switching Characteristics (Vdd = 5 V ± 5%, Ta = 0–70°C)

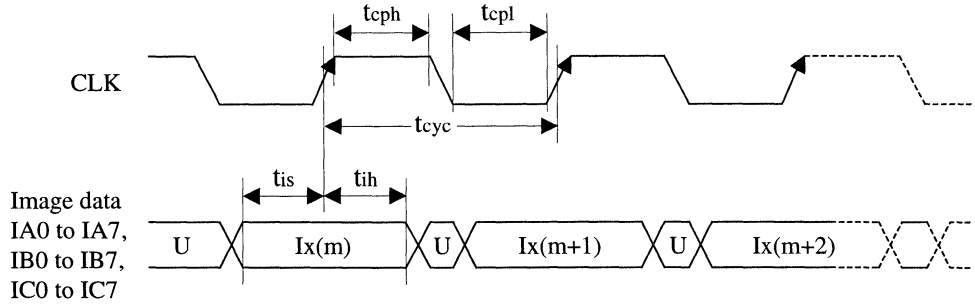
Item	Symbol	Condition	Min	Typ	Max	Unit
Output rise time	t_r	Output pins CL = 15 pF	—	1.54	—	ns
Output fall time	t_f	Output pins CL = 15 pF	—	1.42	—	ns

Input/Output Pin Capacitance (Vdd = V_i = 0 V)

Item	Symbol	Condition	Min	Typ	Max	Unit
Input signal pins	C_{in}	f=1MHz	—	10	20	pF
Output signal pins	C_{out}	f=1MHz	—	10	20	pF

7.4 AC Characteristics

Image Data Input Timing

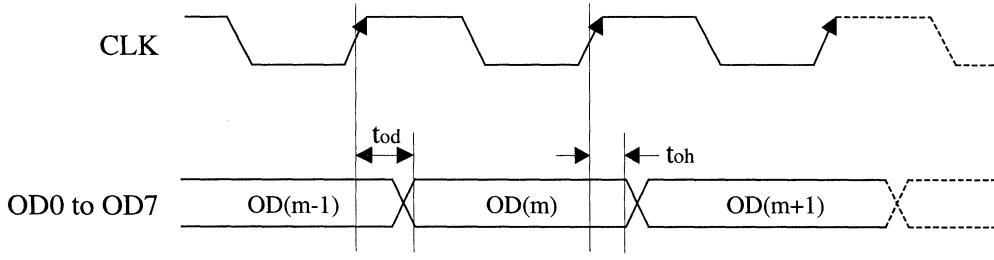


U: Unknown value

units: ns

Parameter	Symbol	Min.	Typ.	Max.
Video clock cycle	t_{cyc}	19	—	—
Video clock H level pulse	t_{cph}	8	—	—
Video clock L level pulse	t_{cpl}	8	—	—
Image data input setup time	t_{is}	2	—	—
Image data input hold time	t_{ih}	4	—	—

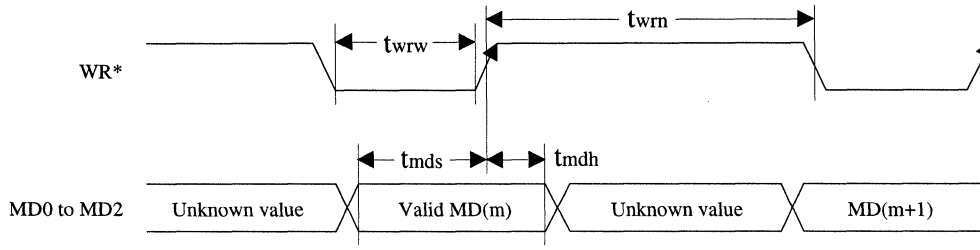
Image Data Output Timing Load Capacitance, CL = 30 pF



units: ns

Parameter	Symbol	Min.	Typ.	Max.
Delay time from CLK rise to data output	t_{od}	3.5	—	13
Hold time from CLK rise to data output	t_{oh}	3.5	—	13

Mode Set, Reset Cycle Timing

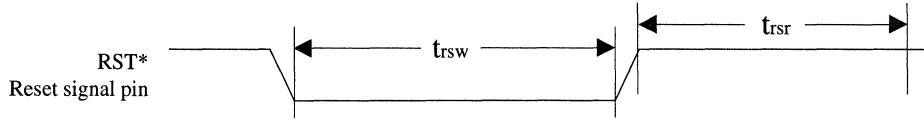


Writing from the CPU to the mode register need not be synchronized with the image data clock signal (CLK). WR* functions as a write clock from the CPU, and values can be written to the register at the rise of WR*, as shown above. However, after writing to the mode register, accuracy of internal operations is not assured for a certain period (t_{wrn}). Output data becomes valid after mode setting and after the fixed period (t_{wrn}), when the input data forms a 3x3 local area, is filter-processed, and is output a minimum of 8 clock cycles later.

units: ns

Parameter	Symbol	Min.	Typ.	Max.
WR* pulse width (low period)	t_{wrw}	8.0	—	—
WR* pulse interval	t_{wrn}	16.0	—	—
MD2, MD1, MD0 setup time for WR* signal rise	t_{mds}	8.0	—	—
MD2, MD1, MD0 hold time for WR* signal rise	t_{mdh}	3.0	—	—

Reset Timing



The reset signal can be input without regard to the timing of the image clock or the register write signal (WR*). However, image data input and values written to the register become valid beginning with the first image input cycle or register write cycle after the rise of the reset signal and after the reset release period (t_{rsr}). Therefore, output data does not become valid until after the rise of the reset signal and after the reset release period, when the next set of input data forms a 3x3 local area, is filter-processed, and is output a minimum of 8 clock cycles later.

units: ns

Parameter	Symbol	Min.	Typ.	Max.
RST* pulse width (low period)	t_{rsw}	8.0	—	—
RST* release time	t_{rsr}	15	—	—

Section 8: Sample Processing and Applications

8.1 Noise Elimination by Median Filtering

Figure 1 shows the rear license plate of an automobile in an image that includes Gaussian noise. This original image is then subjected to median filtering, with the result shown in Figure 2. The binary conversion process, frequently used in image measurement applications, is applied to both Figure 1 and Figure 2, with the results shown in Figure 3 and Figure 4. The results show that filtering has virtually eliminated the individual pixels of spike-type noise.

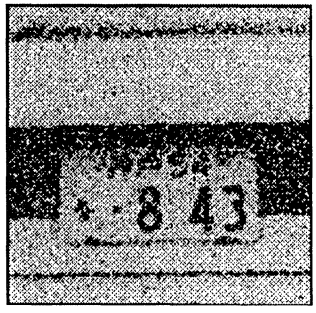


Figure 1:
Original image,
140x140 pixels

Binary conversion:

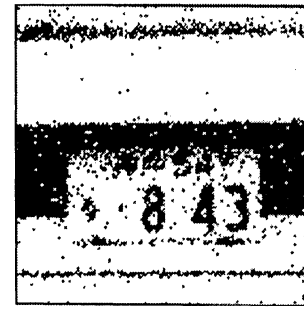


Figure 3:
Binary conversion performed directly on original image

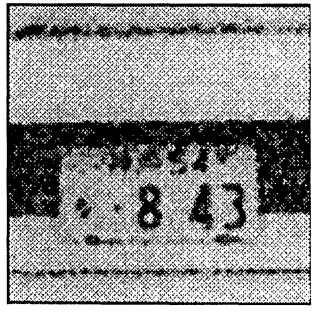


Figure 2:
After median filtering

Binary conversion:

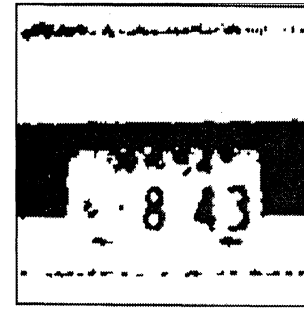


Figure 4:
Binary conversion performed on median-filtered image

Noise Elimination By Median Filtering

8.2 Sample Median Filtering Process

The following example of median filtering illustrates the typical operation of a rank filter. Photo 1 shows an image of a printed letter "C" approximately 60 x 60 pixels in size, obtained by sampling. Figure 1 shows a 3-dimensional graph of this image, using grayscale values as the vertical axis.

Photo 2 shows the original image from Photo 1 with artificial Gaussian noise added. Figure 2 shows a 3-dimensional graph of Photo 2. Each 3x3-pixel local area of the image from Photo 2 is passed through median filtering. The results are shown in Photo 3 and the 3-dimensional graph in Figure 3. The result is a fairly effective elimination of noise, which retains the boundary between character and background, demonstrating the effectiveness of median filtering.

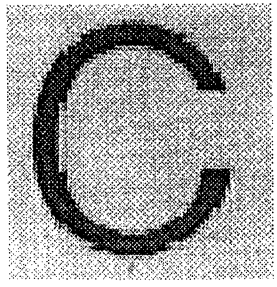


Photo 1: Original Image

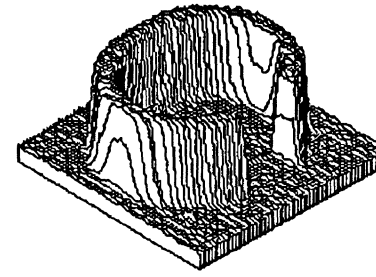


Figure 1: Three-Dimensional Graph of Original Image

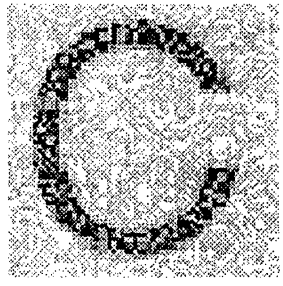


Photo 2: Noise-Added Image

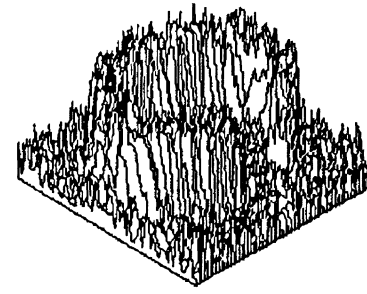


Figure 2: Three-Dimensional Graph of Noise-Added Image

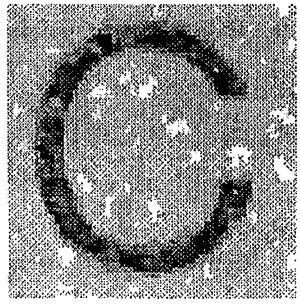


Photo 3: Median Filter Processed Image

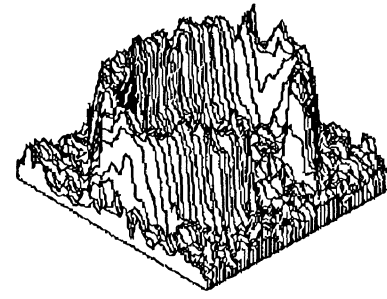
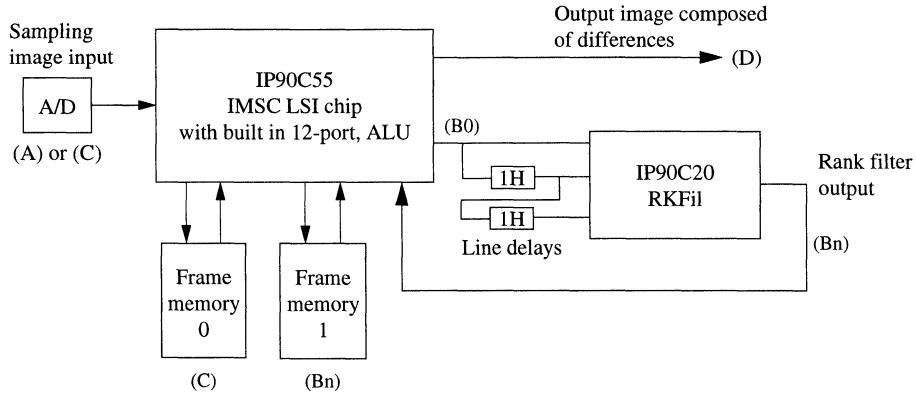


Figure 3: Three-Dimensional Graph of Median Filter Processed Image

8.3 Creation of Basic Screens for Shading Compensation

The median filter's excellent noise-elimination characteristics can be used to generate basic images from sampled images, using this general process:

1. Data is sampled for the basic image (point A in the figure below).
2. The data is then subjected to median filter processing (B0).
3. Step 2 is repeated as necessary to obtain a basic image with the desired resolution (Bn).
4. Image data is sampled for processing (C).
5. The difference between the sampled and processed images is calculated (D = C – Bn).
6. The resulting image composed of differences is output (D).



Overview of the IP90C55 IMSC (Image Data Stream Controller) LSI Chip

This is a 12-port data stream switching LSI with 8-bit I/O signal pin (digital image bus) ports.

- The chip has twelve 8-bit image data I/O ports allowing image data stream bus configurations between each set or ports.
- On-chip 16-bit ALU and barrel shifter enable inter-frame computations (and can be partitioned into two 8-bit ALU-plus-barrel shifter units).
- Eight internal area of interest (AOI) functions allow an image area of up to 64Kx64K-pixels to be designated.
- The chip has a maximum operating frequency of 40/20 MHz.
- The chip is contained in a PGA-181/QFP-184 package.

SMI ASSP

Image Processing LSI Series

IP90C25

Spatial & Logical Filter (SLFC)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 2.3

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Overview

1.1 Product Overview

The IP90C25/IP90C25-HS LSI chip performs filtering processes on 3 x 3-pixel local image areas at speeds up to 25 MHz (IP90C25) or 50 MHz (IP90C25-HS). The chip's high-speed capabilities make it suitable for use with high-definition television, as well as in industrial applications.

The IP90C25's operating modes can be broadly grouped into spatial filter modes, which process 8-bit grayscale image data, and logical filter modes, which process binary image data.

In spatial filter mode, the IP90C25 performs spatial filtering on 8-bit grayscale image data input, using a filtering coefficient that can be programmed using 8-bit two's-complement format.

In logical filter mode, the IP90C25 performs logical filtering on binary image data input. Logical formulas can be defined by writing truth-table values into a look-up table (LUT). Thus, a wide variety of logical filter processes can be executed by rewriting the LUT contents.

The LUT can also be used for grayscale conversion in spatial filter mode.

1.2 Features and Applications

Features

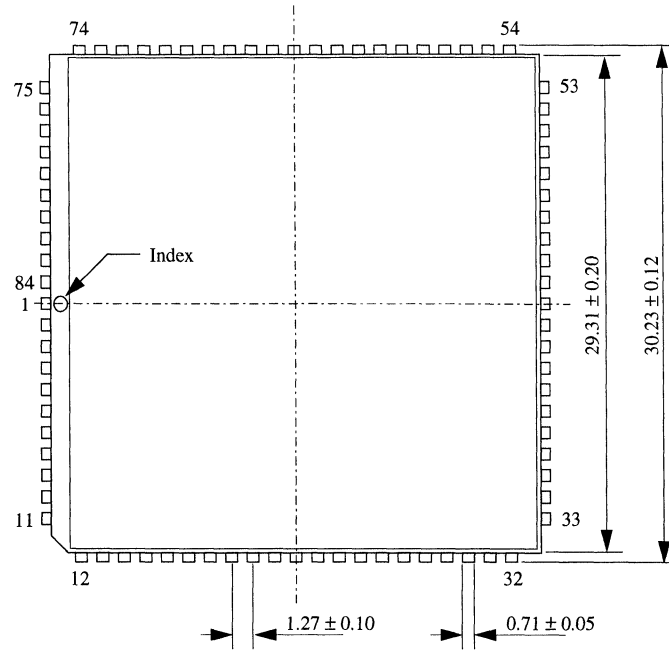
- Spatial filter mode:
  - -- Programmable coefficient strings
  - -- Sum-of-products calculation eliminates overflow
  - -- Right bit-shifter and adder for output compensation
  - -- LUT for grayscale conversion
- Logical filter mode:
  - -- LUT allows input of logical formulas
- Maximum operating frequency (pixel clock):
  - -- IP90C25: f_{max} = 25 MHz
  - -- IP90C25-HS: f_{max} = 50 MHz
- Local area size: 3 x 3 pixels
- Process: CMOS
- Power supply: 5V single source
- Package: 84-pin PLCC

Sample Applications

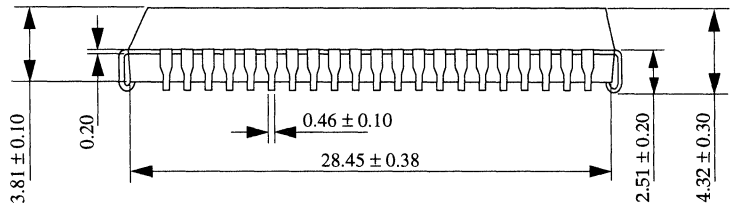
- Spatial filtering
  - -- Noise reduction
  - -- Edge emphasis
  - -- Edge extraction
  - -- Density conversion
- Logical filtering
  - -- Expansion
  - -- Reduction
  - -- Fine-line drawing
  - -- Outline extraction
  - -- Feature extraction: end points, intersections, lines (horizontal, vertical, diagonal), etc.
  - -- Elimination of single points

Section 2: Pins and Functions

2.1 Package Dimensions



units: mm



84-pin PLCC package

2.2 Pin Descriptions

Pin group	Symbol	No. of Pins	Type	Description
Image input bus	PIX1IN7–0	8	I	1st line of pixel input (Previous)
	PIX2IN7–0	8	I	2nd line of pixel input (Current)
	PIX3IN7–0	8	I	3rd line of pixel input (Following)
	PIXCLKI	1	I	Pixel clock input
	NLI	1	I	New line in
Image output bus	PIXOUT7–0	8	O	Pixel output
	PIXCLKO	1	O	Pixel clock output
	NLO	1	O	New line out
CPU bus	AB9–0	10	I	Address bus
	DB7–0	8	I/O	Data bus
	WE*	1	I	Write enable
	RE*	1	I	Read enable
	CE*	1	I	Chip enable
Test signals	TEST RESET	1	I	Test input signal
	TEST RAM	1	I	Test input signal
	TEST R/W	1	I	Test input signal
	TESTING2	1	O	Test output signal
	TESTING1	1	O	Test output signal
Power supply and GND	Vddi	5	PWR	Internal cell power supply
	Vddo	5	PWR	I/O buffer power supply
	VSSi	5	PWR	Internal cell ground
	VSSo	5	PWR	I/O buffer ground
Unconnected	NC	2	NC	
Total number of pins		84		

Note: Make sure the sync/force current for PIXOUT7–0, PIXCLK0, and NLO does not exceed 4 mA, and the load capacity does not exceed 20 pF. Also, make sure the sync/force current for DB7–0 does not exceed 2 mA, and the load capacity does not exceed 40 pF.

2.3 Pin Functions

2.3.1 Clock Power and Test Pins

PIXCLK1

Main clock for the device. Pixel data input through pins PIX1IN7–0, PIX2IN7–0, and PIX3IN7-0 is sampled at the rising edge of the PIXCLK1 signal.

PIXCLK0 (2 mA, 20 pF load max)

Output clock for the device. Output pixel data is latched in synchronization with the rising edge of this signal.

TEST RESET (Level = TTL)

Used for pre-delivery testing of the IP90C25. Connect this pin to Vdd during normal operation.

TEST RAM (Level = TTL)

Used for pre-delivery testing of the IP90C25. Connect this pin to Vdd during normal operation.

TEST R/W (Level = TTL)

Used for pre-delivery testing of the IP90C25. Connect this pin to Vdd during normal operation.

TESTSIG1 (Level = TTL)

Used for pre-delivery testing of the IP90C25. Leave this pin unconnected during normal operation.

TESTSIG2 (Level = TTL)

Used for pre-delivery testing of the IP90C25. Leave this pin unconnected during normal operation.

VSSi (Device ground)

Ground pin for internal logic cells and input buffer.

Vddi (Device power)

Power supply (5V) for internal logic cells.

VSSo (Device ground external power)

Ground pin for the output buffer.

Vddo (Device power external power)

Power supply pin for the output buffer.

Note: Make sure the power pins (VSSi, Vddi, VSSo, Vddo) are electrically connected to the outside.

2.3.2 Host Interface Pins

CE* (Chip Enable) (Level = TTL)

Enables host computer access (including read and write cycles) when a low-level signal is received.

RE* (Read Enable) (Level = TTL)

Functions with the CE* pin to enable the read cycle when both signals are low.

WE* (Write Enable) (Level = TTL)

Functions with the CE* pin to enable the write cycle when both signals are low.

Note: If the RE* and WE* signals are both low at the same time, the chip may not function normally.

AB9–0 (Address Bus) (Level = TTL)

Address bus for LUT and register addresses.

DB7–0 (Data Bus) (Level = TTL, 2 mA, 40 pF)

Data bus that transfers data to and from the host computer during read and write cycles. In a read cycle, data is read through the DB bus from the designated LUT or register. In a write cycle, data is sent to the chip through the DB bus and written into the designated LUT or register addresses.

2.3.3 Pixel Data Interface Pins

PIX1IN7–0 (Pixel Line 1 Input) (Level = TTL)

Receive input of the first line of pixel data, which is latched by the rising edge of the PIXCLKI clock signal. The first line of pixel data represents the line immediately above the current scan line.

PIX2IN7–0 (Pixel Line 2 Input) (Level = TTL)

Receive input of the second line of pixel data, which is latched by the rising edge of the PIXCLKI clock signal. The second line of pixel data represents the current scan line.

PIX3IN7–0 (Pixel Line 3 Input) (Level = TTL)

Receive input of the third line of pixel data, which is latched by the rising edge of the PIXCLKI clock signal. The third line of pixel data represents the line immediately following the current scan line.

NLI (New Line In) (Level = TTL)

Carries the signal indicating that the current pixel data input is valid. This signal must be low whenever valid data input is being received.

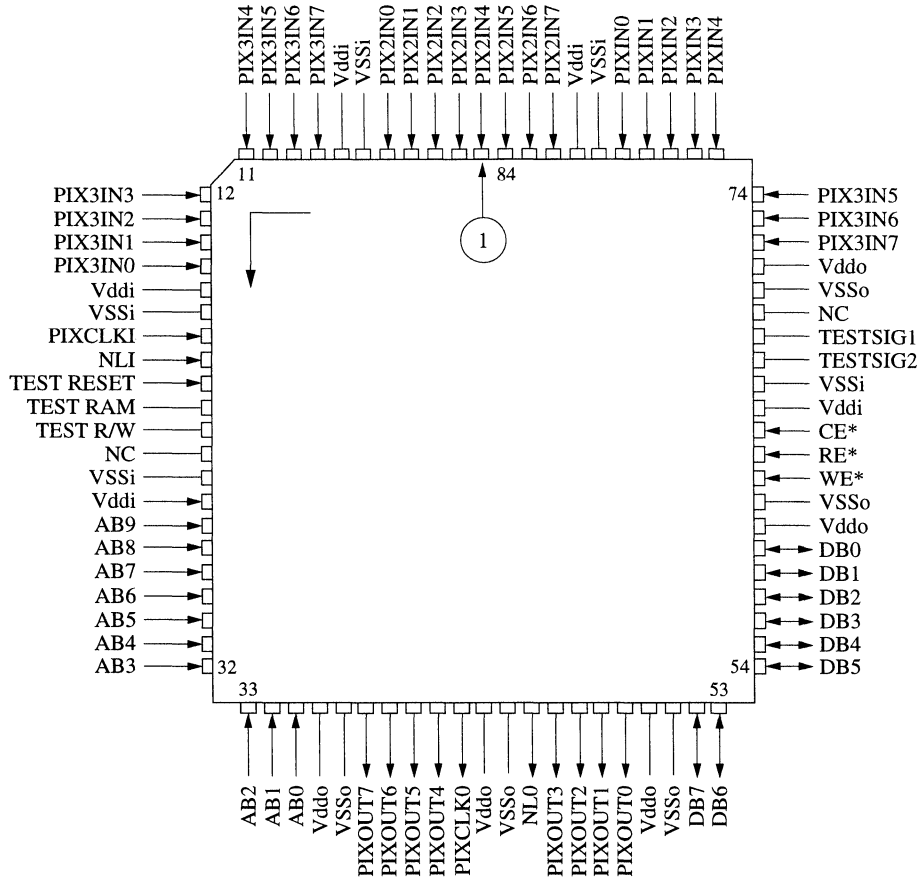
PIXOUT7–0 (Pixel Out) (Level = TTL, 2 mA, 20 pF load max)

Carry data output after filter processing. If this data is latched by the system, its should be latched to the rising edge of the PIXCLKO clock signal.

NLO (New Line Out) (Level = TTL, 2 mA, 20 pF load max)

Carries the signal indicating that the current pixel data output is valid. The NLO signal is delayed with respect to the NLI signal by the length of time required for filter processing.

2.4 Pin Configuration



2.5 Pin Assignments

Pin No.	Name	Type
1	PIX2IN4	I
2	PIX2IN3	I
3	PIX2IN2	I
4	PIX2IN1	I
5	PIX2IN0	I
6	VSSi	PWR
7	Vddi	PWR
8	PIX3IN7	I
9	PIX3IN6	I
10	PIX3IN5	I
11	PIX3IN4	I
12	PIX3IN3	I
13	PIX3IN2	I
14	PIX3IN1	I
15	PIX3IN0	I
16	Vddi	PWR
17	VSSi	PWR
18	PIXCLKI	I
19	NLI	I
20	TEST RESET	I
21	TEST RAM	I
22	TEST R/W	I
23	NC	NC
24	VSSi	PWR
25	Vddi	PWR
26	AB9	I
27	AB8	I
28	AB7	I

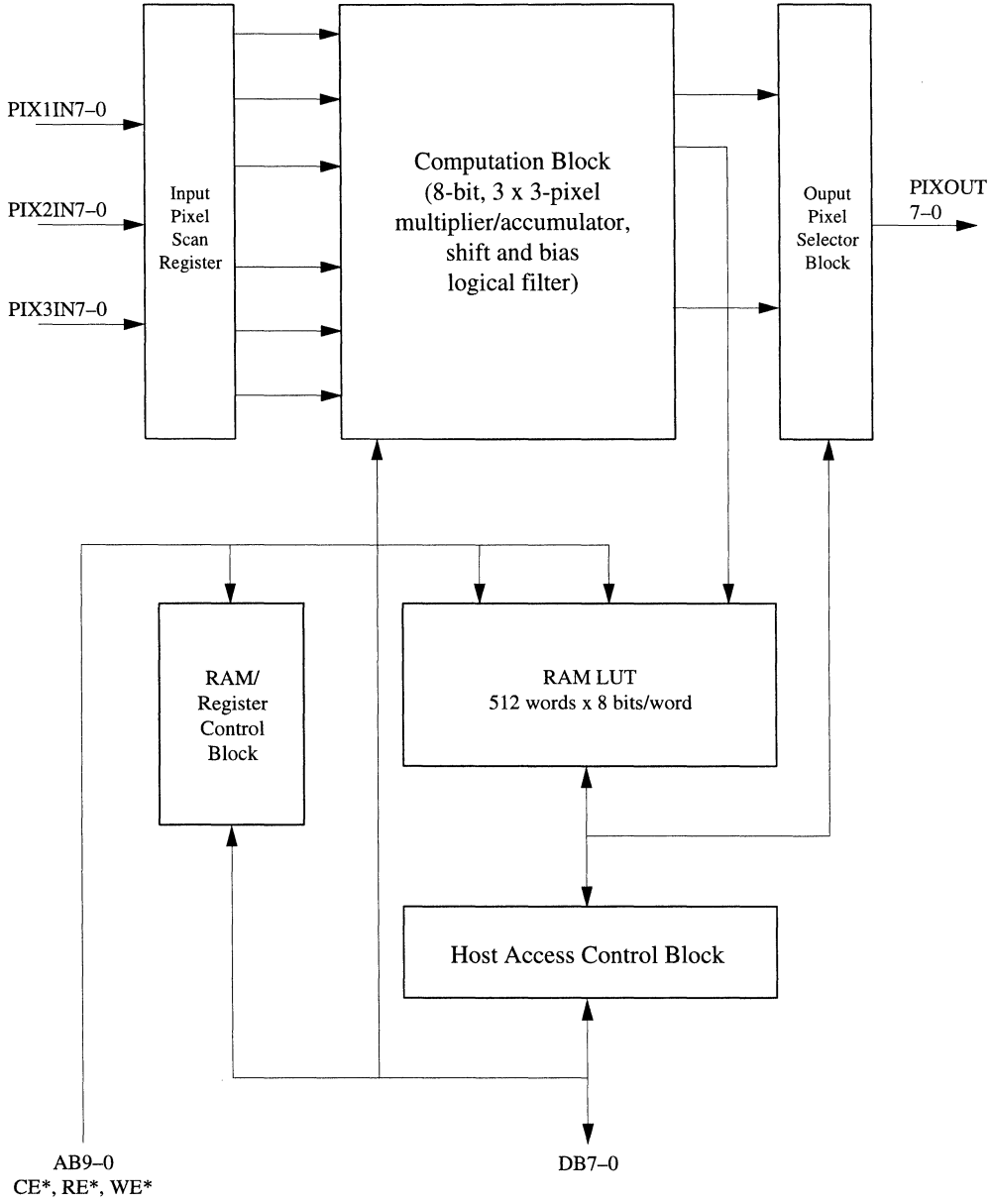
Pin No.	Name	Type
29	AB6	I
30	AB5	I
31	AB4	I
32	AB3	I
33	AB2	I
34	AB1	I
35	AB0	I
36	Vddo	PWR
37	VSSo	PWR
38	PIXOUT7	O
39	PIXOUT6	O
40	PIXOUT5	O
41	PIXOUT4	O
42	PIXCLKO	O
43	Vddo	PWR
44	VSSo	PWR
45	NLO	O
46	PIXOUT3	O
47	PIXOUT2	O
48	PIXOUT1	O
49	PIXOUT0	O
50	Vddo	PWR
51	VSSo	PWR
52	DB7	I/O
53	DB6	I/O
54	DB5	I/O
55	DB4	I/O
56	DB3	I/O

Pin No.	Name	Type
57	DB2	I/O
58	DB1	I/O
59	DB0	I/O
60	Vddo	PWR
61	VSSo	PWR
62	WE*	I
63	RE*	I
64	CE*	I
65	Vddi	PWR
66	VSSi	PWR
67	TESTSIG2	I
68	TESTSIG1	I
69	NC	NC
70	VSSo	PWR
71	Vddo	PWR
72	PIX1IN7	I
73	PIX1IN6	I
74	PIX1IN5	I
75	PIX1IN4	I
76	PIX1IN3	I
77	PIX1IN2	I
78	PIX1IN1	I
79	PIX1IN0	I
80	VSSi	PWR
81	Vddi	PWR
82	PIX2IN7	I
83	PIX2IN6	I
84	PIX2IN5	I

Note: An asterisk following a signal name indicates negative logic.

Section 3: Internal Block Functions

3.1 Block Diagram



3.2 Input Pixel Scan Register

This register samples pixel data input. Shift register configuration converts three successive rows of pixel data input into a 3 x 3-pixel local area for processing by the Computation Block.

3.3 Computation Block

This block performs all filter processing of the data sent from the input pixel scan register.

3.4 Output Pixel Selector Block

This block receives data from the computation block or LUT and transmits it as output pixel data in synchronization with the PIXCLKO clock signal.

3.5 RAM/Register Control Block

The RAM/Register Control Block coordinates the IP90C25 chip's operations using RAM timing controls and register settings.

3.6 RAM Look-Up Table (LUT)

The look-up table acts as a grayscale conversion table in spatial filtering mode, or a truth table in logical filtering mode. Reading or writing to RAM requires that the corresponding LUT bit in the control register be set to 1. To reduce the chip's power consumption, set non-essential LUT bits to 0.

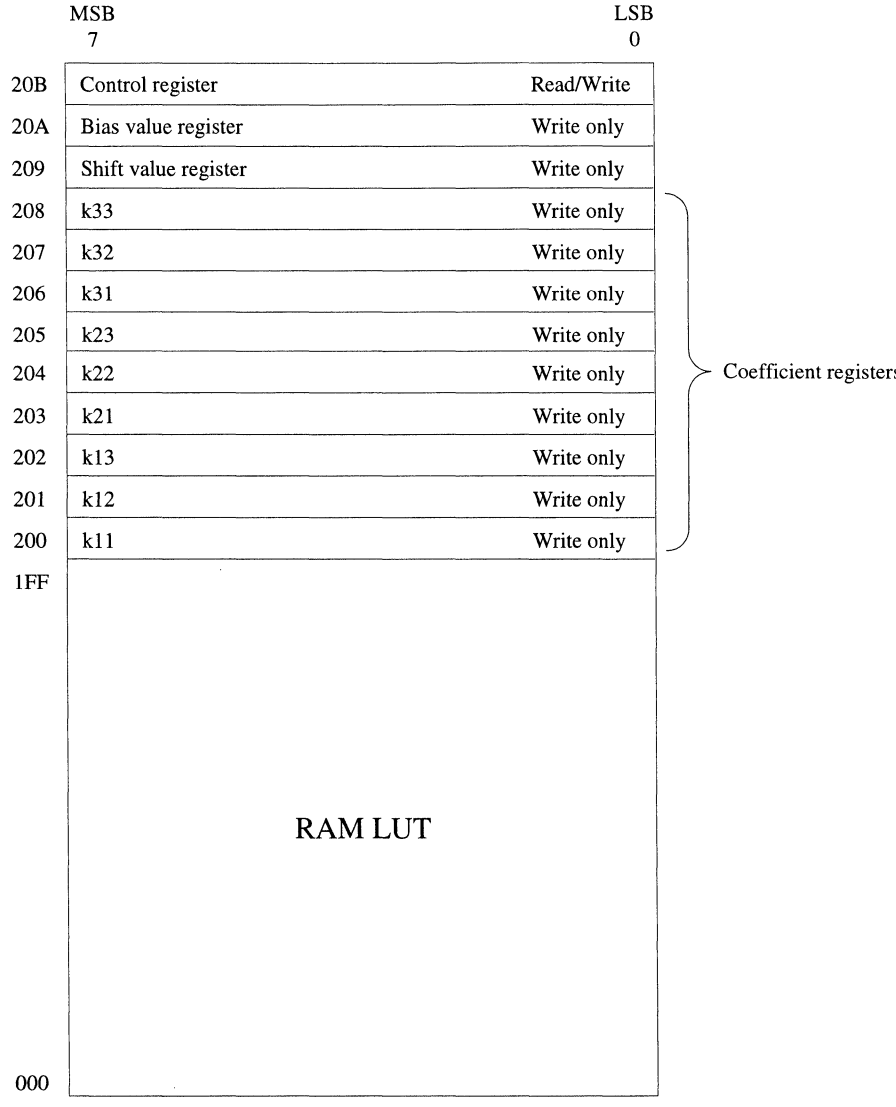
3.7 Host Access Control Block

The Host Access Control Block controls host computer access to the LUT and registers.

Note: When the host computer accesses the LUT or registers, data on the PIXOUT pins is not valid, and does not become valid again until 13 clock cycles after the last host access.

Section 4: Registers

4.1 Address Map



4.2 Control Register

Address 20Bh

Controls the processing mode and has a 3-bit configuration.

Bit 0: S/L Bit

Indicates whether the IP90C25 is in spatial filtering mode (when the bit has the value 1) or logical filtering mode (when the bit has the value 0).

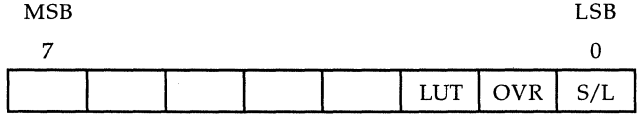
Bit 1: OVR Bit

Set to 1 when an overflow condition occurs during spatial filtering. The value is automatically reset within eight clock cycles after the control register is read.

Bit 2: LUT Bit

Indicates whether the LUT is used: contains a 1 when the LUT is used, and a 0 when the LUT is not used. Also controls LUT access from the host computer: contains a 0 when the LUT cannot be accessed.

Bits 3–7: Undefined



4.3 Bias Value Register

Address 20Ah

Sets the bias value used in spatial filter processing, in 8-bit, two's-complement notation.

4.4 Shift Value Register

Address 209h (Write only)

Sets the shift value (0–8) used in spatial filter processing.

4.5 Coefficient Registers

Address 20Ah (Write only)

Stores the nine coefficients used for convolutional computation in spatial filter processing. The coefficients must be in 8-bit, two's-complement notation.

Section 5: Operating Description

5.1 Mode Summary

The following table shows the combination of mode settings available in the IP90C25 chip.

Mode	LUT bit	S/L bit	Remarks
Spatial filtering: Direct	0	1	
Spatial filtering: LUT	1	1	MSB of LUT RAM address is 0
Logical filtering: Direct	0	0	Pixel data a33 is not output
Logical filtering: LUT	1	0	

5.2 Spatial Filtering: Direct Mode

The IP90C25 chip can perform spatial filtering at maximum speeds of 25 or 50 MHz (for the IP90C25 and IP90C25-HS, respectively). The chip processes local areas of 3 x 3-pixels in which each pixel (a_{ij}) is expressed as an unencoded 8-bit value, with a coefficient (k_{ij}) in 8-bit two's-complement notation. Coefficient values are programmable, and can be used to configure filters of Robert's operators, Laplacian operators, Sobel operators, etc.

Spatial filtering is executed as a two-dimensional convolution computation of the following form, using pixel values (a_{ij}) and coefficient values (k_{ij}):

$$C(m,n) = \sum_{i=1}^{3}\sum_{j=1}^{3} k(i,j) \cdot a(m-2+i, n-2+j)$$

If the results of convolution computation C(2,2) with respect to the central pixel of the area of interest (a_{22}) are represented as b, then b can be described as follows:

$$b = \sum_{i=1}^{3}\sum_{j=1}^{3} k(i,j) \cdot a(i,j)$$

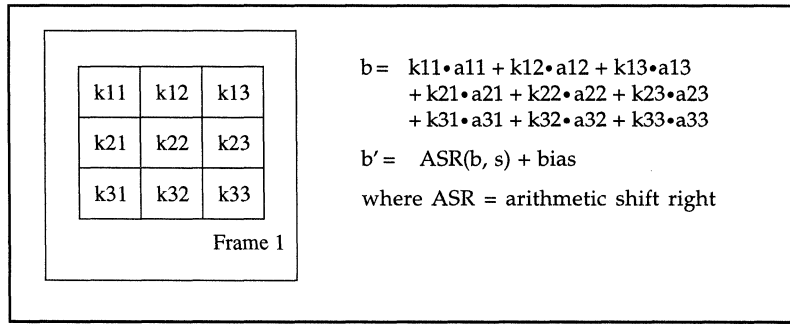
The value b is then right-bit shifted and added to the bias value, and so becomes the final result of logical filtering processing, b':

$$b' = ASR(b,s) + bias$$

where:

ASR = arithmetic shift right

S = shift value



Spatial Filtering: Direct Mode

Note: Pixel and coefficient determinants:

To simplify this discussion, the computation process is described in terms of pixel values (a_{ij}) and coefficient values (k_{ij}). However, the actual relation between input pixels and determinants is as follows:

- Determinant i indicates the row number. For pixels input from the PIX1IN signal the value i = 1.
- Determinant j indicates the relative position of pixels within the row. The value j = 1 indicates the oldest pixel, and j = 3 indicates the newest pixel.

5.2.1 Data Types

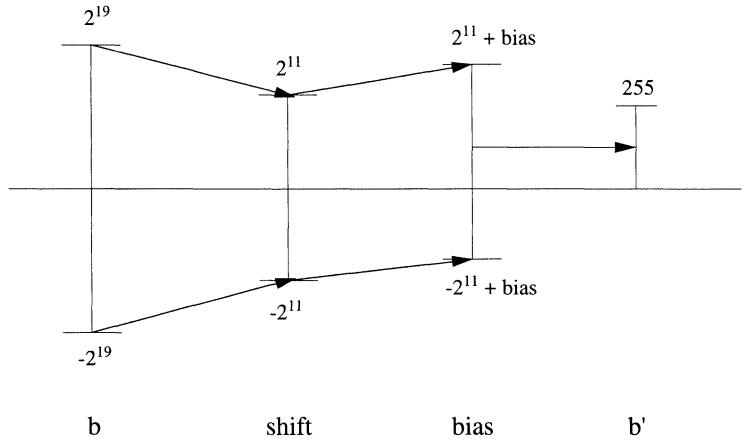
The results of convolution computation are expressed in 18-bit, two's-complement form. Following this step, an arithmetic right-bit shift (determined by the shift register, maximum 8 bits) is applied to reduce the data range, and a bias value (also expressed in 8-bit, two's-complement notation as determined by the bias value register) is added.

	Signed/Unsigned	Range	Input/Output/Internal
PIXIN data (a_{ij})	unsigned	0 to 255	input
Mask data (k_{ij})	signed	-128 to 127	input
Shift	unsigned	0 to 8	input
Bias	signed	-128 to 127	input
b	signed	-293,760 to 291,465	internal
b'	unsigned	0 to 255	internal
PIXOUT data	unsigned	0 to 255	output

Data Types

5.2.2 Overflow Conditions

Overflow occurs only when the results of logical filtering are truncated to 8 bits. Results in excess of 255 are clipped to 255, and results less than 0 are set to 0. In either of these cases, the OVR bit in the control register is set to 1. The value of the OVR bit is then automatically reset within 8 clock cycles after the control register is read.



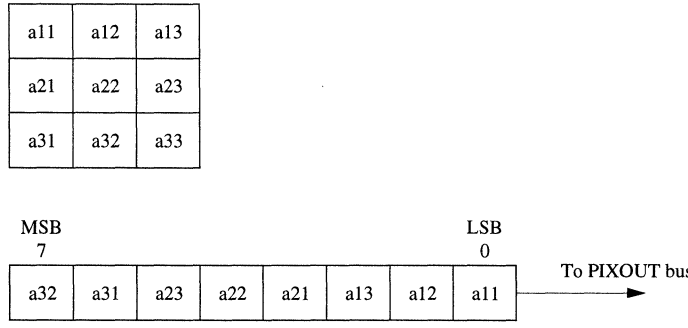
Internal Data Range

5.3 Spatial Filtering: LUT Mode

In LUT mode spatial filtering, the results of spatial filtering are used as an address location in the LUT. In this case, the most significant bit of the 9-bit RAM LUT address is set to 0. The value output with the PIXOUT signal then becomes the 8-bit value written to the corresponding address in the LUT.

5.4 Logical Filtering: Direct Mode

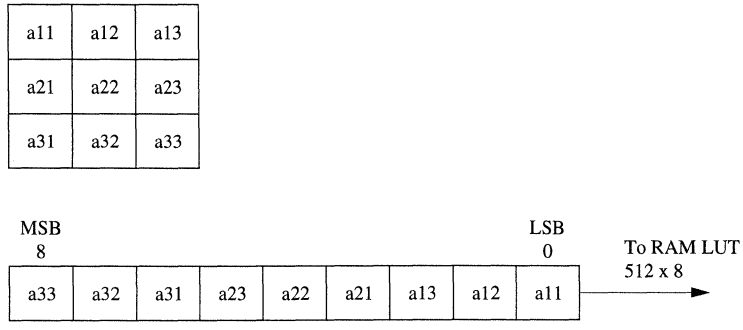
In direct mode logical filtering, the most significant bits of each input pixel are arrayed as shown in the following diagram. (The value of pixel a_{33} is deleted to form an 8-bit word.)



Logical Filtering: Direct Mode

5.5 Logical Filtering: LUT Mode

In LUT mode logical filtering, after the most significant bits of each input pixel are arrayed as shown in the following diagram, the results are used as an address location in the LUT.



Logical Filtering: LUT Mode

The LUT is used to define 8-bit output values corresponding to each of the 512 possible sequences of pixel values, thereby constructing a logical filter composed of outcome sequences. The output is expressed in the form of a binary image, so any logical expression can be assigned to each bit plane in the 8-bit result.

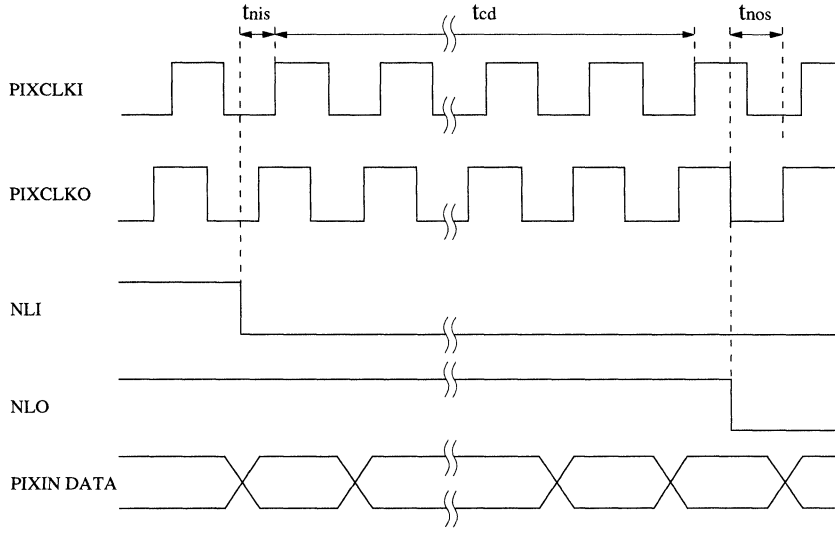
5.6 Startup Sequence

To help make sure the IP90C25 operates properly, always use this startup sequence:

1. Set the NLI signal to high.
2. Write setup values to the control register.
3. Write values to the coefficient registers (if not used, write dummy data).
4. Write values to the shift registers (if not used, write dummy data).
5. Write values to the bias registers (if not used, write dummy data).
6. Write values to the LUT if needed.
7. Read the control register, then wait ten clock cycles.
8. Begin filter processing.

5.7 Relation of NLI and NLO

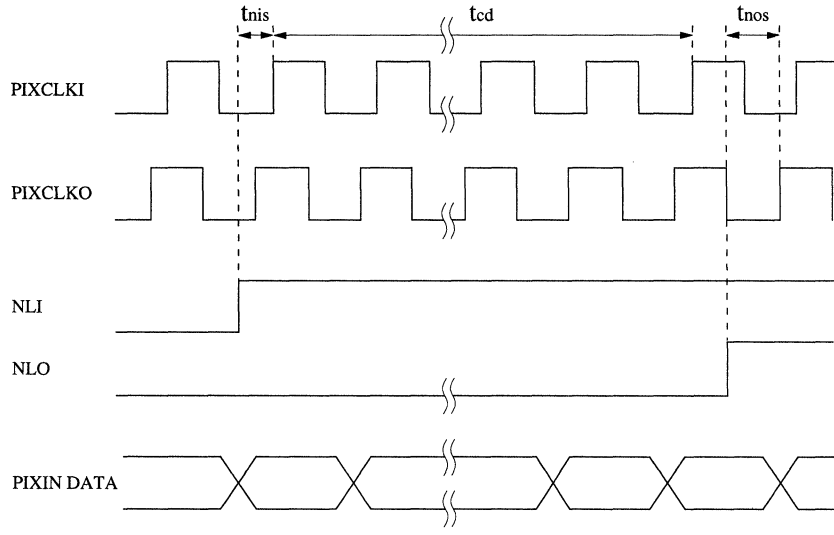
The NLO signal controls when valid output data is present on the PIXOUT output bus, and is delayed by a given number of clock cycles from the NLI signal. The length of the delay depends on the operating mode. Neither NLI nor NLO are affected by filter processing, so PIXOUT data is processed continually regardless of the values of NLI and NLO. The relation between the two signals and the delay is shown below.



Note: See "Image Interface Timing" on page 21 for the values of t_{nis} and t_{nos}.

Relation of Falling Edge of NLI and NLO

The delay is set by the number of clock cycles between the sampling of a given pixel from the PIX2IN input signal and output of the results of that pixel in the position of pixel a_{22}. The delay factor t_{cd} is measured from the first rise of the PIXCLK1 signal after NLI changes.



Note: See "Image Interface Timing" on page 21 for the values of t_{nis} and t_{nos}.

Relation of Rising Edge of NLI and NLO

NLI-to-NLO Delay

Mode	Delay (t_{cd})
Spatial filtering: Direct	32 cycles
Spatial filtering: LUT	38 cycles
Logical filtering: Direct	8 cycles
Logical filtering: LUT	14 cycles

Section 6. Electrical Characteristics

6.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Supply voltage	Vdd	-0.3 to 6.5	V
Input voltage	Vin	-0.3 to Vdd + 0.3	V
Operating temperature	Topt	0 to 70	°C
Storage temperature	Tstg	-10 to 80	°C

6.2 Recommended Operating Conditions

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Supply voltage	Vdd	—	4.75	—	5.25	V
H level input voltage	V_{IH}	—	2.0	—	—	V
L level input voltage	V_{IL}	—	—	—	0.8	V

6.3 DC Characteristics

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input leak current	I_{IN}	—	-10	—	10	µA
H level output voltage	V_{OH}	I_{OH} = -2 mA	2.4	—	—	V
L level output voltage	V_{OL}	I_{OL} = 2 mA	—	—	0.4	V
Off-state leak current	I_{OZ}	—	-10	—	10	µA

Note: The output shorting current is the absolute maximum voltage that can be applied to a pin for one second without causing a short.

6.4 Pin Capacitance

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input capacitance	C_{IN}	—	—	—	7	pF
Output capacitance	C_{OUT}	—	—	—	10	pF
I/O capacitance	C_{IO}	—	—	—	10	pF

6.5 AC Characteristics

Load capacitance = 30 pF.

Host Computer Interface Timing

4.75 V ≤ Vdd ≤ 5.25 V
VSS = 0 V
Ta = 0 to 70°C

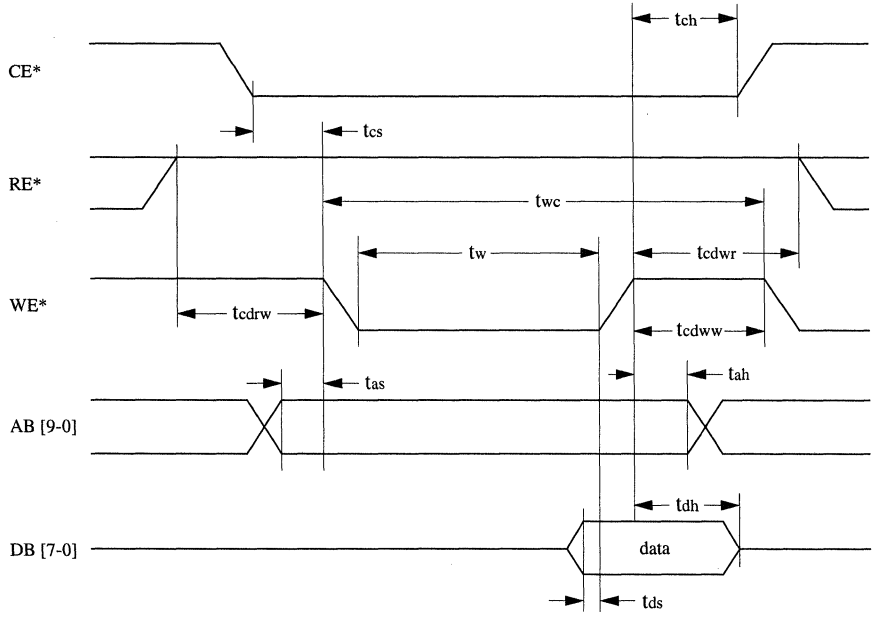
Parameter	Symbol	Min.	Max.	50 MHz
CE* setup time	t_{cs}	0 ns		0 ns
CE* hold time	t_{ch}	0 ns		0 ns
Address setup time	t_{as}	10 ns		10 ns
Address hold time	t_{ah}	0 ns		0 ns
Data setup time	t_{ds}	10 ns		10 ns
Data hold time	t_{dh}	10 ns		10 ns
Data float delay	t_{dhz}	—	20 ns	20 ns
Data bus active delay	t_{dlz}	20 ns	t_{acc}	20 ns
Data delay	t_{acc}	40 ns + 11t_{cyc}		260 ns
RE* pulse width	t_{r}	40 ns + 11t_{cyc}		260 ns
Read cycle time	t_{rc}	50 ns + 13t_{cyc}		310 ns
WE* pulse width	t_{w}	10 ns + 2t_{cyc}		50 ns
Write cycle time	t_{wc}	20 ns + 10t_{cyc}		220 ns
WE* high period	t_{cdww}	10 ns + 8t_{cyc}		170 ns
Write/read interval time	t_{cdwr}	10 ns + 8t_{cyc}		170 ns
Read/write interval time	t_{cdrw}	10 ns + 2t_{cyc}		50 ns
RE* high period	t_{cdrr}	10 ns + 2t_{cyc}		50 ns

Image Interface Timing

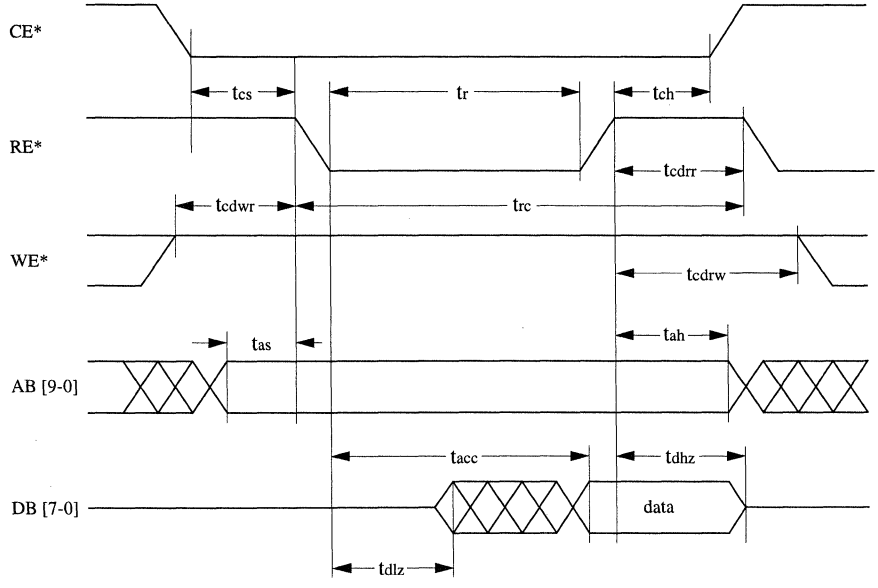
4.75 V ≤ Vdd ≤ 5.25 V
VSS = 0 V
Ta = 0 to 70°C

Parameter	Symbol	Min.	Max.	50 MHz
IP90C25 cycle time	t_{cyc}	38.0		ns
IP90C25-HS cycle time	t_{cyc}	18.4		ns
NLI minimum signal width	t_{nl}	10 ns + 11t_{cyc}		ns
NLI setup time	t_{nis}	4.0		ns
NLI hold time	t_{nih}	8.0		ns
Image input setup time	t_{pis}	4.0		ns
Image input hold time	t_{pih}	8.0		ns
Clock signal rise time	t_{cir}		2	ns
Clock signal fall time	t_{cif}		2	ns
NLI input rise time	t_{nir}		2	ns
NLI input fall time	t_{nif}		2	ns
NLO output setup time	t_{nos}	4.0		ns
NLO output hold time	t_{noh}	4.0		ns
Pixel output setup time	t_{pos}	4.0		ns
Pixel output hold time	t_{poh}	4.0		ns
NLO output delay	t_{nod}	2.0	20	ns
Pixel output delay	t_{pod}	2.0	20	ns
Clock output delay	t_{cod}	1.5	15	ns
Clock output rise time	t_{cor}		2	ns
Clock output fall time	t_{cof}		2	ns
NLO output rise time	t_{nor}		2	ns
NLO output fall time	t_{nof}		2	ns

Host Computer Write Cycle



Host Computer Read Cycle



Note: See "Host Computer Interface Timing" on page 20 for the values of the parameters shown above.

Image Input Timing

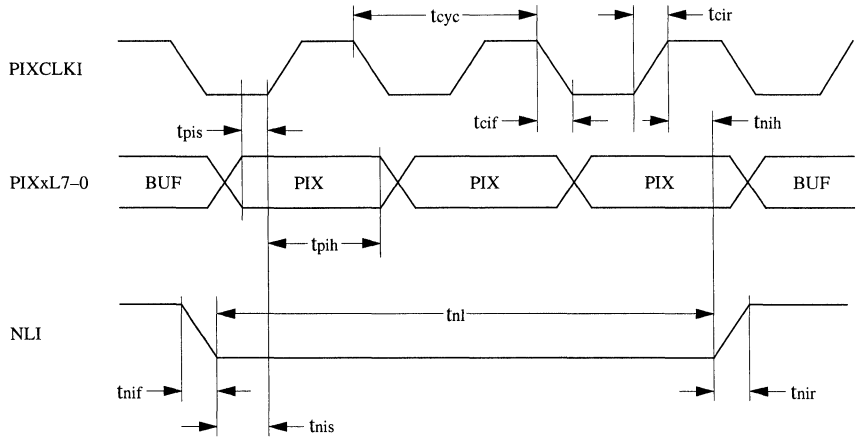
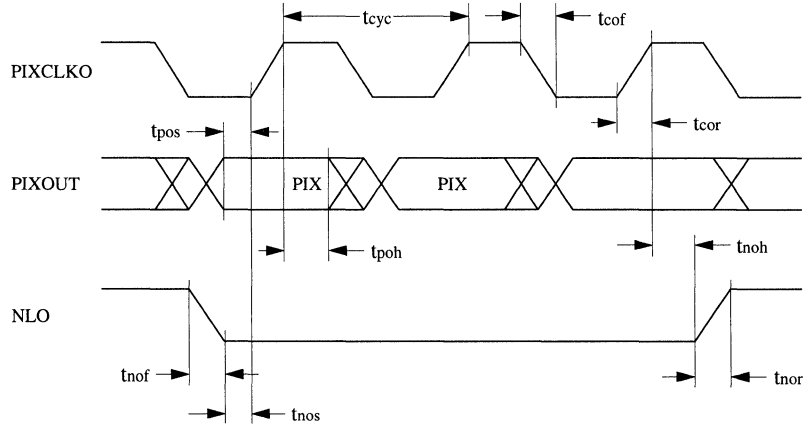
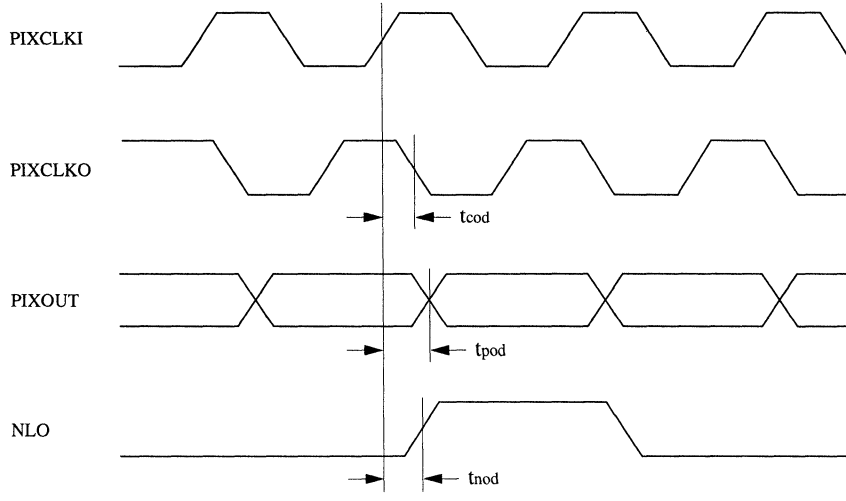


Image Output Timing



Note: See "Image Interface Timing" on page 21 for the values of the parameters shown above.

Image Output Timing (from PIXCLKI)



Note: See "Image Interface Timing" on page 21 for the values of the parameters shown above.

SMI ASSP

Image Processing LSI Series

IP90C31

Multiplier & Accumulator with 4 Multipliers (MAC4)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.4

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1. Overview

1.1 Functional Overview and Characteristics

The IP90C31 LSI chip is designed for high-speed product-sum calculations. It contains four 8-bit x 8-bit multipliers and a 40-bit accumulator, and has an optional setting that allows it to perform 16-bit x 16-bit multiplication and product-sum calculations. The chip outputs data in 16-bit floating decimal-point (floating-point) format, and includes a selector that enables access to any 20 bits on the 40-bit bus. The IP90C31 also includes a setting for rounding processing. Multiple IP90C31 chips can be connected through the sample register block to form a cascade connection.

The IP90C31 has the following characteristics:

- Floating-point format — 1-bit sign, 6-bit exponent, 9-bit mantissa
- Accumulator bit width — 40 bits
- External interface — CPU interface or direct terminal connection
- Maximum operating frequency — 30 MHz
- Power supply — 5-V single source
- Process — CMOS
- I/O interface — TTL-level compatible
- Package — 160-pin plastic QFP

1.2 Floating-Point Format

The IP90C31 output block includes a floating-point conversion circuit, which allows output from the IP90C31 to be passed directly to an IP90C32 chip.

- Total width 16 bits (bit 15–bit 0).

bit 15:	1 bit	sign bit	S	0 = pos, 1 = neg
bit 14–bit 9:	6 bits	exponent	e	0–63

  The actual exponent value is e-b, where b is the bias value.

bit 8–bit 0:	9 bits	mantissa	m

  Positive binary values (offset binary) indicate values after the decimal point, and do not include integer components.

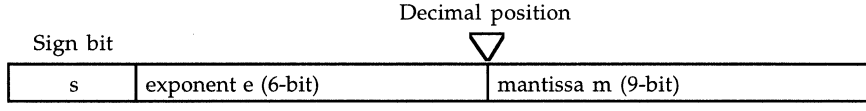
- The integer component is determined by the exponent.

$e \neq 0$	—>	integer component I = 1
e = 0	—>	integer component I = 0

- Zero (0) is expressed as e = 0 and m = 0.

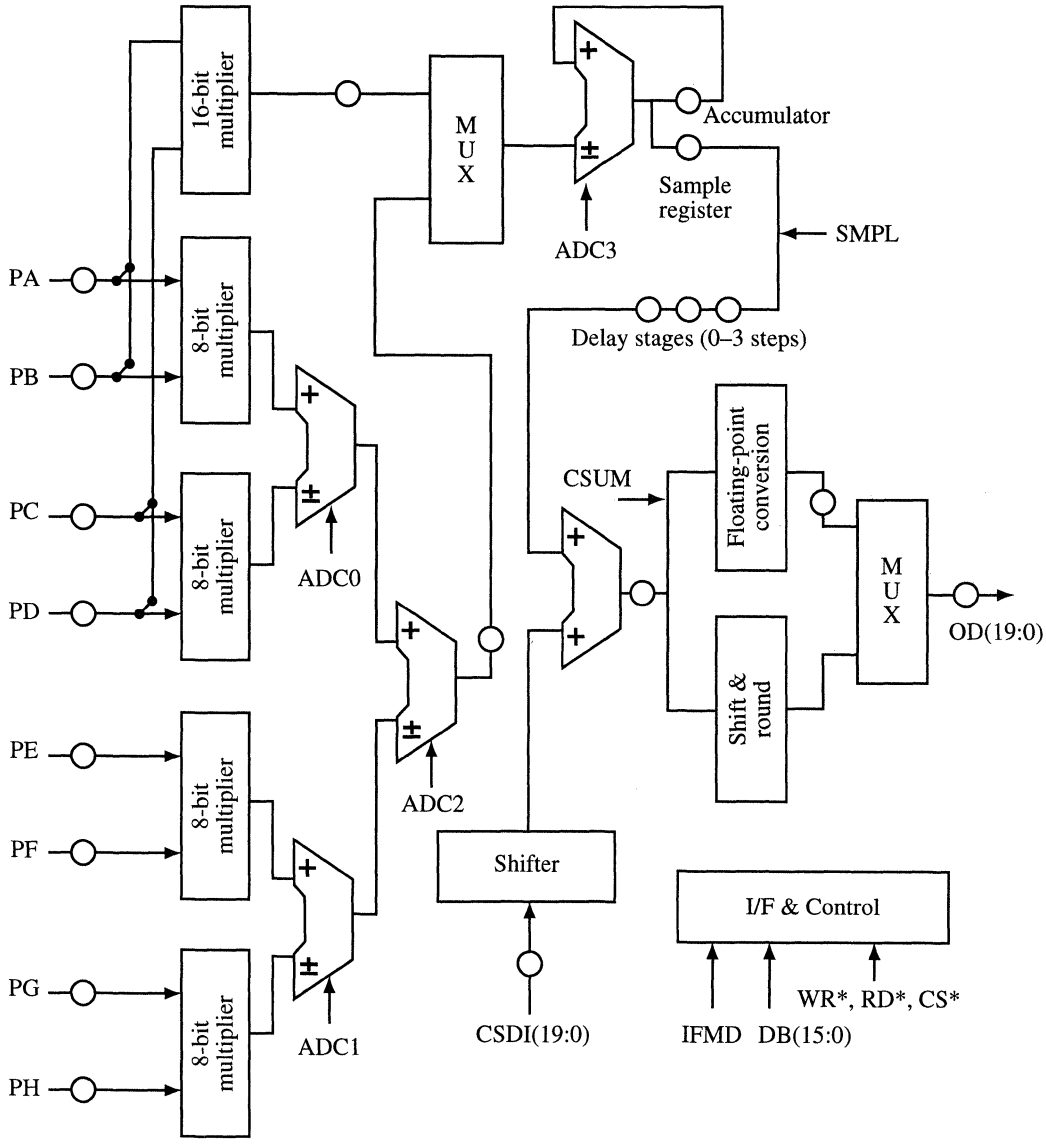
  Non-normal data (de-normalized values) are not handled, so that values with e = 0 are always processed as m = 0.

- In equation form, $(-1)^S \times I.m \times 2^{(e-b)}$.



Note: The IP90C31 always has a bias value b of 1.

1.3 Block Diagram



Note: the "○" symbol indicates pipeline registers.

Section 2. Pins and Functions

2.1 Pin Lists and Functional Descriptions

Signal group	Pin symbol	No.	Type	Function
Control signals	RST*	1	I	Reset
	ICLK	1	I	Clock input
	RD*	1	I	Read signal
	WR*	1	I	Write signal
	CS*	1	I	Chip select signal
	DB(15:0)	16	I/O	Data bus
	IFMD	1	I	Interface mode select pin
	ACC(1:0)	2	I	Accumulator control signal
	SMP	1	I	Simple register control signal
	MOZ(3:0)	4	I	Sets multiplier output to zero
	ODEN*	1	I	Output port OD enable pin
	TEST*	1	I	Test signal, normally 1
Input ports	PA(7:0)	8	I	Input port PA
	PB(7:0)	8	I	Input port PB
	PC(7:0)	8	I	Input port PC
	PD(7:0)	8	I	Input port PD
	PE(7:0)	8	I	Input port PE
	PF(7:0)	8	I	Input port PF
	PG(7:0)	8	I	Input port PG
	PH(7:0)	8	I	Input port PH
	CSDI(19:0)	20	I	Cascade input port
Output ports	OD(19:0)	20	O	Output port OD
	OVF	1	O	Overflow flag output
Power & Ground	Vdd	12	Power	5 V
	GND	12	Ground	0 V

Note: An asterisk (*) after a signal name denotes that signal uses negative logic (active low).

2.2 Pin Assignments

Pin No.	Pin Symbol	(I/O)
1	GND	PW
2	PA0	I
3	PA1	"
4	PA2	"
5	PA3	"
6	PA4	"
7	PA5	"
8	PA6	"
9	PA7	"
10	PB0	"
11	PB1	"
12	PB2	"
13	PB3	"
14	PB4	"
15	PB5	"
16	PB6	"
17	PB7	"
18	PC0	"
19	PC1	"
20	Vdd	PW
21	GND	PW
22	PC2	I
23	PC3	"
24	PC4	"
25	PC5	"
26	PC6	"
27	PC7	"
28	PD0	"
29	PD1	"
30	PD2	"
31	PD3	"
32	PD4	"
33	PD5	"
34	PD6	"
35	PD7	"
36	PE0	"
37	PE1	"
38	PE2	"
39	PE3	"
40	Vdd	PW

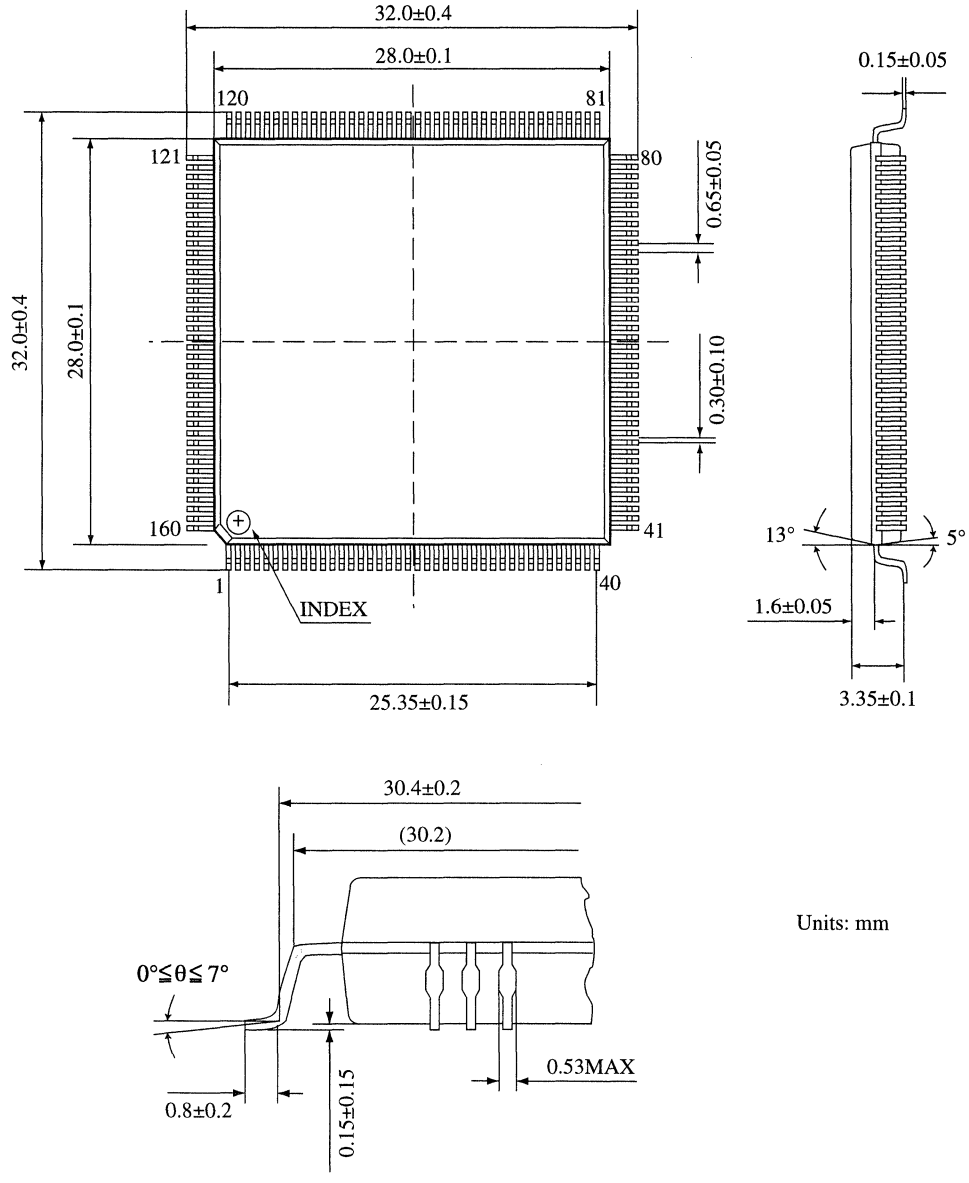
Pin No.	Pin Symbol	(I/O)
41	GND	PW
42	PE4	I
43	PE5	"
44	PE6	"
45	PE7	"
46	PF0	"
47	PF1	"
48	PF2	"
49	PF3	"
50	PF4	"
51	PF5	"
52	PF6	"
53	PF7	"
54	PG0	"
55	PG1	"
56	PG2	"
57	PG3	"
58	PG4	"
59	PG5	"
60	Vdd	PW
61	GND	PW
62	PG6	I
63	PG7	"
64	PH0	"
65	PH1	"
66	PH2	"
67	PH3	"
68	PH4	"
69	PH5	"
70	PH6	"
71	PH7	"
72	CSDI0	"
73	CSDI1	"
74	CSDI2	"
75	CSDI3	"
76	CSDI4	"
77	CSDI5	"
78	CSDI6	"
79	CSDI7	"
80	Vdd	PW

Pin No.	Pin Symbol	(I/O)
81	GND	PW
82	CSDI8	I
83	CSDI9	"
84	CSDI10	"
85	CSDI11	"
86	CSDI12	"
87	CSDI13	"
88	CSDI14	"
89	CSDI15	"
90	CSDI16	"
91	CSDI17	"
92	CSDI18	"
93	CSDI19	"
94	Vdd	PW
95	GND	PW
96	OD0	O
97	OD1	"
98	OD2	"
99	OD3	"
100	Vdd	PW
101	GND	PW
102	OD4	O
103	OD5	"
104	OD6	"
105	OD7	"
106	OD8	"
107	OD9	"
108	OD10	"
109	OD11	"
110	Vdd	PW
111	GND	PW
112	OD12	O
113	OD13	"
114	OD14	"
115	OD15	"
116	OD16	"
117	OD17	"
118	OD18	"
119	OD19	"
120	Vdd	PW

Pin No.	Pin Symbol	(I/O)
121	GND	PW
122	OVF	O
123	Vdd	PW
124	ODEN*	I
125	ACC0	"
126	ACC1	"
127	MOZ0	"
128	MOZ1	"
129	MOZ2	"
130	MOZ3	"
131	SMP	"
132	IFMD	"
133	TEST*	"
134	RST*	"
135	CS*	"
136	WR*	"
137	RD*	"
138	GND	PW
139	ICLK	I
140	Vdd	PW
141	GND	PW
142	DB0	I/O
143	DB1	"
144	DB2	"
145	DB3	"
146	DB4	"
147	DB5	"
148	DB6	"
149	DB7	"
150	Vdd	PW
151	GND	PW
152	DB8	I/O
153	DB9	"
154	DB10	"
155	DB11	"
156	DB12	"
157	DB13	"
158	DB14	"
159	DB15	"
160	Vdd	PW

Note 1: The IFMD pin is a Schmitt input.
Note 2: The TEST* pin is a Schmitt input, with pull-up resistance.
Note 3: The RST* pin is Schmitt input, with pull-up resistance.

2.3 External Dimensions



Units: mm

Section 3. Operating Description

3.1 Overview of Operations

The IP90C31 LSI chip is designed for high-speed product-sum calculations. It contains four 8-bit x 8-bit multipliers and a 40-bit accumulator, and has an optional setting that allows it to perform 16-bit x
16-bit multiplication and product-sum calculations. The chip outputs data in 16-bit floating-point format, and includes a selector that enables access to any 20 bits on the 40-bit bus. The IP90C31 also includes a setting for rounding. Multiple IP90C31 chips can be connected through the sample register block to form a cascade connection.

3.2 Multiplication

The IP90C31 contains four 8-bit x 8-bit multiplication units. These units pair input from ports PA through PH for multiplication as follows: PA x PB, PC x PD, PE x PF, and PG x PH.

When 16-bit multiplication mode is selected, PA and PB are combined as 16-bit data with PA as the upper byte, and PC and PD are likewise combined as 16-bit data with PC as the upper byte (input from PE through PH is ignored). The input data can be expressed in two's-complement or unsigned binary. After the multiplication, data is expressed in two's complement form.

3.3 Cumulative Adder-Subtractor Blocks

The adder-subtractor blocks take the input values PA x PB, PC x PD, PE x PF, and PG x PH from the multiplication block, and add or subtract in the following form according to designated parameters:

$$SUM = SUM \pm [\ \{\ (PA \times PB) \pm (PC \times PD)\} \pm \{(PE \times PF) \pm (PG \times PH)\ \}\ ]$$

In 16-bit multiplication mode, the processing is in this form:

$$SUM = SUM \pm (PAPB \times PCPD)$$

where PAPB represents 16-bit data in which PA is the upper byte and PB the lower byte, and PCPD represents 16-bit data in which PC is the upper byte and PD is the lower byte.

The resulting SUM is expressed in two's-complement form, in 40-bit width.

3.4 Sample Register

When load is enabled by the signal SMP, the SUM output by the cumulative adder-subtractor blocks is loaded into the sample register. When load is disabled, the value in the sample register is held until load is enabled again.

3.5 Cascade Connections

When the cascade connection feature is enabled, values in the sample register are added to values that are determined by taking the values of the cascade data input CSDI and applying a shift value defined by OSL parameter selections. Cascade data input can be synchronized with internally generated data by adjusting the internal delay parameter.

$$CSUM = SMPL + shift(CSDI)$$

When the cascade connection feature is disabled, CSUM = SMPL.

3.6 Floating-Point Output

When floating-point output format is used, CSUM is converted to floating-point format and output at OD15–OD0. In this case, OD19 through OD16 are set low.

3.7 Fixed-Point Output

When fixed-point output format is selected, 20 bits (selected from the 40-bit CSUM by the output selection parameter OSL) are output in fixed-point format at OD19-OD0. The output is rounded according to a rounding-cutoff parameter (RCO).

3.8 Number of Steps in Pipeline Processing

Input sampling	1 step
Multiplication	1 step
Cumulative addition - sample	1 step
Delay steps	0–3 steps
CSDI input sample	1 step
Cascade addition	1 step
Floating-point conversion	1 step (0 if fixed-point output is used)
Output	1 step
Total	6–10 steps

Note that if fixed-point format is used, output does not go through the floating-point conversion circuit, but instead passes from cascade addition directly into the output register, reducing the number of steps by one.

The number of steps is internally adjusted for direct and real-time parameters, so that as the parameters are sampled, they are simultaneously applied to the sampled data.

3.9 Overflow Flag

The overflow flag is set if an overflow occurs during cumulative or cascade addition.

- Accumulator (cumulative addition) overflow

  When an overflow occurs during cumulative addition, the overflow signal is sent when the overflow data output occurs. If the sample register is in standby, the overflow signal is held internally, and output once the sample register returns to sample status and begins to output the overflow data.

- Cascade addition overflow

  When an overflow occurs during cascade addition, the overflow signal is output at the same time as the overflow data.

The OVF output value is the logical OR of these two overflow conditions.

Section 4. Programmable Resources

4.1 The Control Register and IFMD Pin

The IP90C31 uses three types of parameters: direct, real-time, and fixed.

Direct parameters are control signals that are input directly from pins, and used to process data that is sampled simultaneously from ports PA through PH at every clock cycle.

Real-time and fixed parameters can be set using control registers through the CPU interface. However, when the IFMD pin is set to 1, pins DB15–DB0 are used for direct parameter input. In this case, the real-time parameters are sampled simultaneously with data from ports PA through PH at every clock cycle, and used to process this data. Also, when IFMD = 1, the WR* and CS* signals should be set high. Because fixed parameters are sampled at every clock cycle, they should be used to enter necessary constant parameter values. When fixed parameters have been altered, use the ACC parameter to reset the accumulator or to load valid values. When IFMD = 0, each write cycle invalidates earlier parameters, and a new set of parameters is written and becomes valid two clock cycles after the write operation is finished.

The following illustration shows the relation between each parameter, pin, and register. Because the control register is only 16 bits wide, the position of each parameter bit is the same as the position (bit) of that parameter when the IFMD pin is set to 1. After a reset, all bits in the register are set to 0.

bit 15 <---------- Control register bit position ----------> bit 0

ADC3	ADC2	ADC1	ADC0	IDC1	IDC0	CSD	OSL4	OSL3	OSL2	OSL1	OSL0	RCO	OPO	MPC1	MPC0

D15 <---------- Pin position when IFMD = 1 ----------> D0

4.2 Parameter Descriptions

MPC Multiplication mode control: 2 bits, fixed parameter

MPC1	MPC0	Operation
0	0	Two's-complement format 8-bit x 8-bit 4-multiplier mode
0	1	Positive binary format 8-bit x 8-bit 4-multiplier mode
1	0	Two's complement format 16-bit x 16-bit 1-multiplier mode
1	1	Positive binary format 16-bit x 16-bit 1-multiplier mode

OPO Output format selection: 1 bit, fixed parameter

OPO	Operation
0	Floating-point format output
1	Fixed-point format output

RCO Round/truncate selection: 1 bit, fixed parameter

RCO	Operation
0	Round
1	Truncate

OSL Output select: 5 bits, fixed parameter

This is valid when OPO = 1. On the internal bus (CSUM), the 20 bits beginning with the bit position corresponding to the values of OSL4–OSL0 are output to the OD pin. This parameter is also used with a code to shift the cascade input CSDI (20 bits), and connect it to a cascade adder. All IP90C31s used in cascade connection should be set to the same value.

Examples:
OSL = 0 ——> OD = CSUM(19:0)
OSL = 5 ——> OD = CSUM(24:5)
OSL = 20 ——> OD = CSUM(39:20)
OSL > 20 ——> Output value not assured

CSD Cascade input enable: 1 bit, fixed parameter

CSD	Operation
0	Enable cascade input
1	Disable cascade input

IDC Internal delay step control: 2 bits, fixed parameter

IDC1	IDC0	Delay
0	0	None
0	1	One clock cycle
1	0	Two clock cycles
1	1	Three clock cycles

ADC Add/subtract control: 4 bits, real-time parameter

The cumulative adder result SUM is determined as follows.

- 8-bit multiplication mode:

$$SUM = SUM + (-1)^{ADC3} [\{ (PA*PB) + (-1)^{ADC0} (PC*PD) \} + (-1)^{ADC2} \{ (PE*PF) + (-1)^{ADC1} (PG*PH) \}]$$

- 16-bit multiplication mode

$$SUM = SUM + (-1)^{ADC3} (PAPB) * (PCPD)$$

PAPB represents 16-bit data in which PA is the upper byte and PB is the lower byte. PCPD represents 16-bit data in which PC is the upper byte and PD is the lower byte.

ACC Accumulator control: 2 bits, direct parameter

ACC1	ACC0	Operation
0	0	Cumulative addition/subtraction
0	1	Save value
1	0	Load value from multiplier (add/subtract control from ADC3 is valid)
1	1	Reset (set cumulative add/subtract SUM to 0)

SMP Sample register control: 1 bit, direct parameter

SMP	Operation
0	Save sample register value
1	Load accumulator input value to sample register

MOZ Multiplier zero control: 4 bits, direct parameter

MOZn	Operation
0	Normal operation
1	Set n^{th} multiplier input to zero

MOZ0 = 1: Fix PA and PB input at zero.

MOZ1 = 1: Fix PC and PD input at zero.

MOZ2 = 1: Fix PE and PF input at zero.

MOZ3 = 1: Fix PG and PH input at zero.

In addition to controlling calculations, MOZ can be used to reduce power consumption when the IP90C31 is not in use.

4.3 Status After Reset

After reset, all control registers are set to 0. The values in the accumulator, sample register, and output registers are undetermined.

4.4 List of Parameters

Parameter	Bits	Bit location	Type	Description
MPC	2	1:0	Fixed	Multiplication mode control
OPO	1	2	Fixed	Floating/fixed-point output format
RCO	1	3	Fixed	Round/truncate selection
OSL	5	8:4	Fixed	Output select
CSD	1	9	Fixed	Cascade input enable
IDC	2	11:10	Fixed	Internal delay step control
ADC	4	15:12	Real-time	Add/subtract control
ACC	2	Direct	Direct	Accumulator control
SMP	1	Direct	Direct	Sample register control
MOZ	4	Direct	Direct	Multiplier zero control

Section 5. Electrical Characteristics

5.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Power supply voltage	Vdd	-0.3 to 6.5	V
Input voltage	Vi	-0.3 to Vdd+0.3	V
Input current	Ii	±10	mA
Output current	Io	10	mA
Operating temperature	Topt	0 to 70	°C
Storage temperature	Tstg	-10 to 80	°C

5.2 Recommended Operating Conditions (GND = 0V, Ta = 0° to 70°C)

Item	Symbol	Conditions	Min	Typ	Max	Unit
Power supply voltage	Vdd		4.75	5.0	5.25	V
High-level input voltage	Vih	TTL level	2.0		Vdd	V
Low-level input voltage	Vil	normal input	0		0.8	V
High-level input voltage	Vih	SCHMITT	2.25		Vdd	V
Low-level input voltage	Vil	input (note 1)	0		0.8	V
Input rise time	Tri	TTL level	0		100	ns
Input fall time	Tfi	normal input	0		100	ns
Input rise time	Tri	SCHMITT	0		1000	ns
Input fall time	Tfi	input (note 1)	0		1000	ns

Note 1: The three Schmitt trigger input pins are RST*, TEST*, and IFMD.

5.3 Input/Output Pin Capacity (Vdd = Vi = 0V)

Item	Symbol	Conditions	Min	Typ	Max	Unit
Input pins	Cin	f = 1 MHz		10		pF
Output pins	Cout	f = 1 MHz		10		pF
I/O pins	Cin	f = 1 MHz		10		pF

5.4 DC Characteristics (Vdd = 5V±5%, GND = 0V, Ta = 0° to 70°C)

Item		Symbol	Conditions	Min	Typ	Max	Unit
Static current consumption (note 1)		Il	Vi = Vdd or GND			200	µA
Output short current (note 2)		Ios	Vdd = Max, Vo = Vdd	15	50	130	mA
(all output and I/O pins)			Vdd = Max, Vo = 0	-5	-25	-100	
Low-level input leak current							
	Normal I/O pins	Iil	Vi = GND	-10	±1	10	µA
	Pins with pull-up resistance (note 3)	Iipl	Vi = GND	-35	-115	-350	µA
High-level input leak current (all input and I/O pins)		Iih	Vi = Vdd	-10	±1	10	µA
Low-level output voltage		Vol	Iol = 4 mA		0.2	0.4	V
High-level output voltage (note 4)		Voh	Ioh = -0.5 mA	4.5		Vdd	V
		Voh	Iol = -4 mA	2.4		Vdd	V
SCHMITT hysteresis voltage		Vsch	Vil to Vih	0.4	0.8		V

Note 1: Excluding static current consumption to pull-up and pull-down resistors.

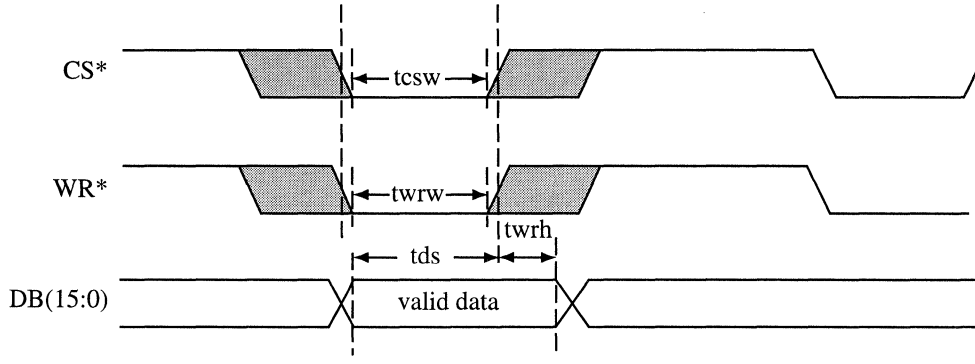
Note 2: Output short current for one second or less, at one LSI pin.

Note 3: The two pins' pull-up resistors are RST* and TEST*.

Note 4: Output is CMOS level (TTL level).

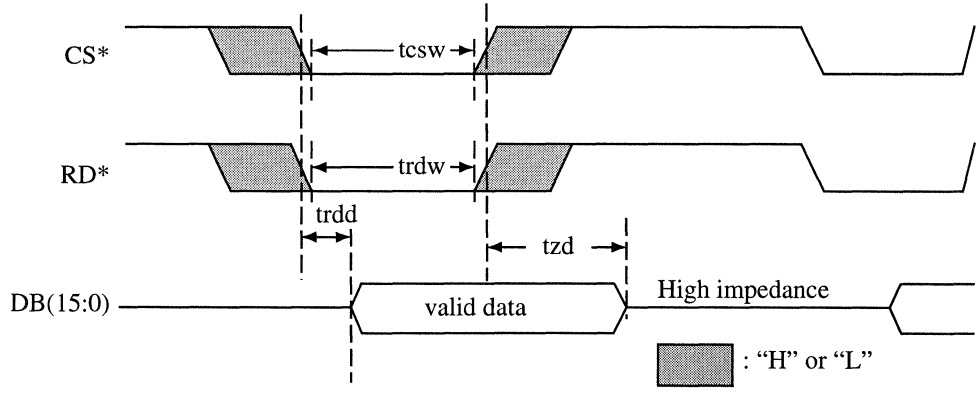
5.5 AC Characteristics

Write cycle:



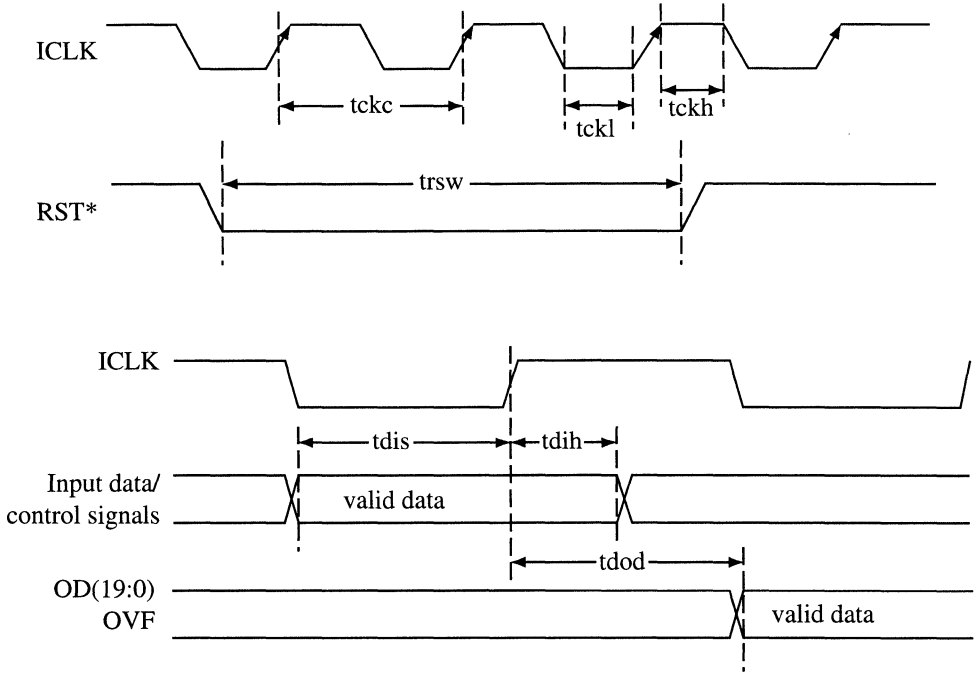
Note: tds and twrh are determined by the rising edge of CS* or WR*, whichever is faster. Writing occurs when both CS* and WR* are low.

Read cycle:



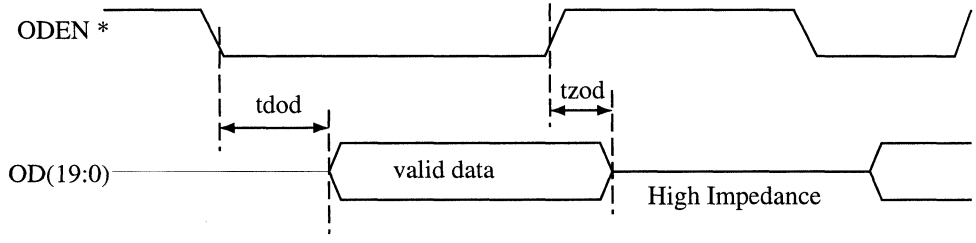
Note: trdd is determined by the falling edge of CS* or RD*, whichever is slower. tzd is determined by the rising edge of CS* or RD*, whichever is faster. Reading occurs when both CS* and RD* are low.

2) Clock Reset Timing



Ports related to input data/control signal when ACC, SMP, MOZ, IFMD = 1: DB(15:0), PA, PB, PC, PD, PE, PF, PG, PH, CSDI

3) Control/Data Signal Timing



Timing Table

Load capacity for all output pins is 30 pF.

Unit: ns

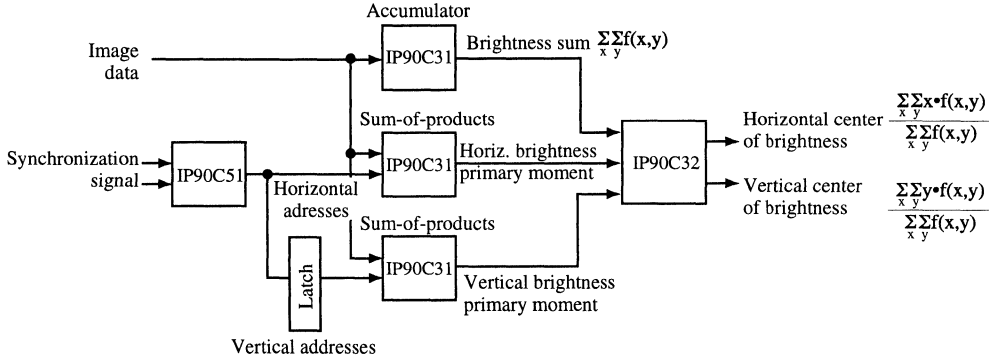
Item	Symbol	Min	Typ	Max
CS* pulse width	tcsw	20		
WR* pulse width	twrw	20		
Data setup time	tds	15		
Data hold time	twrh	3		
RD* pulse width	trdw	20		
Data delay	trdd			20
Data delay to HiZ	tzd			15
Clock cycle time	tckc	33		
Clock high time	tckh	13		
Clock low time	tckl	13		
Input/control signal setup time	tdis	8		
Input/control signal hold time	tdih	3		
Output delay	tdod	3		20
Output delay to HiZ	tzod			15

Section 6. Sample Applications

The IP90C31 and IP90C32 are designed to perform arithmetic calculations, and can therefore be used in a wide variety of fields. This section introduces one possible application, calculating the center of an image, which involves normalized correlation calculations.

6.1 Center of Brightness

Used in combination, the IP90C31, IP90C32, and IP90C51 can calculate the center of brightness of an image using hardware functions alone, including the IP90C31's 40-bit accumulator and the IP90C32's floating-point calculation functions.



About Centers of Brightness

A two-dimensional image has a quantitative measurement of brightness at any point (x, y), which can be represented by f(x, y). From this can be calculated a primary moment and center of gravity, referred to respectively as the primary moment of brightness and the center of gravity of brightness.

The coordinates of the center of gravity of brightness can be expressed as the horizontal primary moment of brightness and the vertical primary moment of brightness divided by the sum of brightness values (the 0-order moment of brightness) , thus:

$$G_x = m_{10} / m_{00}$$

$$G_y = m_{01} / m_{00}$$

Sum of Brightness Values

The sum of brightness values, which can also be thought of as the 0-order moment of brightness, are expressed by the following formula:

$$M_{00} = m_{00} = \sum_x \sum_y f(x,y)$$

where f(x, y) is the brightness (grayscale value) of the pixel at coordinates (x, y).

Primary Moment of Brightness

The primary moment of brightness is defined by the following equations. The IP90C51 provides horizontal and vertical coordinates and grayscale values. The IP90C31 then calculates the sum of the products of these coordinates and grayscale values, and calculates the primary moment of brightness around the origin.

$$m_{01} = \sum_x \sum_y y \bullet f(x,y)$$

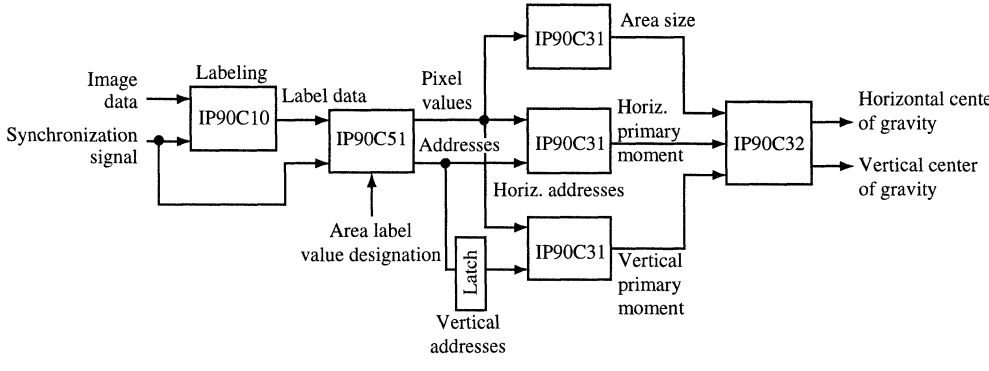
$$m_{10} = \sum_x \sum_y x \bullet f(x,y)$$

Using these measurements and the sum of brightness values, the IP90C32 determines the center of gravity of brightness by division.

6.2 Center of a Designated Labeling Area

The IP90C31, IP90C32, and IP90C10 (or IP90C11) in combination can determine the center of a designated labeling area using hardware functions alone.

The IP90C10 labels incoming binary image data and passes it to the IP90C51, which uses ternary functions to extract an area designated by label numbers. The data for this area then goes to the center-of-gravity calculation circuits of the IP90C31 and IP90C32. The center-of-gravity circuits in this example function in the same way as in the center of brightness example above, except that they handle binary data.

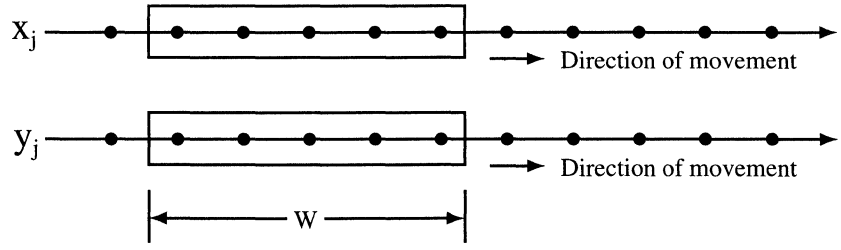


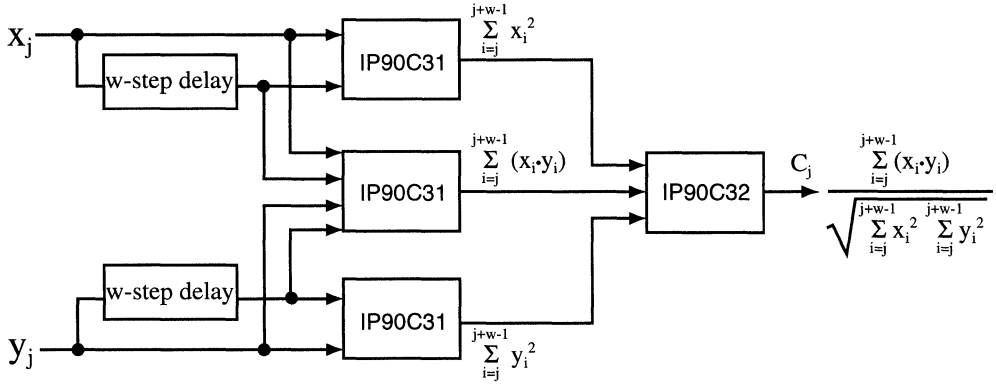
6.3 Moving Normalized Correlation Calculation

The formula below can be used to provide real-time calculation of moving normalized correlation, without subtracting average values. This type of calculation is used in a wide variety of fields, including measuring signal delays with distance measurement equipment, finding pitch in harmonic analysis, detecting synchronization in communications, and reducing noise.

$$C_j = \frac{\sum_{i=j}^{j+w-1} (x_i \cdot y_i)}{\sqrt{\sum_{i=j}^{j+w-1} x_i^2 \sum_{i=j}^{j+w-1} y_i^2}}$$

Where *w* is the width of the area (window) for normalized processing



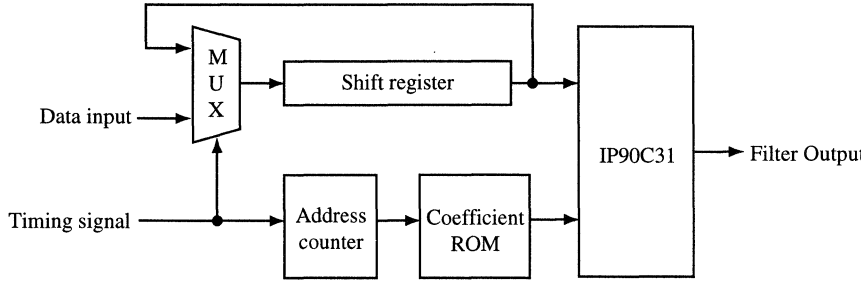


6.4 Audio FIR Filter

The IP90C31's high speed and wide processing capability make it well suited for use in high-order audio FIR filters. The general FIR filter function is expressed by the following formula:

$$y[n] = \sum_{k=0}^{N} h[k] \cdot x[n-k]$$

The various tap coefficients h[k] are written to coefficient ROM, and the past N cycles of audio data, x[n–k], are stored in the shift register. The IP90C31 multipliers are used in time multiplex scheme, and the tap coefficients and past sampling values from the shift register are taken as input to a sum-of-products calculation. Because each sum-of-products calculation must be completed before new audio data is sampled, theoretically the filter has a maximum level of taps (in 16-bit, 1-multiplier mode) equal to the ratio between the IP90C31's operating frequency and the sampling frequency of the audio signal. If 8-bit, 4-multiplier mode can be used, the above ratio is extended by a factor of four, which is effective for simulations. Also, the IP90C31's ability to use cascade connections makes it easy to link several IP90C31s for still higher order filtering.



SMI ASSP

Image Processing LSI Series

IP90C32

Configurable Arithmetic Operator (CAROL)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.5

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1. Overview

1.1 Functional Overview and Characteristics

The IP90C32 LSI calculator chip is designed to perform a variety of calculations on floating-point data. It does this using internal bus switching and on-chip functions that include three multipliers, and inverse value, square root, and inverse square root calculators. The IP90C32 also includes an adder to sum computation results, and external connections for registers and shift registers for cumulative addition. A shifter is placed at the readout for results of cumulative addition.

The IP90C32 is designed to be used with the IP90C31 chip for high-speed execution of normalized correlation calculations.

- Input floating-point format — 1-bit code, 6-bit exponent, 9-bit mantissa
- Internal floating-point format — 1-bit code, 8-bit exponent, 9-bit mantissa
- Types of calculation — Multiplication, division, square root, etc.
- Output — Fixed or floating-point format
- Maximum operating frequency — 30 MHz
- Power supply — 5-V single source
- Process — CMOS
- I/O interface — TTL-level compatible
- Package — 160-pin plastic QFP

1.2 Floating Decimal Point Format

The IP90C32 accepts data in the following format (the same as for the IP90C31):

- Total width 16 bits (bit 15–bit 0).

  bit 15: 1 bit code bit S 0 = pos, 1 = neg
  bit 14–bit 9: 6 bits exponent e 0–63

  The actual exponent value is e–b, where b is the bias value.

  bit 8–bit 0: 9 bits mantissa m

  Positive binary values (offset binary) indicate values after the decimal point, and do not include integer components.

- The integer component is determined by the exponent.

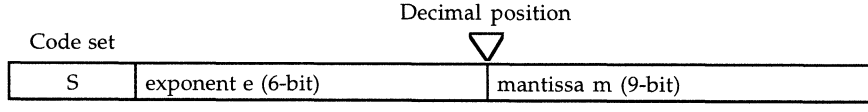
  $e \neq 0$ —> integer component I = 1
  $e = 0$ —> integer component I = 0

- Zero (0) is expressed as e = 0 and m = 0.

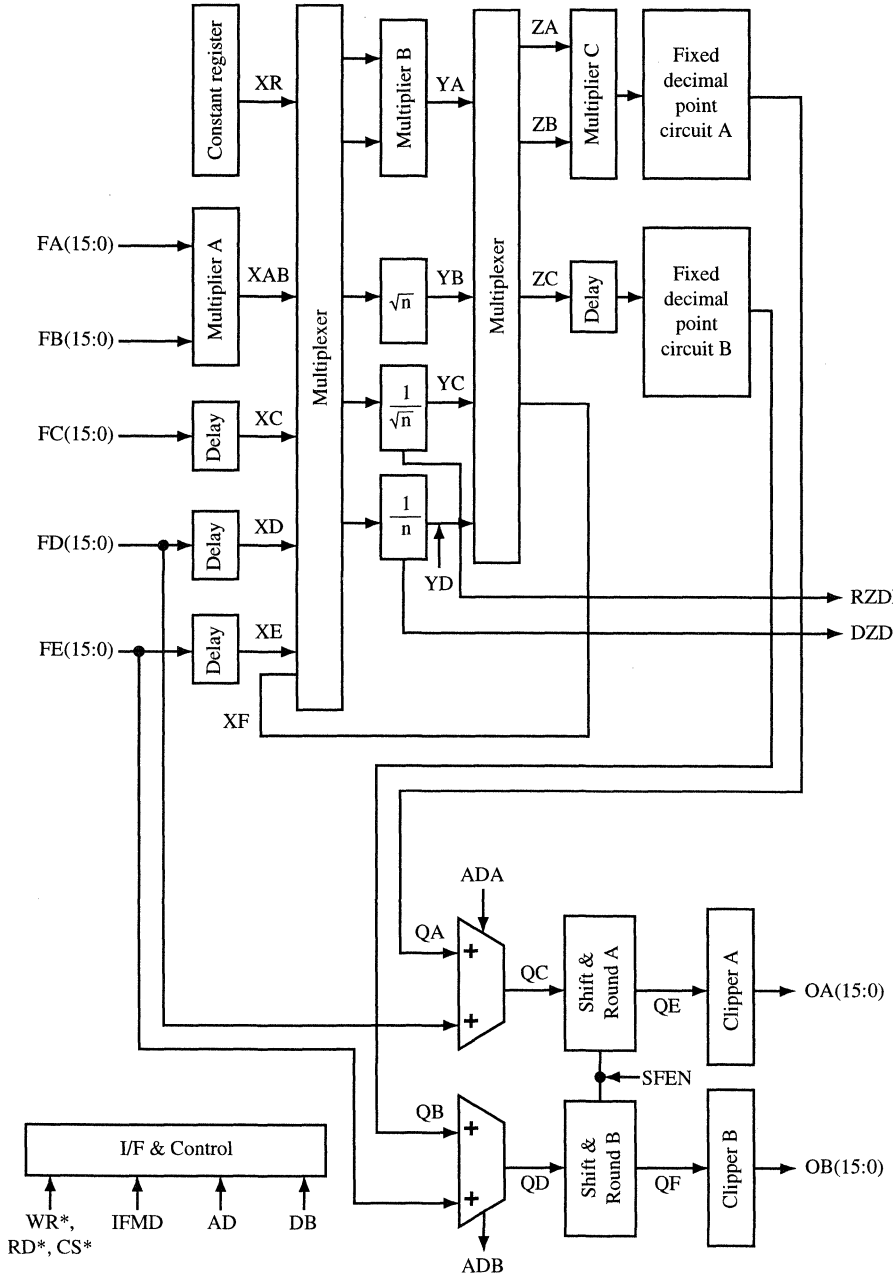
  Non-normal data (de-normalized values) are not handled, so that values in which e = 0 are always processed as m = 0.

- In equation form, $(-1)^3 \times I.m \times 2^{(e-b)}$.

  In the IP90C32 chip, the bias value b can be set to 1 or 32.



1.3 Block Diagram



Section 2. Pins and Functions

2.1 Pin Lists and Functional Descriptions

Signal group	Pin symbol	No.	Type	Function
Control signals	RST*	1	I	Reset
	ICLK	1	I	Clock input
	RD*	1	I	Read signal
	WR*	1	I	Write signal
	CS*	1	I	Chip select signal
	DB(3:0)	4	I/O	Data bus
	AD(4:0)	5	I/O	Address bus
	IFMD	1	I	Interface mode select pin
	ADA(1:0)	2	I	Adder A control signal
	ADB(1:0)	2	I	Adder B control signal
	SFEN	1	I	Shift enable signal
	TEST*	1	I	Test signal, normally 1
Input ports	FA(15:0)	16	I	Input port FA
	FB(15:0)	16	I	Input port FB
	FC(15:0)	16	I	Input port FC
	FD(15:0)	16	I	Input port FD
	FE(15:0)	16	I	Input port FE
Output ports	OA(15:0)	16	O	Output port OA
	OB(15:0)	16	O	Output port OB
	DZGF	1	O	Zero division flag
	RZDF	1	O	Inverse root zero division flag
Power &	Vdd	12	Power	5 V
Ground	GND	13	Ground	0 V

2.2 Pin Assignments

Pin No.	Pin Symbol	(I/O)
1	GND	PW
2	FA0	I
3	FA1	"
4	FA2	"
5	FA3	"
6	FA4	"
7	FA5	"
8	FA6	"
9	FA7	"
10	FA8	"
11	FA9	"
12	FA10	"
13	FA11	"
14	FA12	"
15	FA13	"
16	FA14	"
17	FA15	"
18	FB0	"
19	FB1	"
20	Vdd	PW
21	GND	PW
22	FB2	I
23	FB3	"
24	FB4	"
25	FB5	"
26	FB6	"
27	FB7	"
28	FB8	"
29	FB9	"
30	FB10	"
31	FB11	"
32	FB12	"
33	FB13	"
34	FB14	"
35	FB15	"
36	FC0	"
37	FC1	"
38	FC2	"
39	FC3	"
40	Vdd	PW

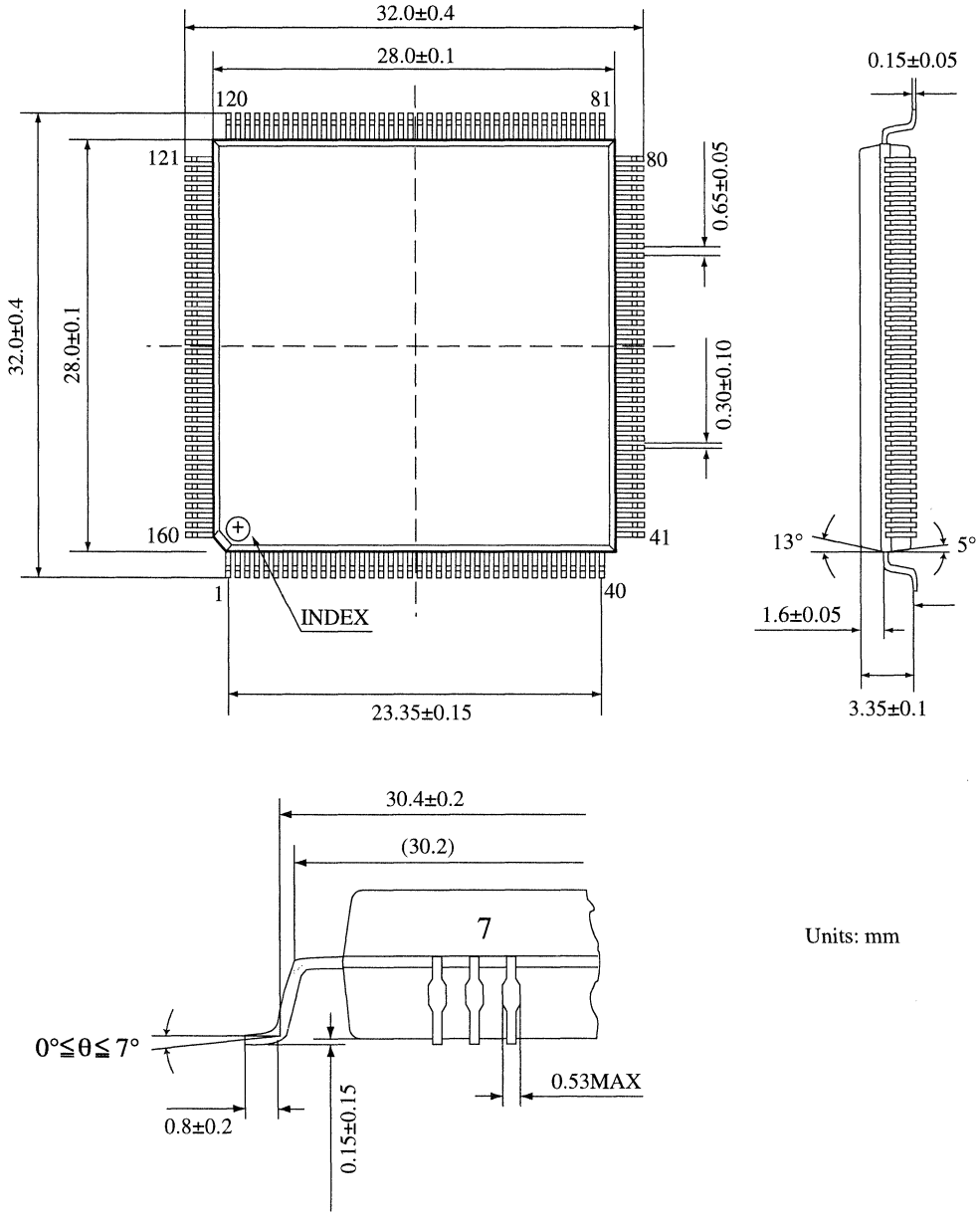
Pin No.	Pin Symbol	(I/O)
41	GND	PW
42	FC4	I
43	FC5	"
44	FC6	"
45	FC7	"
46	FC8	"
47	FC9	"
48	FC10	"
49	FC11	"
50	FC12	"
51	FC13	"
52	FC14	"
53	FC15	"
54	FD0	"
55	FD1	"
56	FD2	"
57	FD3	"
58	FD4	"
59	FD5	"
60	Vdd	PW
61	GND	PW
62	FD6	I
63	FD7	"
64	FD8	"
65	FD9	"
66	FD10	"
67	FD11	"
68	FD12	"
69	FD13	"
70	FD14	"
71	FD15	"
72	FE0	"
73	FE1	"
74	FE2	"
75	FE3	"
76	FE4	"
77	FE5	"
78	FE6	"
79	FE7	"
80	Vdd	PW

Pin No.	Pin Symbol	(I/O)
81	GND	PW
82	FE8	I
83	FE9	"
84	FE10	"
85	FE11	"
86	FE12	"
87	FE13	"
88	FE14	"
89	FE15	"
90	Vdd	PW
91	GND	PW
92	OA0	O
93	OA1	"
94	OA2	"
95	OA3	"
96	OA4	"
97	OA5	"
98	OA6	"
99	OA7	"
100	Vdd	PW
101	GND	PW
102	OA8	O
103	OA9	"
104	OA10	"
105	OA11	"
106	OA12	"
107	OA13	"
108	OA14	"
109	OA15	"
110	Vdd	PW
111	GND	PW
112	OB0	O
113	OB1	"
114	OB2	"
115	OB3	"
116	OB4	"
117	OB5	"
118	OB6	"
119	OB7	"
120	Vdd	PW

	Pin Symbol	(I/O)
121	GND	PW
122	OB8	O
123	OB9	"
124	OB10	"
125	OB11	"
126	OB12	"
127	OB13	"
128	OB14	"
129	OB15	"
130	RZDF	"
131	DZDF	"
132	GND	PW
133	Vdd	PW
134	SFEN	I
135	ADA0	"
136	ADA1	"
137	ADB0	"
138	ADB1	"
139	GND	PW
140	ICLK	I
141	Vdd	PW
142	TEST*	I
143	RST*	"
144	IFMD	"
145	CS*	"
146	WR*	"
147	RD*	"
148	GND	PW
149	Vdd	PW
150	AD0	I/O
151	AD1	"
152	AD2	"
153	AD3	"
154	AD4	"
155	GND	PW
156	DB0	I/O
157	DB1	"
158	DB2	"
159	DB3	"
160	Vdd	PW

Note 1: The TEST* pin is Schmitt input, with pull-up resistance.
Note 2: The RST* pin is Schmitt input, with pull-up resistance.
Note 3: The IFMD pin is Schmitt input.

2.3 External Dimensions



Units: mm

Section 3. Operating Description

3.1 Overview of Operation

The IP90C32 LSI calculator chip is designed to perform a variety of calculations on floating-point data. It does this using internal bus switching and on-chip functions that include three multipliers, and inverse value, square root, and inverse square root calculators. The IP90C32 also includes an adder to sum computation results, and external connections for registers and shift registers for cumulative addition. A shifter is placed at the readout for results of cumulative addition.

3.2 Input Format

The IP90C32 accepts input in the floating-point format described earlier, with a bias value of 1 or 32. Note that data input to the FD or FE port is in fixed-point notation.

3.3 Input Data

Input data passes through format conversion, in which the exponent is expanded to 8 bits to enhance the dynamic range for multiplication and division. This conversion enables the IP90C32 to perform calculations with a dynamic range from 2^{-126} to 2^{126}.

3.4 Pipeline Synchronization

The IP90C32 can be set up to perform a wide variety of calculations using pipeline processing. Therefore, the resulting data must be synchronized. To do this, two clock count programmable delay steps are provided for input data at ports FC, FD, and FE, and at the input side of fixed-point circuit B.

3.5 Constant Register

The IP90C32 has a 1 data-unit (16 bits) constant register. Data in this register is written in floating-point format. The bias is selected by parameter setting.

3.6 Main Calculator

The IP90C32's main calculator includes multiplier B, an inverse value calculator, a square-root calculator, and an inverse square-root calculator.

3.7 Main Calculator Input/Output Optional Circuits

In addition to multiplier A and multiplier C, optional circuits are provided at both the input and output sides of the main calculator. These circuits can be switched and combined as needed to perform a wide variety of calculations.

3.8 Fixed-Point Circuits

Calculation results are output through the fixed-point circuits, with the decimal point's position controlled by a parameter setting. Results can be output in floating-point format; however, the output is limited to 16-bit width, so that the result must be within the limits of 16-bit fixed-point format. The output format selection parameters allow a choice of two's-complement or positive binary format. If an overflow condition in fixed-point format occurs at this point, the output value is clipped. Clipping is applied to the mantissa portion in floating-point format, which is then converted to two's-complement format, so that the smallest negative value will be one greater than the least value expressible in two's-complement format.

3.9 Addition Circuits

Calculation results can be output directly. In addition, the IP90C32's design also allows results to be added to data input from ports FE or FF. (See the calculator connections shown in the block diagram in Section 1.3.) In this case, the data input at ports FE or FF must have the same fixed-point format as the data output from the fixed-point circuits.

3.10 Shift and Rounding Circuits

The calculation results described above are sent to the shift and rounding circuits. The maximum right-shift is 4 bits. Data in two's-complement format is shifted arithmetically, and data in positive binary format is shifted logically. The shift and rounding functions round results to multiples of 2, 4, 8, or 16 places.

3.11 Output Clipping

Overflow output from adding or rounding operations can be clipped. Clipping is applied when the IP90C32 detects an overflow before executing a shift in the shift circuit.

For example, if an addition creates the result FFFFh in positive binary format, a 4-bit shift would produce the result 0FFFh. This value would normally be rounded to the next highest digit (1000h), but enabling the clipping function keeps the result at 0FFFh. This is because when a 4-bit reverse shift is applied to 1000h, the result is outside the original 16-bit range. Alternatively, if the 4-bit shift is applied to the sum 8000 + 8000, an overflow occurs in the sum before shifting, so the output is written as 0FFFh.

In two's-complement mode, the highest bit is the code bit, so that positive or negative overflow conditions are detected at the 15th bit. If a 4-bit shift is applied to the sum 7FFFh + 0001h, the result with the clipping function enabled is 07FFh. If the 4-bit shift is applied to the sum 7000h + 0FFFh, rounding normally produces 0800h. However, enabling the clipping function keeps the result at 07FFh, because applying a 4-bit shift to 0800h before clipping would give 0800h and cause an overflow.

3.12 Number of Steps in Pipeline Processing

The following shows the number of steps in each circuit.

Stage	Steps	
Floating-point input sampling	1 step	
Floating-point internal format conversion	1 step	
Multiplier A	1 step	
Multiplier B	1 step	
Multiplier C	1 step	
Inverse value calculator	1 step	
Square-root calculator	1 step	
Inverse square-root calculator	1 step	
Fixed-point circuit	1 step	
Addition, shift, and rounding	1 step	
Output	1 step	
FD, FE addition input sampling	1 step	(... connected to add, shift and rounding circuits)

The addition control and shift enable signals simultaneously control the addition and shift of data input from ports FD and FE.

3.13 Zero Division Flag

The inverse value calculator and inverse square-root calculators are equipped with zero division flags. The flag signal output is timed to the same clock count as is the placement of these calculator's results in their respective registers. When a division by zero occurs, the result is zero.

3.14 Power-Saving Mode

The IP90C32 is equipped with a power-saving mode, which reduces power consumption when no calculations are required. In this mode, data from each input port FA through FE is internally reset to zero, thus reducing power consumption.

Section 4. Programmable Resources

4.1 IFMD Pin

The IFMD pin signal is set to 0 for normal CPU interface functions. When this signal is 1, the IP90C32 automatically generates address AD(4:0) for sampling data input at DB(3:0). Parameters can be set without depending on the CPU by decoding the address through PLD or another external resource, generating the necessary parameters for that address and returning them to DB(3:0). This can only be done once per reset. Each parameter is valid for two clock cycles after it is written.

4.2 Signal Names Used in Descriptions

The following table lists the signals used in parameter descriptions. See also the block diagram in Section 1.3.

Signal	Description
XR	Constant register output
XAB	Multiplier A output
XC	Input from FC through a delay circuit
XD	Input from FD through a delay circuit
XE	Input from FE through a delay circuit
XF	Return data from calculator output
YA	Multiplier B output
YB	Square-root calculator output
YC	Inverse square-root calculator output
YD	Reciprocal calculator output
ZA, ZB	Multiplier C input
ZC	Fixed-point conversion circuit B input
QA	Fixed-point conversion circuit A output
QB	Fixed-point conversion circuit B output
QC	Addition circuit A output
QD	Addition circuit B output
QE	Shift & round circuit A output
QF	Shift & round circuit B output

4.3 Parameter Descriptions

■ DLYC(1:0): FC input delay step selection parameter
Address 0 bit 1:0

bit 1	bit 0	Delay
0	0	None
0	1	1 clock cycle
1	x	2 clock cycles

■ DLYD(1:0) FD input delay step selection parameter
Address 0 bit 3:2

bit 3	bit 2	Delay
0	0	None
0	1	1 clock cycle
1	x	2 clock cycles

■ DLYE(1:0) FE input delay step selection parameter
Address 1 bit 1:0

bit 1	bit 0	Delay
0	0	None
0	1	1 clock cycle
1	x	2 clock cycles

■ DLYF(1:0) Fixed-point conversion circuit B input delay selection
Address 1 bit 3:2

bit 3	bit 2	Delay
0	0	None
0	1	1 clock cycle
1	x	2 clock cycles

■ SMPYB1(2:0) Multiplier B input selection 1
Address 2 bit 2:0

bit 2	bit 1	bit 0	Signal selected
0	0	0	XR
0	0	1	XAB
0	1	0	XC
0	1	1	XD
1	0	0	XE
1	0	1	XF
1	1	x	Prohibited

■ SMPYB2(2:0) Multiplier B input selection 2
Address 3 bit 2:0

bit 2	bit 1	bit 0	Signal selected
0	0	0	XR
0	0	1	XAB
0	1	0	XC
0	1	1	XD
1	0	0	XE
1	0	1	XF
1	1	x	Prohibited

■ SRT(2:0) Square-root calculator input selection
Address 4 bit 2:0

bit 2	bit 1	bit 0	Signal selected
0	0	0	XR
0	0	1	XAB
0	1	0	XC
0	1	1	XD
1	0	0	XE
1	0	1	XF
1	1	x	Prohibited

■ SIRT(2:0) Inverse square-root calculator input selection
Address 5 bit 2:0

bit 2	bit 1	bit 0	Signal selected
0	0	0	XR
0	0	1	XAB
0	1	0	XC
0	1	1	XD
1	0	0	XE
1	0	1	XF
1	1	x	Prohibited

■ SINV(2:0) Inverse calculator input selection
Address 6 bit 2:0

bit 2	bit 1	bit 0	Signal selected
0	0	0	XR
0	0	1	XAB
0	1	0	XC
0	1	1	XD
1	0	0	XE
1	0	1	XF
1	1	x	Prohibited

■ SMPYC1(1:0) Multiplier C input selection 1
Address 7 bit 1:0

bit 1	bit 0	Signal selected
0	0	YA
0	1	YB
1	0	YC
1	1	YD

■ SMPYC2(1:0) Multiplier C input selection 2
Address 7 bit 3:2

bit 3	bit 2	Signal selected
0	0	YA
0	1	YB
1	0	YC
1	1	YD

■ SFPB(1:0) Fixed decimal conversion circuit B input selection
Address 8 bit 1:0

bit 1	bit 0	Signal selected
0	0	YA
0	1	YB
1	0	YC
1	1	YD

- SFDB(1:0) Return data selection
  Address 8 bit 3:2

bit 3	bit 2	Signal selected
0	0	YA
0	1	YB
1	0	YC
1	1	YD

- FXPAH Fixed-point conversion circuit A decimal-point position (7:4)
  Address 9 bit 3:0
- FXPAL Fixed-point conversion circuit A decimal-point position (3:0)
  Address A bit 3:0

The location of the 1's position in positive binary mode is designated by setting the highest bit of the output from the fixed-point conversion circuit to 0, and the second highest bit to 1, in two's-complement format. In two's-complement mode, the highest bit of the output is a code bit, so that the second highest bit is set to 0 and the third highest bit to 1. Alternatively, by designating the number -128, the output can be left in floating-point format. In this case the bias value is the same as the FBS floating-point bias setting.

- FXPBH Fixed-point conversion circuit B decimal-point position (7:4)
  Address B bit 3:0
- FXPBL Fixed-point conversion circuit B decimal-point position (3:0)
  Address C bit 3:0

The location of the 1's position in positive binary mode is designated by setting the highest bit of the output from the fixed decimal conversion circuit to 0, and the second highest bit to 1, in two's-complement format. In two's-complement mode, the highest bit of the output is a code bit, so that the second highest bit is set to 0 and the third highest bit to 1. Alternatively, by designating the number -128, the output can be left in floating-point format. In this case the bias value is the same as the FBS floating-point bias setting.

Note: The mantissa is expanded internally for use in calculations. However, the mantissa of floating-point output is limited to 6 bits. Be aware that no overflow control is applied.

- SFTA(2:0) Addition circuit A shift
  Address D bit 2:0

  This parameter designates the degree of shift applied by shift circuit A. SFTA must be in the range 0 to 4.

- SFTB(2:0) Addition circuit B shift
  Address E bit 2:0

  This parameter designates the degree of shift applied by shift circuit B. SFTB must be in the range 0 to 4.

- FXMDA Fixed-point conversion circuit A operating mode
  Address F bit 0
  FXMDA = 0 ——> two's-complement output
  FXMDA = 1 ——> positive binary output

- FXMDB Fixed-point conversion circuit B operating mode
  Address F bit 1
  FXMDB = 0 ——> two's-complement output
  FXMDB = 1 ——> positive binary output

- FBS Floating-point bias
  Address F bit 2
  FBS = 0 ——> bias value = 32
  FBS = 1 ——> bias value = 1

  This parameter affects only the floating-point format for input and output from the IP90C32 chip, and has no effect on fixed-point notation or on internal calculations.

- COA Round/truncate option A
  Address 10 bit 0
  Valid only when shift is enabled.
  COA = 0 ——> round
  COA = 1 ——> truncate

- COB Round/truncate option B
  Address 10 bit 1
  Valid only when shift is enabled.
  COB = 0 ——> round
  COB = 1 ——> truncate

- OACRPEN Output clip A enable
  Address 10 bit 2
  OACRPEN = 0 ——> disable
  OACRPEN = 1 ——> enable

- OBCRPEN Output clip B enable
  Address 10 bit 3
  OBCRPEN = 0 —> disable
  OBCRPEN = 1 —> enable

- CNSTHH Constant register 15:12
  Address 11 bit 3:0
  This parameter writes to bit 15:12 of the constant register.

- CNSTHL Constant register 11:8
  Address 12 bit 3:0
  This parameter writes to bit 11:8 of the constant register.

- CNSTLH Constant register 7:4
  Address 13 bit 3:0
  This parameter writes to bit 7:4 of the constant register.

- CNSTLL Constant register 3:0
  Address 14 bit 3:0
  This parameter writes to bit 3:0 of the constant register.

- PSV Power saving mode select 3:0
  Address 15 bit 3:0
  PSVn = 0 —> normal operation
  PSVn = 1 —> power saving mode: set input to 0

- Bit position n corresponds to the following ports.
  n = 0: FA, FB input
  n = 1: FC input
  n = 2: FD input
  n = 3: FE input

- VBW Variable bus width enable
  Address 16 bit 0
  VBW = 0 —> disable
  VBW = 1 —> enable

- Reserved
  Address 16 bit 3:1
  This parameter should always be set to 0.

■ ADA(1:0) Addition circuit A control

Pin direct

ADA1	ADA0	Operation
0	0	Addition operation.
0	1	Output fixed-point conversion circuit output with no changes.
1	0	Output FD input with no changes.
1	1	Output 0 (zero).

■ ADB(1:0) Addition circuit B control

Pin direct

ADB1	ADB0	Operation
0	0	Addition operation.
0	1	Output fixed-point conversion circuit output with no changes.
1	0	Output FE input with no changes.
1	1	Output 0 (zero).

■ SFEN Shift enable

Pin direct

SFEN = 0 —> shift disable

SFEN = 1 —> shift enable

4.4 Default Values

After a reset, all parameter values are zero.

4.5 List of Parameters

Address	Bit(s)	Symbol	Description
00	1:0	DLYC	FC input delay step selection parameter
00	3:2	DLYD	FD input delay step selection parameter
01	1:0	DLYE	FE input delay step selection parameter
01	3:2	DLYF	Fixed-point conversion circuit B input delay selection
02	2:0	SMPYB1	Multiplier B input selection 1
03	2:0	SMPYB2	Multiplier B input selection 2
04	2:0	SRT	Square-root calculator input selection
05	2:0	SIRT	Inverse square-root calculator input selection
06	2:0	SINV	Inverse calculator input selection
07	1:0	SMPYC1	Multiplier C input selection 1
07	3:2	SMPYC2	Multiplier C input selection 2
08	1:0	SFPB	Fixed-point conversion circuit B input selection
08	3:2	SFDB	Return data selection
09	3:0	FXPAH	Fixed-point conversion circuit A decimal-point position 7:4
0A	3:0	FXPAL	Fixed-point conversion circuit A decimal-point position 3:0
0B	3:0	FXPBH	Fixed-point conversion circuit B decimal-point position 7:4
0C	3:0	FXPBL	Fixed-point conversion circuit B decimal-point position 3:0
0D	2:0	SFTA	Addition circuit A shift
0E	2:0	SFTB	Addition circuit B shift
0F	0	FXMDA	Fixed-point conversion circuit A operating mode
0F	1	FXMDB	Fixed-point conversion circuit B operating mode
0F	2	FBS	Floating-point bias
10	0	COA	Round/truncate option A
10	1	COB	Round/truncate option B
10	2	OACRPEN	Output clip A enable
10	3	OBCRPEN	Output clip B enable
11	3:0	CNSTHH	Constant register 15:12
12	3:0	CNSTHL	Constant register 11:8
13	3:0	CNSTLH	Constant register 7:4
14	3:0	CNSTLL	Constant register 3:0
15	3:0	PSV	Power-saving mode select
16	0	VBW	Variable bus width enable
16	3:1	Reserved	Set to 0

Section 5. Variable Bit-Width Mode

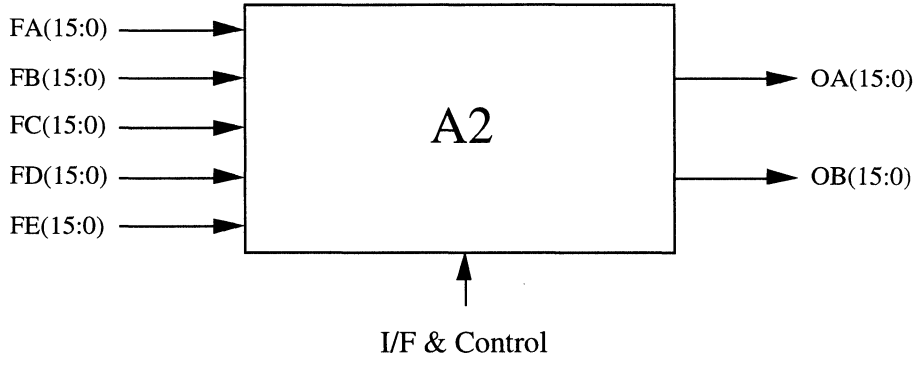
The IP90C32 chip is equipped with a variable bit-width mode that allows variable use of output ports OA and OB and input ports FD and FE. For simplification, the following description uses the designations VOA and VOB for the output ports, and VFD and VFE for the input ports.

5.1 Pin Correspondence

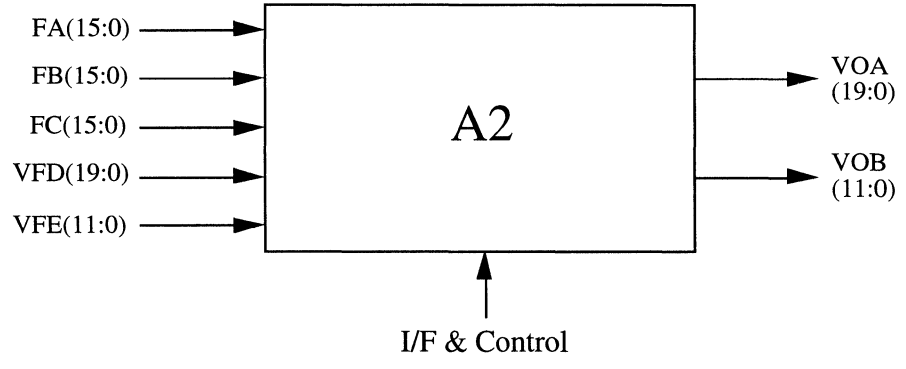
VOA and VFD are 20 bits wide, and VOB and VFE are 12 bits wide. In normal mode they correspond to pins as follows:

- VOA(15:0) = OA(15:0), VOA(19:16) = OB(15:12)
- VOB(11:0) = OB(11:0)
- VFD(15:0) = FD(15:0), VFD(19:16) = FE(15:12)
- VFE(11:0) = FE(11:0)

Normal Operation



Variable Bit-Width Mode



5.2 Internal Hardware

5.2.1 Fixed-Point Conversion Circuits

The fixed-point conversion circuits work the same way in normal and variable bit-width modes. Output is always in 16 bits, and clipping is always limited to 16-bit width.

5.2.2 Addition Circuits

In normal mode, addition circuits A and B operate in 16 bits. In variable bit-width mode, however, addition circuit VOA basically operates in 20 bits. Addition circuit VOB operates in 16 bits, though its output is immediately clipped to 12 bits. All shift, round, and clip circuits connected to the addition circuits must correspond to the appropriate bit widths. The addition and fixed-point conversion circuits are aligned with respect to the lowest bit value.

5.2.3 Operation During Bit Width Changes

In variable bit-width mode, the bit width of the input to the addition circuits may differ from the bit width actually assigned. The procedure used to compensate depends on whether the operation uses two's-complement or positive binary data, and is designated by the fixed decimal conversion circuit operating mode selection parameter.

In power-saving mode, parameters PSV2 and PSV3 use the variable bit-width ports VFD and VFE.

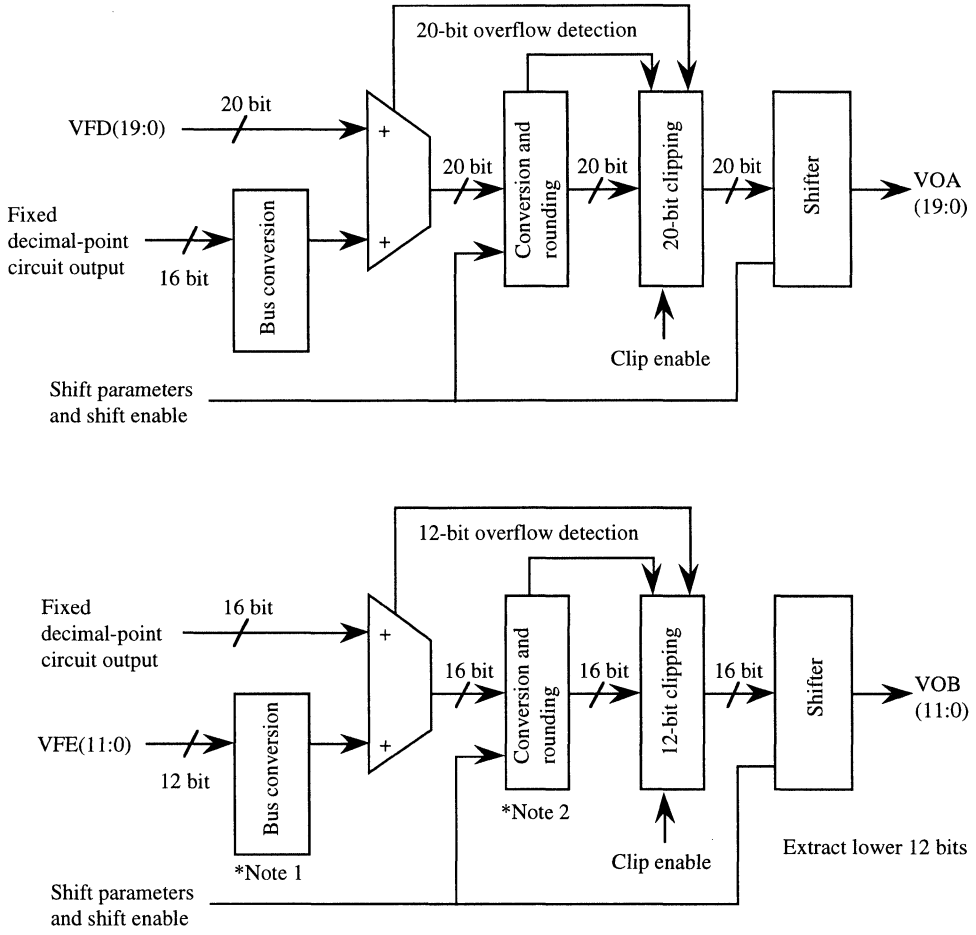
5.2.4 Clipping Operations

Clipping circuits are affected along with the addition circuits (see above), in that the overflow detection bit count is changed from 16 bits to 20 bits for addition circuit VA, and to 12 bits for addition circuit B.

For example, assume a 20-bit bus width from fixed-point conversion circuit A to addition circuit A to output port VOA. A positive binary output from the addition circuit of FFFFFh becomes 0FFFF after a 4-bit shift. Rounding-up adds one more digit: if clipping is disabled, the outcome is 10000h, but if clipping is enabled the result is 0FFFFh (because when a 4-bit reverse shift is applied, the value 10000 exceeds the original 20-bit range). Also, if a 4-bit shift is applied to the value F8000+08000, the addition before the shift causes an overflow, so that the output becomes 0FFFFh.

In two's-complement mode, the upper bit is a code bit, so that positive/negative overflows are determined at the 19th bit. For example, if a 4-bit shift is applied to 7FFFFh+00001h, the result with clipping enabled is 07FFFh. Also, if a 4-bit shift is applied to 70000h+0FFFFh, rounding-up yields 08000h, but if clipping is enabled the result is 07FFFh (because the result before clipping is 08000h, which with a 4-bit left shift becomes 80000h, an overflow value).

Overview of Output in Variable Bit-Width Mode



Note 1: "Bus conversion" refers to matching input bit width to addition circuit input bit width.

Note 2: "Variable rounding" refers to rounding with respect to the size of the shift applied. For a shift of n bits (with n=1 as the LSB), a value of 1 is added to the n^{th} bit.

Section 6. Electrical Characteristics

6.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Power supply voltage	Vdd	-0.3 to 6.5	V
Input voltage	Vi	-0.3 to Vdd+0.3	V
Input current	Ii	±10	mA
Output current	Io	10	mA
Operating temperature	Topt	0 to 70	°C
Storage temperature	Tstg	-10 to 80	°C

6.2 Recommended Operating Conditions (GND = 0V, Ta = 0° to 70°C)

Item	Symbol	Conditions	Min	Typ	Max	Unit
Power supply voltage	Vdd		4.75	5.0	5.25	V
High-level input voltage	Vih	TTL level	2.0		Vdd	V
Low-level input voltage	Vil	normal input	0		0.8	V
High-level input voltage	Vih	SCHMITT	2.25		Vdd	V
Low-level input voltage	Vil	input (note 1)	0		0.8	V
Input rise time	Tri	TTL level	0		100	ns
Input fall time	Tfi	normal input	0		100	ns
Input rise time	Tri	SCHMITT	0		1000	ns
Input fall time	Tfi	input (note 1)	0		1000	ns

Note 1: The three Schmitt trigger input pins are RST*, TEST*, and IFMD.

6.3 Input/Output Pin Capacity (Vdd = Vi = 0V)

Item	Symbol	Conditions	Min	Typ	Max	Unit
Input pins	Cin	f = 1 MHz		10		pF
Output pins	Cout	f = 1 MHz		10		pF
I/O pins	Cin	f = 1 MHz		10		pF

6.4 DC Characteristics (Vdd = 5V±5%, GND = 0V, Ta = 0° to 70°C)

Item		Symbol	Conditions	Min	Typ	Max	Unit
Static current consumption (note 1)		Il	Vi = Vdd or GND			200	µA
Output short current (note 2)		Ios	Vdd = Max, Vo = Vdd	15	50	130	mA
(all output and I/O pins)			Vdd = Max, Vo = 0	-5	-25	-100	
Low-level input leak current							
	Normal I/O pins	Iil	Vi = GND	-10	±1	10	µA
	Pins with pull-up resistance (note 3)	Iipl	Vi = GND	-35	-115	-350	µA
High-level input leak current (all input and I/O pins)		Iih	Vi = Vdd	-10	±1	10	µA
Low-level output voltage		Vol	Iol = 4 mA		0.2	0.4	V
High-level output voltage		Voh	Ioh = -0.5 mA	4.5		Vdd	V
(notes 4 and 5)		Voh	Iol = -4 mA	2.4		Vdd	V
SCHMITT hysteresis voltage		Vsch	Vil to Vih	0.4	0.8		V

Note 1: Excluding static current consumption to pull-up and pull-down resistors.

Note 2: Output short current for one second or less, at one LSI pin.

Note 3: The two pins pull-up resistors are RST* and TEST*.

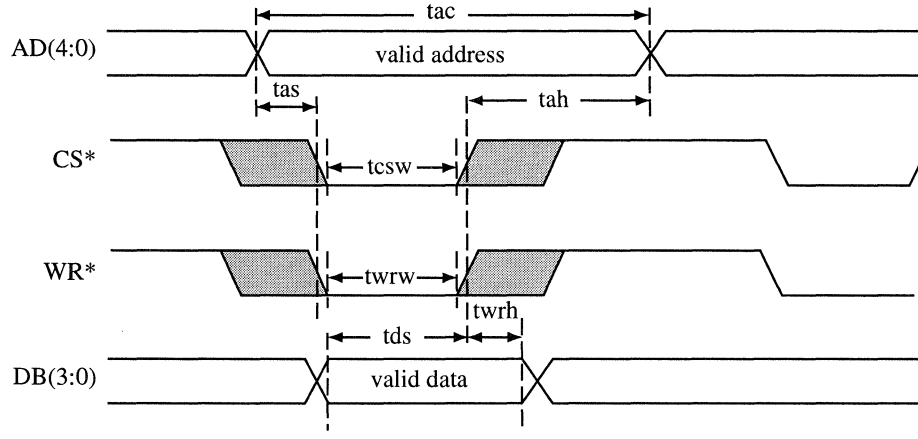
Note 4: Output is CMOS level (TTL level).

Note 5: Maximum external load capacity: less than 25 MHz: 50 pF
25 MHz or greater: 30 pF

6.5 AC Characteristics

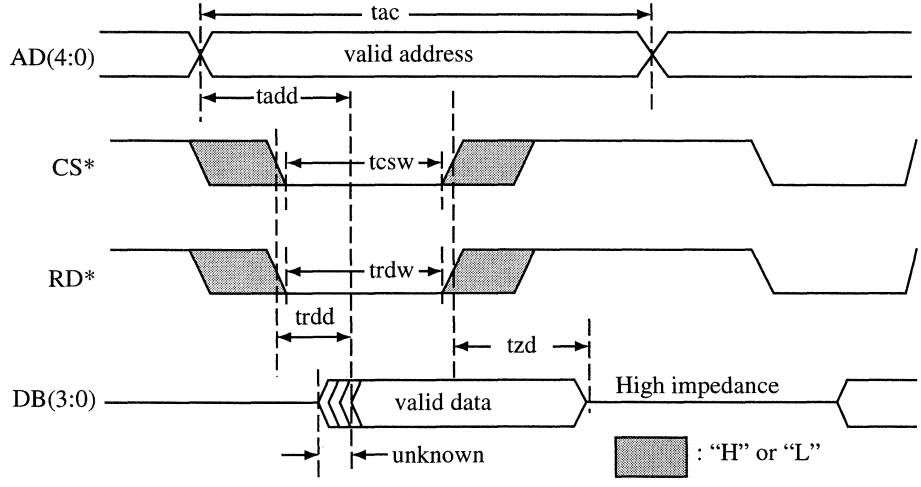
1) CPU Interface Timing

Write cycle:



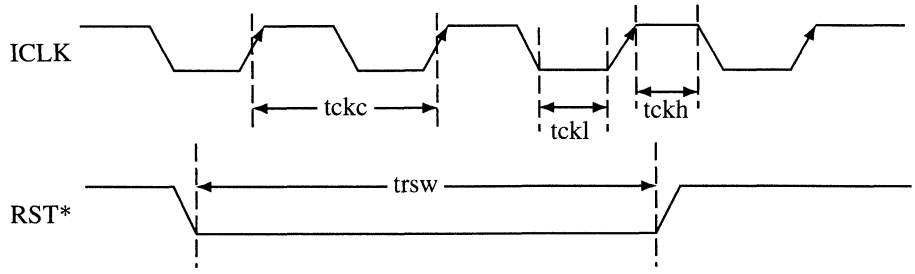
Note: twrh is determined by the rising edge of CS* or WR*, whichever is faster. Writing occurs when CS* and WR* are both low.

Read cycle:



Note: trdd is determined by the falling edge of CS* or RD*, whichever is slower. tzd is determined by the rising edge of CS* or RD*, whichever is faster. Reading occurs when CS* and RD* are both at low level.

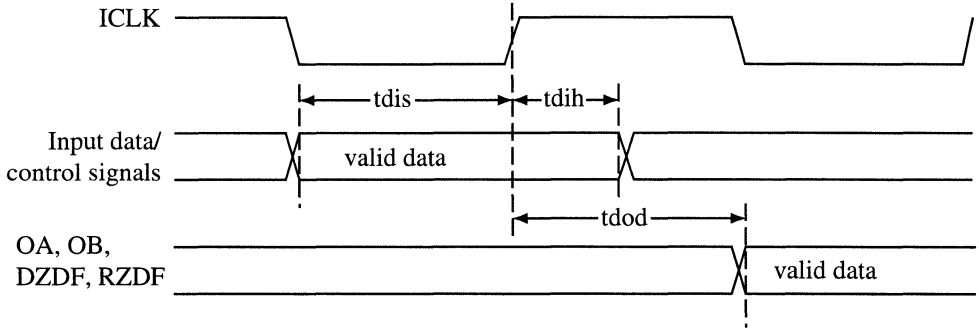
2) Clock Reset Timing



The minimum reset signal input period is four clock cycles.

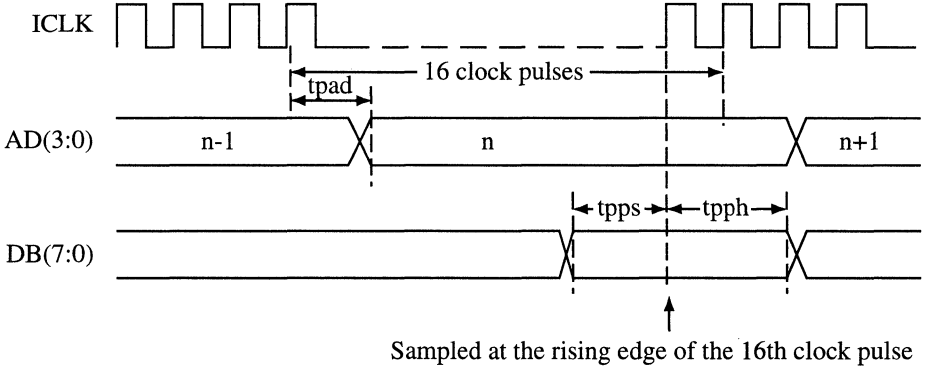
3) Control / Data Signal Timing

Input data/control signal related.



ADA, ABD, SFEN, FA, FB, FC, FD, FE

4) Timing when IFMD = 1



Timing Table

Load capacity for all output pins is 30 pF.

Unit: ns

Item	Symbol	Min	Typ	Max
Write cycle time	tac	60		
AD(4:0) setup time	tas	15		
AD(4:0) hold time	tah	3		
CS* pulse width	tcsw	20		
WR* pulse width	twrw	20		
Data setup time	tds	15		
Data hold time	twrh	3		
RD* pulse width	trdw	20		
Data delay	trdd			20
Data delay from address	tadd			25
Data delay to HIZ	tzd			15
Clock cycle time	tckc	33		
Clock high time	tckh	13		
Clock low time	tckl	13		
Input/control signal setup time	tdis	8		
Input/control signal hold time	tdih	3		
Output data delay	tdod	3		20
IFMD = 1 address delay	tpad			30
IFMD = 1 data setup time	tpps	30		
IFMD = 1 data hold time	tpph	10		

Load Capacity for Output Pins OA, OB

The IP90C32 has 32 output pins (including the OA and OB pins) that operate at high speed. The load capacities of these pins can lead to problems with ground balance and EMI. Therefore, use output pins OA and OB with loads of no more than 50 pF. Also, using large loads greatly increases power consumption and raises transistor junction temperatures. Calculating from the highest allowable transistor junction temperature, at operating frequencies of 25 MHz and higher, the load capacities of OA and OB should be no more than 30 pF. (The IP90C32's maximum allowable output load is 30 pF at 30-MHz operating frequency, 5.25-volt power supply, and ambient temperature of 70°C.)

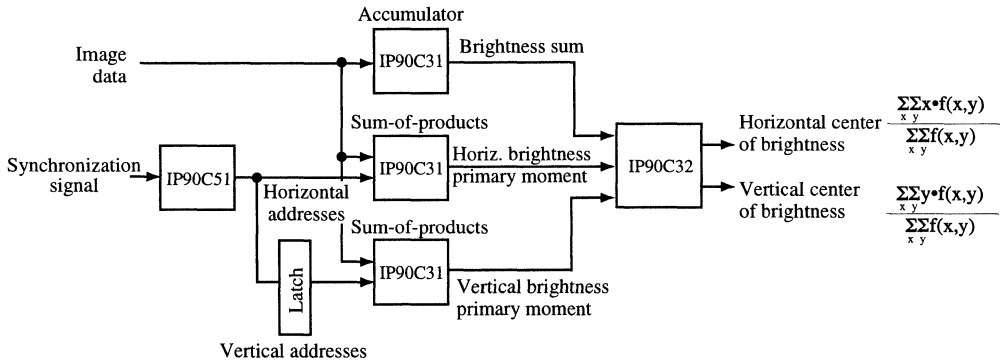
Power consumption can be reduced by connecting circuits not needed for internal calculations to circuits whose signals do not vary (such as the constant register). If heat generation appears to affect peripheral circuits, reduce power consumption to the lowest possible level.

Section 7. Sample Applications

The IP90C31 and IP90C32 are designed to perform arithmetic calculations, and can therefore be used in a wide variety of fields. This section introduces one possible application, calculating the center of an image, which involves normalized correlation calculations.

7.1 Center of Brightness

The IP90C31, IP90C32, and IP90C51 can combine to calculate the center of brightness of an image using hardware functions alone. Using the IP90C31's 40-bit accumulator, and the IP90C32's floating-point calculation functions, this chip set has the wide range of calculation capacity needed to perform this calculation.



About Centers of Brightness

A two-dimensional image has a quantitative measurement of brightness at any point (x, y), which can be represented by f(x, y). From this can be calculated a primary moment and center of gravity, referred to respectively as the primary moment of brightness and the center of gravity of brightness.

The coordinates of the center of gravity of brightness can be expressed as the sum of brightness values (the 0-order moment of brightness) divided by the horizontal primary moment of brightness and the vertical primary moment of brightness, thus:

$$G_x = m_{10} / m_{00}$$

$$G_y = m_{01} / m_{00}$$

Sum of Brightness Values

The sum of brightness values, which can also be thought of as the 0-order moment of brightness, is expressed by the following formula:

$$M_{00} = m_{00} = \sum_x \sum_y f(x,y)$$

where f(x, y) is the brightness (grayscale value) of the pixel at coordinates (x, y).

Primary Moment of Brightness

The primary moment of brightness is defined by the following equations. The IP90C51 chip provides horizontal and vertical coordinates and grayscale values. The IP90C31 LSI chip calculates the sum of the products of these coordinates and grayscale values, and calculates the primary moment of brightness around the origin.

$$m_{01} = \sum_x \sum_y y \bullet f(x,y)$$

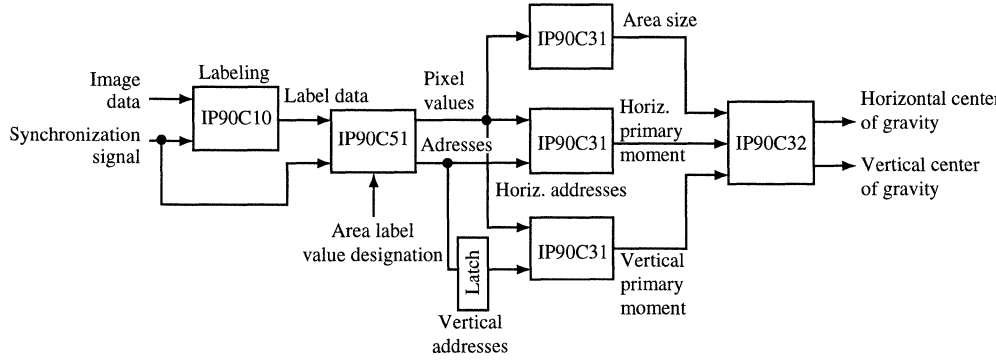
$$m_{10} = \sum_x \sum_y x \bullet f(x,y)$$

Using these measurements and the sum of brightness values, the IP90C32 determines the center of gravity of brightness by division.

7.2 Center of a Designated Labeling Area

The IP90C31, IP90C32, and IP90C10 (or IP90C11) LSI chips in combination can determine the center of a designated labeling area by hardware functions alone.

The IP90C10 labels incoming binary image data, which then passes to the IP90C51, which uses ternary functions to extract an area designated by label numbers. This data then goes to the center-of-gravity calculation circuits of the IP90C31 and IP90C32. The center-of-gravity circuits in this example function in the same way as in the center of brightness example above, except that they handle binary data.

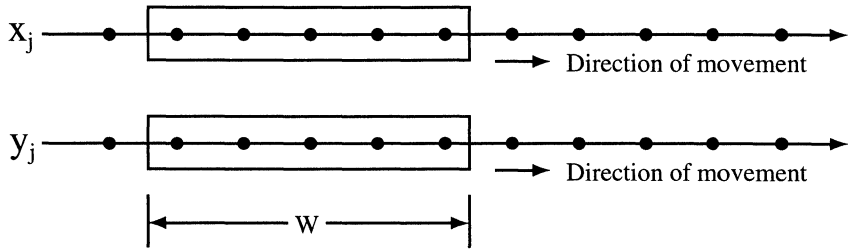


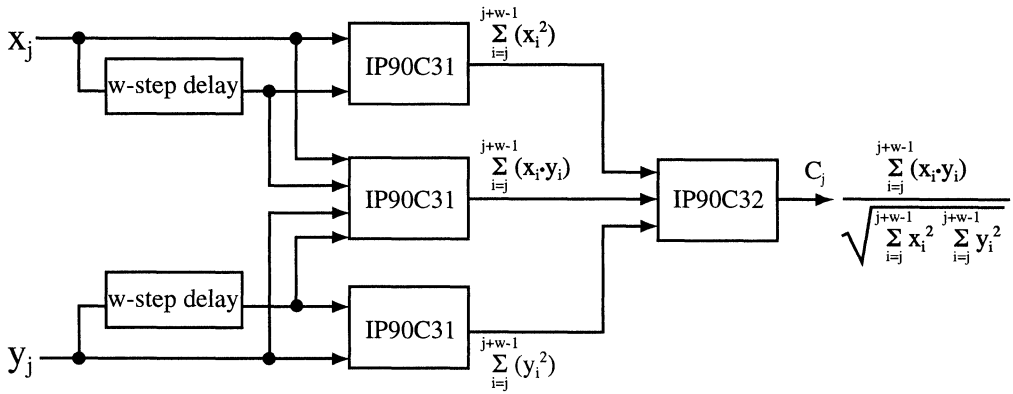
7.3 Moving Normalized Correlation Calculation

The formula below can be used to provide real-time calculation of moving normalized correlation, without subtracting average values. This type of calculation is used in a wide variety of fields, including measuring signal delays with distance measurement equipment, finding pitch in harmonic analysis, detecting synchronization in communications, and reducing noise.

$$C_j = \frac{\sum_{i=j}^{j+w-1} (x_i \cdot y_i)}{\sqrt{\sum_{i=j}^{j+w-1} x_i^2 \sum_{i=j}^{j+w-1} y_i^2}}$$

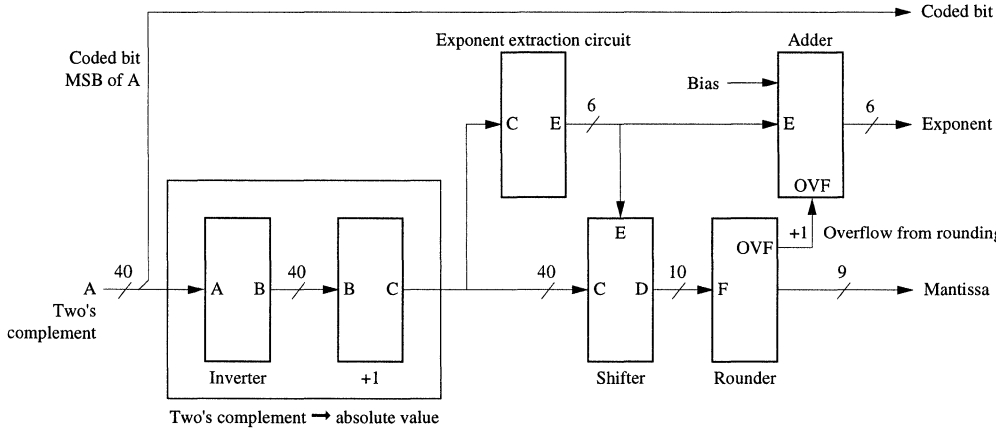
Where w is the width of the area (window) for normalized processing





7.4 Floating-Point Conversion Circuit

This section introduces an overview of circuits used to externally generate floating point format data for input to the IP90C32 chip.



Operating Description

Step 1: Extract all coded bits.

The MSB of data in two's-complement format is the coded bit.

Step 2: Convert data from two's-complement format to absolute value.

If the coded bit is zero, use the data as it is. If the coded bit is 1, convert the data to absolute value using this process:

1. Invert all data bits: B = A (Example: A = 010101, B = 101010).
2. Add 1 to B: C = B + 1.

Step 3: Extract the exponent (E).

The exponent is determined by the position of the highest 1 in the absolute value (C).

(Examples)

Absolute value:	0100010001	Exponent = 8
Absolute value:	0010010001	Exponent = 7
Absolute value:	0000000001	Exponent = 0

Step 4: Extract the mantissa.

The mantissa is the nine bits following the position of the highest 1 in the absolute value (C).

(Examples)

Absolute value:	001000010100100000111
Shifter output D:	0000101001
Mantissa:	000010101 (rounded)

Absolute value:	00000000000000010111
Shifter output D:	0111000000
Mantissa:	011100000 (rounded)

If the shifter output D consists entirely of 1s, the carry value generated by the round function is transferred to the MSB, 1 is added to the exponent, and the mantissa is set to 0.

Step 5: Add the bias value.

The bias value is added to the exponent. The IP90C32 can switch between bias values of 1 and 32, using a register setting.

Note: Zero expressions are a special case. A circuit is added to handle input that consists of all zeros, and produces an output of all zeros.

SMI ASSP

Image Processing LSI Series

IP90C51

Image Data Bus Controller (IMBC)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.2

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Overview

1.1 Functional Overview and Features

The IP90C51 is a programmable image data bus controller (IMBC) that controls high-speed transmission and processing of image data. It is compatible with all image transfer standards for standard television and high-definition television (HDTV).

The IP90C51 incorporates a digital image data bus interface and basic image data processing functions into a single chip, and is ideal for image data blanking with NTSC-standard image signals.

Features

- Uses standard horizontal and vertical synchronization pulses.
- Supports 8-bit, 256-level grayscale data.
- Supports 24-bit full-color data when used with three IP90C51 chips.
- Supports high-speed, high-density parallel processing for HDTV.
- Generates output frame memory addresses for high-speed memory access.
- Supports both interlaced and non-interlaced scanning.
- Interfaces directly with all image processor ASSPs made by Sumitomo Metal Industries, Ltd.
- Can process image areas as large as 4095 x 4095 pixels, enough to handle large images, or high-resolution images from scanners or HDTV systems.

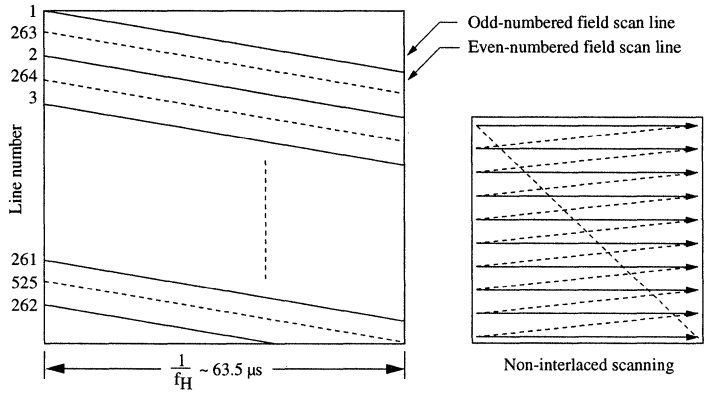
Functions

- Digital image data bus interface controller
  - Programmable image field clipping (area of interest [AOI] function)
  - Frame memory address generation
- Image data processing
  - Bi-level/window compared bi-level output
  - Ternary conversion function
  - Data enable control for each pixel
  - Background processing

Section 2: Image Data Configuration

2.1 Sample Input Image

NTSC Image Data



NTSC Scanning Line (interlaced)

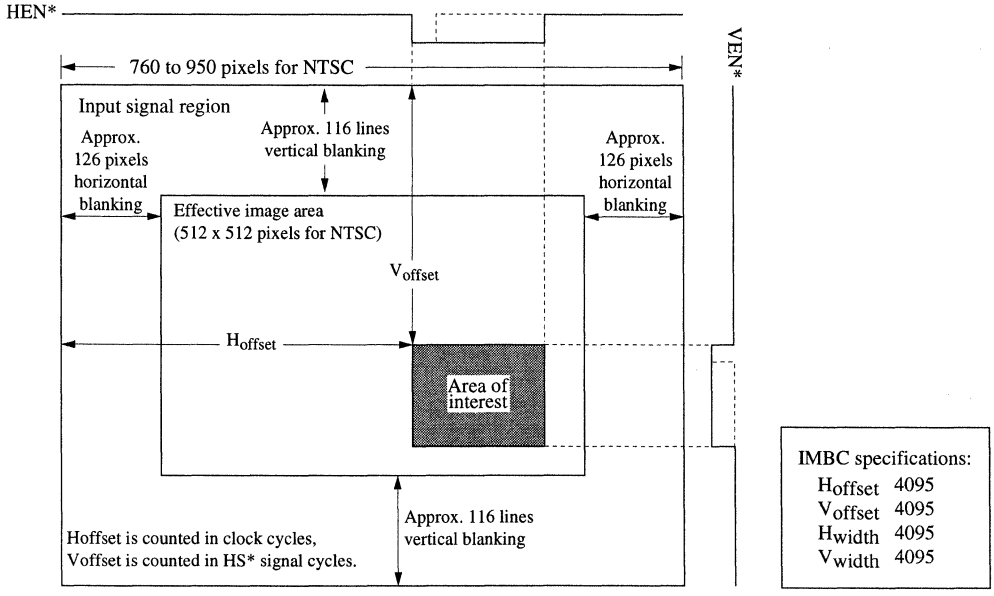


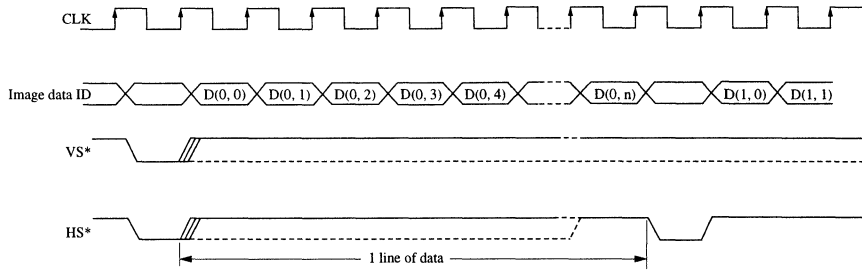
Image data enters as an NTSC signal, undergoes A/D conversion, and becomes the input signal area transferred over the image bus. In the above illustration, the effective image area is the 512 x 512-pixel rectangular area shown in the example above. Image processing applications commonly extract and process only a portion of an image, such as the area of interest (AOI) shown above. The IP90C51 allows the user to define the AOI within the effective image area of the incoming image signal.

2.2 Image Data Configuration

1) Pixel Data Configuration

D(0,0)	D(0,1)	D(0,2)	D(0,3)		D(0,n)
D(1,0)	D(1,1)	D(1,2)	D(1,3)		D(1,n)
D(2,0)	D(2,1)	D(2,2)	D(2,3)		D(2,n)
D(m,0)	D(m,1)	D(m,2)	D(m,3)		D(m,n)

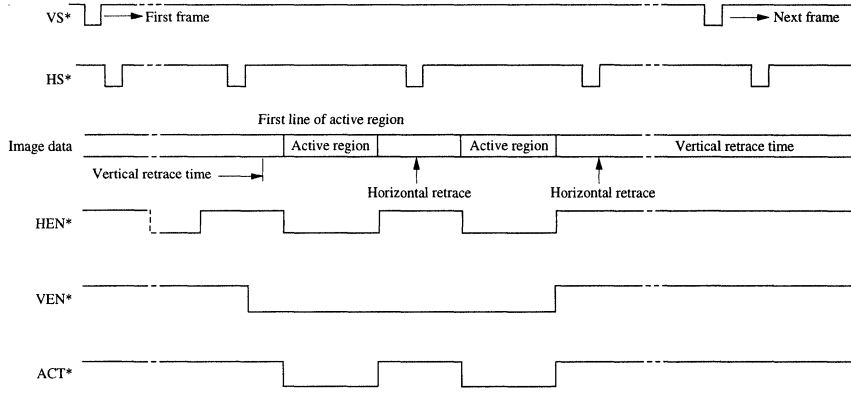
2) Image Data Format for Transmission: Single Line



Note 1: The VS* and HS* signal pulses need not enter the IP90C51 simultaneously. HS* pulses preceding a VS* pulse are ignored, while HS* signals that enter simultaneously with or following the VS* pulse are recognized.

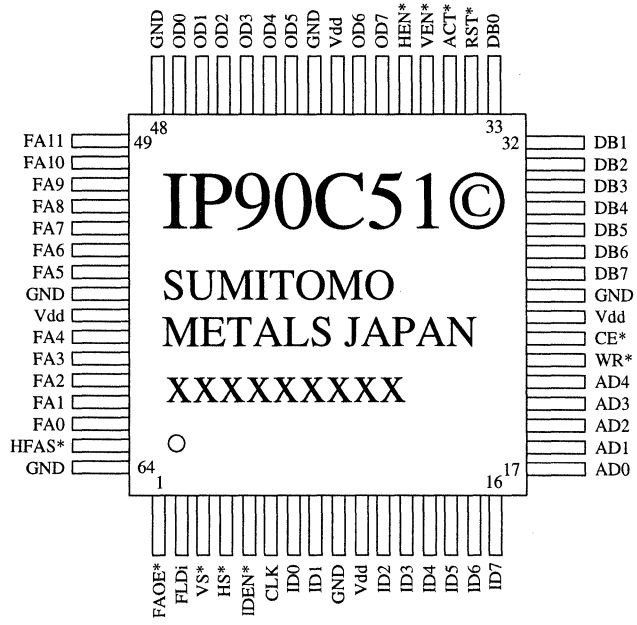
Note 2: The VS* and HS* signals do not need to be input as pulses, because they are differentiated internally by clock synchronized circuits. VS* or HS* input is assumed whenever a low-level VS* or HS* is detected at the rising edge of the clock signal. VS* or HS* must go high again before the next VS* or HS* assertion (low level).

3) Effective Area Along the Time Axis



Section 3: Pin Descriptions

3.1 Pin Placement



64-Pin QFP Package (molded area: 14 mm^2, pin pitch: 0.8 mm)

Pin Assignments

Pin	Name	Pin	Name	Pin	Name	Pin	Name
1	FAOE*	17	AD0	33	DB0	49	FA11
2	FLDi	18	AD1	34	RST*	50	FA10
3	VS*	19	AD2	35	ACT*	51	FA9
4	HS*	20	AD3	36	VEN*	52	FA8
5	IDEN*	21	AD4	37	HEN*	53	FA7
6	CLK	22	WR*	38	OD7	54	FA6
7	ID0	23	CE*	39	OD6	55	FA5
8	ID1	24	Vdd	40	Vdd	56	GND
9	GND	25	GND	41	GND	57	Vdd
10	Vdd	26	DB7	42	OD5	58	FA4
11	ID2	27	DB6	43	OD4	59	FA3
12	ID3	28	DB5	44	OD3	60	FA2
13	ID4	29	DB4	45	OD2	61	FA1
14	ID5	30	DB3	46	OD1	62	FA0
15	ID6	31	DB2	47	OD0	63	HFAS*
16	ID7	32	DB1	48	GND	64	GND

Note: An asterisk (*) after the signal name indicates inverse logic (active low).

3.2 Pins and Their Functions

Pin group	Symbol	No. of Pins	Type	Description
Image input bus	CLK	1	I	Clock input f_{max} = 36 MHz
	ID0–ID7	8	I	8-bit image data input
	HS*	1	I	Line strobe input
	VS*	1	I	Frame strobe input
	IDEN*	1	I	Direct enable input: activates internal counter when low (Note 1).
	FLDi	1	I	Input field discriminator signal input: high for odd-numbered fields, low for even-numbered fields. Set high for non-interlaced scanning.
Image output bus	OD0–OD7	8	O	8-bit image data output
	HEN*	1	O	Horizontal data enable output
	VEN*	1	O	Vertical data enable output
	ACT*	1	O	Active area signal or output field discriminator signal output: function is specified in register SMOD.
Image address	FA0–FA11	12	O	Frame memory address outputs. HCNT's value is output when HEN* is low. The upper-field address is output when HEN* is high (Note 2).
	HFAS*	1	O	Upper-field address strobe output
	FAOE*	1	I	Image address output enable input. Enabled when low. FA0–FA11 are high impedance when FAOE* is high.
CPU bus	WR*	1	I	Write enable input
	CE*	1	I	Chip enable input
	AD0–AD4	5	I	Register select address bus inputs
	DB0–DB7	8	I	Data bus inputs
	RST*	1	I	Reset input. Resets on pulse widths of 3 clock cycles or greater (Note 3).
Power supply	Vdd	4	PW	5V
GND	GND	6	PW	Ground
Total number of pins		64		

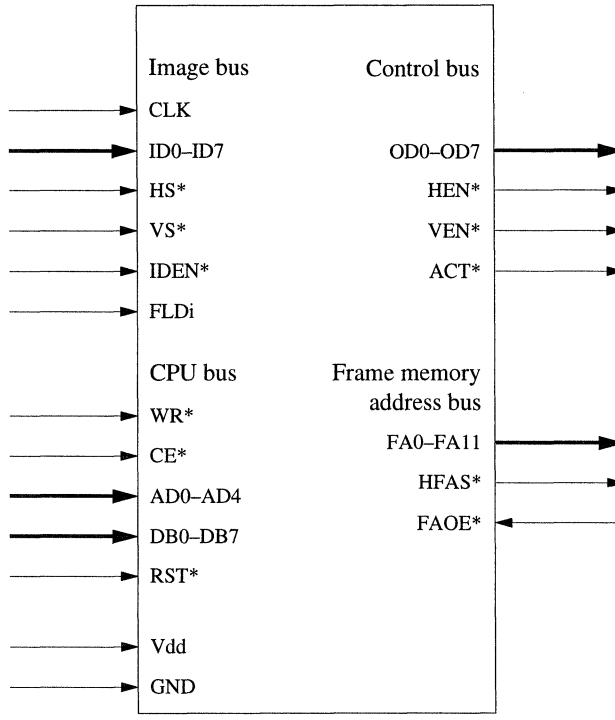
Note 1: If the effective area output signals (VEN*, HEN*) are in pulse form, the width of the signal can not be expanded. The signals are operative at low level. At high level, the counter stops and the data shift register remains on hold. This signal should remain at low level in normal use.

Note 2: FA0–FA11 frame memory output pins have high-Z state with pull-up resistors.

Note 3: The RST* pin has Schmitt-trigger input with pull-up resistor.

Note 4: An asterisk (*) after the signal name indicates inverse logic (active low).

3.3 Schematic Pin Diagram



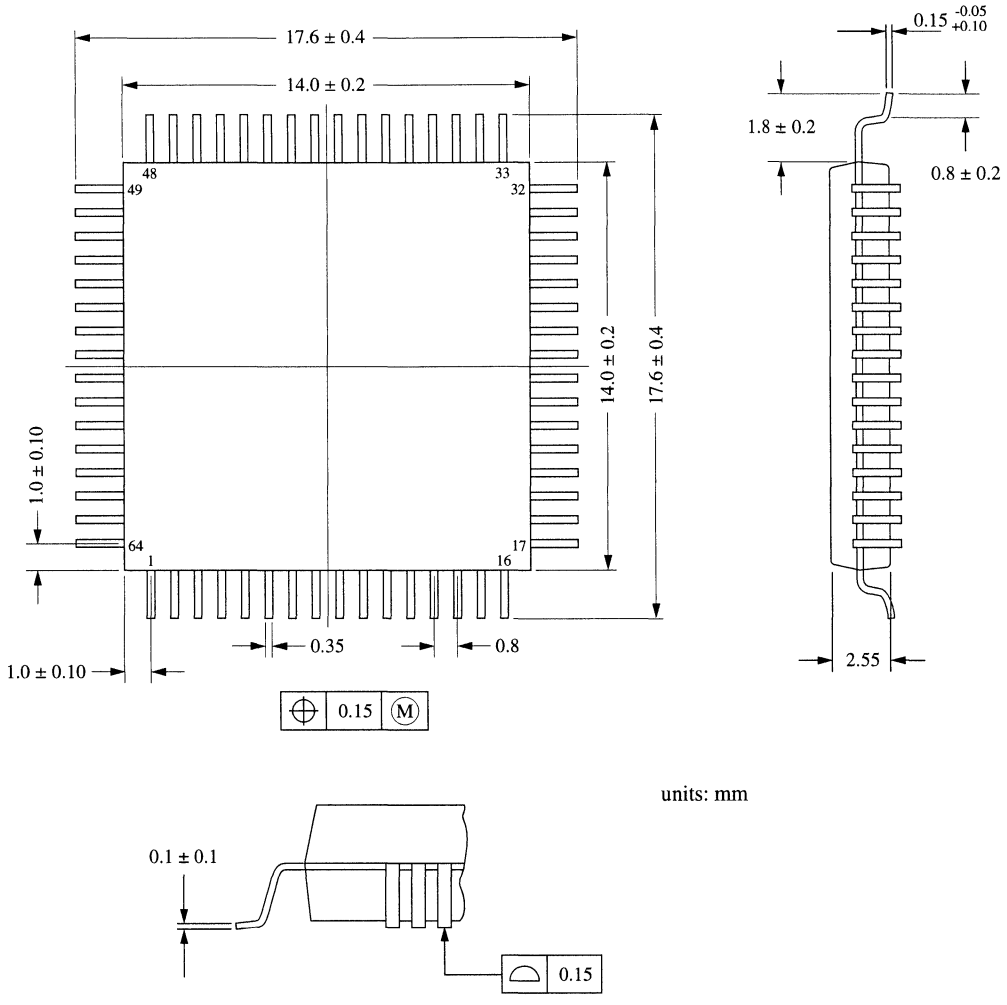
When pins OD0–OD7 are used for comparator output with binary conversion, extended numerical values are used:

- When the value is 0, OD0–OD7 output the value 00H.
- When the value is 1, OD0 to OD7 output the value FFh.

With ternary value conversion, only the three least significant bits (OD0–OD2) are used:

- When the value is 0, OD0–OD7 output the value 01h.
- When the value is 1, OD0–OD7 output the value 02h.
- When the value is 2, OD0–OD7 output the value 04h.

3.4 Package Dimensions

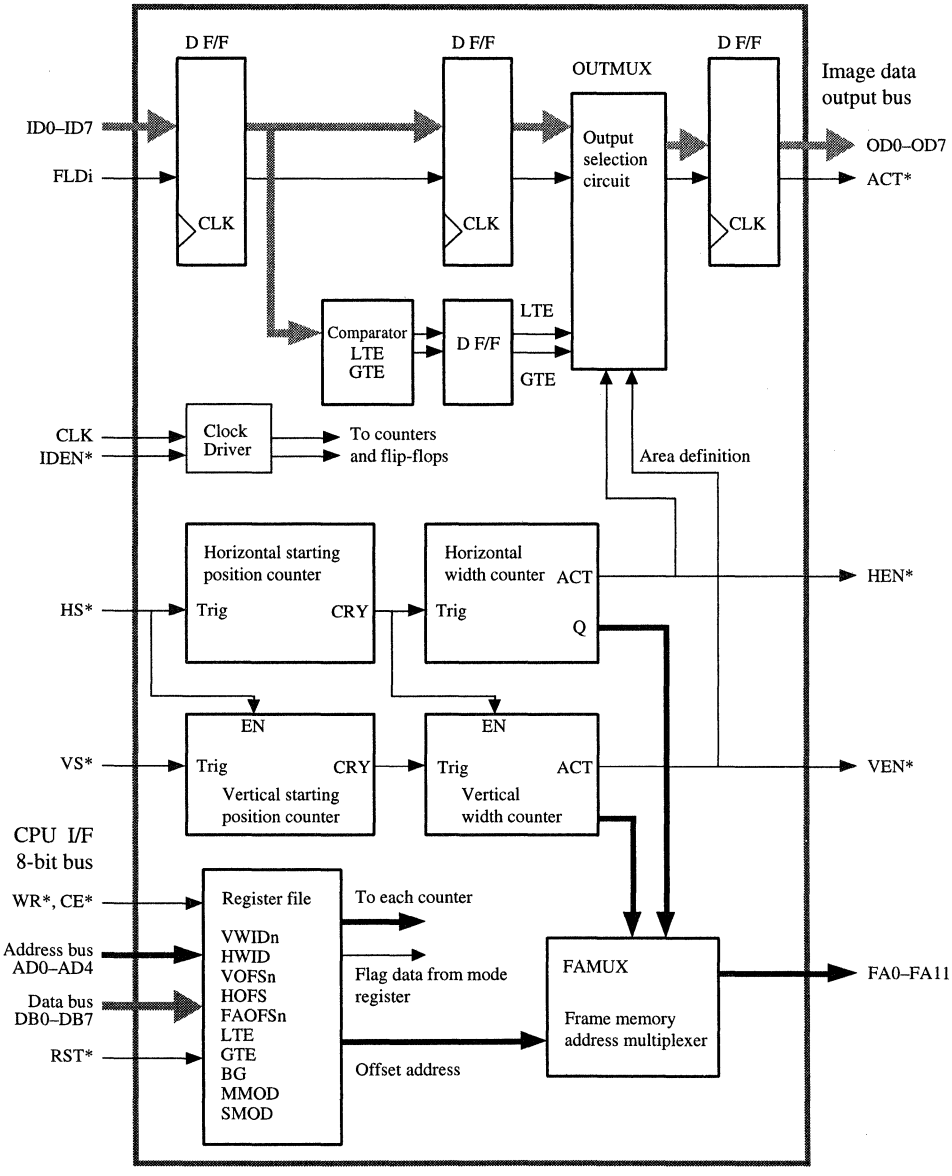


units: mm

64-Pin Plastic QFP Package

Section 4: Block Diagram

4.1 Block Diagram



Note: This block diagram only illustrates the functions of the IP90C51, and does not represent all the chip's circuits and functions. For descriptions of functions and timing, refer to the appropriate sections of this manual.

Section 5: Registers

5.1 Register List

All registers are write-only.

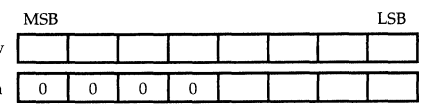
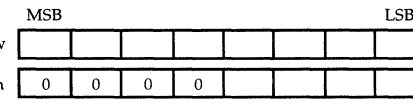
Address	Name		Bits	Description
00h	HWID	Low	8	Screen data width
01h		High	4	
02h	VWID1	Low	8	Odd-numbered field height
03h		High	4	Non-interlaced image height
04h	VWID0	Low	8	Even-numbered field height
05h		High	4	
06h	HOFS	Low	8	Screen data horizontal offset
07h		High	4	
08h	VOFS1	Low	8	Odd-numbered field data vertical offset
09h		High	4	Non-interlaced image vertical offset
0Ah	VOFS0	Low	8	Even-numbered field data vertical offset
0Bh		High	4	
0Ch	FAOFS1	Low	8	Base address value for frame memory address for odd-
0Dh		High	4	numbered field data and interlaced image addresses
0Eh	FAOFS0	Low	8	Base address value for frame memory address of
0Fh		High	4	even-numbered field data
10h	GTE		8	Greater than or equal comparison data
11h	LTE		8	Less than or equal comparison data
12h	BG		8	Background data
13h	—			Reserved
14h	—			Reserved
15h	—			Reserved
16h	SMOD		8	Slave mode register
17h	MMOD		8	Master mode register

Note: All registers are reset to 00h following a hardware or software reset.

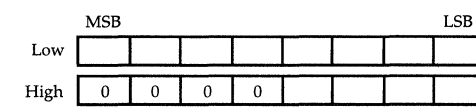
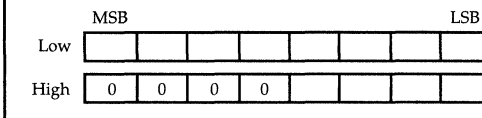
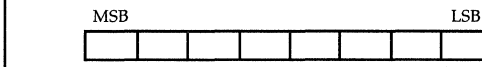
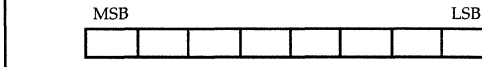
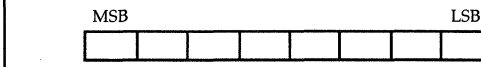
5.2 Area of Interest Setting Registers

The following registers contain settings related to the area of interest (AOI).

Parameter	Register	Description	Address	Width
Width	HWID	MSB LSB Low [ ][ ][ ][ ][ ][ ][ ][ ] High [0][0][0][0][ ][ ][ ][ ] The setting must be one less than the AOI width	00h 01h	12 bits
	VWID1	MSB LSB Low [ ][ ][ ][ ][ ][ ][ ][ ] High [0][0][0][0][ ][ ][ ][ ] The odd-field AOI height for interlaced scanning, or the odd-field AOI height for non-interlaced scanning. This register is used when FLDi is high.	02h 03h	12 bits
	VWID0	MSB LSB Low [ ][ ][ ][ ][ ][ ][ ][ ] High [0][0][0][0][ ][ ][ ][ ] The even-field AOI height for interlaced scanning. Ignored in non-interlaced mode. This register is used when FLDi is low.	04h 05h	12 bits
Horizontal offset	HOFS	MSB LSB Low [ ][ ][ ][ ][ ][ ][ ][ ] High [0][0][0][0][ ][ ][ ][ ] The AOI screen data horizontal offset. When a low HS* signal is detected at the rise of CLK, the data input at the rise of the next CLK is considered D(0), and successive values are then input as D(1), D(2), etc. When the HOFS register is set to m, the first m values D(0) through D(m-1) are ignored, and output begins with D(m).	06h 07h	12 bits

Parameter	Register	Description	Address	Width
Vertical offset	VOFS1	 The odd-numbered field AOI data vertical offset for interlaced scanning, or the AOI field data vertical offset for non-interlaced scanning. The vertical offset is defined as the number of high-to-low transitions of HS* that enter after a low VS* is input. Thus, when the VOFS register is set to the m, the first m-1 lines of data counted in the HS* low-level pulse input circuit after the low VS* pulse are ignored, and the m^{th} line of data is counted. If the falling edges of VS* and HS* occur simultaneously at the rising edge of CLK, the HS* pulse is counted normally.	08h 09h	12 bits
	VOFS0	 The even-numbered AOI field data vertical offset for interlaced scanning. This register is used when FLDi is low.	0Ah 0Bh	12 bits

5.3 Output Data Setting Registers

Parameter	Register	Description	Address	Width
Base address setting	FAOFS1	 The base address for the high-byte frame memory address for odd-numbered field data in interlaced mode, or the base address for the high-byte frame memory address in non-interlace mode. This register is used when FLDi is high.	0Ch 0Dh	12 bits
	FAOFS0	 The base address for high-byte frame memory data for even numbered field data in interlaced mode. Ignored in non-interlace mode. This register is used when FLDi is low.	0Eh 0Fh	12 bits
Comparator	GTE Register	 The minimum comparator data value for comparison (greater than or equal).	10h	8 bits
	LTE Register	 The maximum comparator data value (less than or equal).	11h	8 bits
Background	BG Register	 Data value for all areas (background) outside the AOI defined by HOFS, VOFSn, HWID, and VWIDn.	12h	8 bits

5.4 Operating Mode Select Registers

Set the IP90C51's operating modes with the main mode (MMOD) and secondary mode (SMOD) registers.

5.4.1 Main Mode Register (MMOD)

Reset value: 00h

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Address
tp	p	vs	f	c3	c	e	f	17h

b0: Reset. Write 1 to this bit for reset.

After reset, this bit automatically returns to 0 (so it is not necessary to write 0 to clear the reset condition).

b1: Execute. Write 1 to this bit to execute processing, 0 to stop.

- b1 = 1: Execute
- b0 = 0: Stop

Execution occurs when the IDEN* (input data enable signal) is enabled and this bit is set to 1.

b2: Comparator enable.

- b2 = 1: Comparator output
- b2 = 2: 8-bit data output (input 8-bit data is output as 8-bit data in transparent mode)

b3: Binary/ternary conversion switch.

- b3 = 1: Ternary output (01h, 02h, 04h)
- b3 = 0: Binary output (00h, FFh)

b4: Frame memory address. Controls ON/OFF setting of frame memory address output pins FA0–FA11.

- b4 = 1: Frame address output
- b4 = 0: Frame address pin OFF (high-impedance output)

Has the same function as the FAOE* pin, so the frame memory address signal is enabled when b4 = 1 and FOAE* = low (the frame memory is in high-impedance state at all other times). Note that frame memory address output pins are pulled up so that they are not floating when the next data values arrive.

b5: VS resistor sync. Sets the VS sync mode, which determines whether changes written to register values become valid at the start of the next field.

Only this bit (VS_bit) and the b0 bit (reset bit) are not affected by the VS_bit changes. They become valid when new values are written.

- b5 = 1: VS synchronous mode (changes written to registers become valid at the start of the next field)
- b5 = 0: VS asynchronous mode (changes written to registers become valid as soon as they are written)

b6: Pulse output enable. Switches pulse and level output of the HEN* and VEN* signals.

- b6 = 1: Pulse output mode (outputs a pulse of one clock-cycle width at the start of the area of interest)
- b6 = 0: Level output mode (remains active throughout the area of interest)

b7: Transparent. Sets data values other than area of interest (AOI).

- b7 = 1: Transparent mode (input data is output unchanged)
- b7 = 0: Background data fill-in mode (outputs value from the BG register)

5.4.2 Secondary Mode Register (SMOD Register)

Reset value: 00h

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0	Address
0	0	0	0	0	0	0	fo	16h

b0: Field output bit. Determines whether the ACT* signal pin outputs the ACT* signal or the output field ID signal FLD0.

The ACT* signal is low only when the VEN* and HEN* signals are both low.

- b0 = 1: Output field ID signal
- b0 = 0: Output ACT* signal

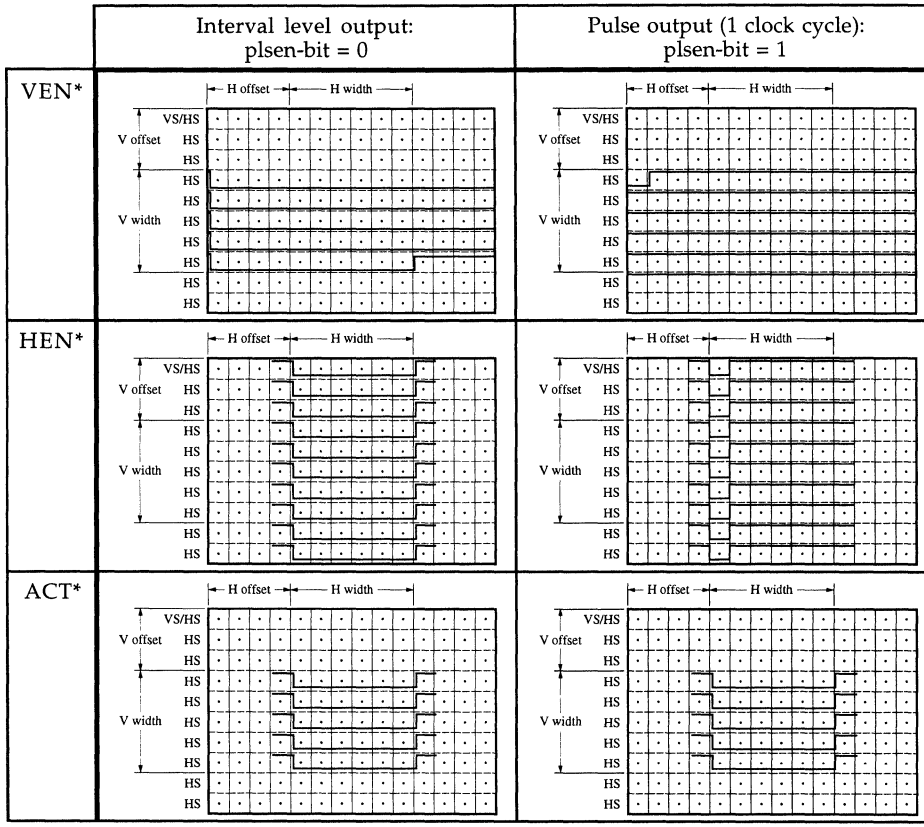
Note 1: Write 0 to bits b1–b7.

Note 2: The output field ID signal FLD0 is output with a delay of three clock cycles relative to the input field ID signal FLDi. This is the same as the delay between input and output of the digital image data itself. This is an effective way to match the delay of the field ID signal to synchronize it with the delay of the data signal when handling images with short blanking periods in interlaced mode.

Section 6: Operating Description and Control Methods

6.1 Programmable Area Definition

The size and location of the area of interest (AOI) are determined by programmable offset and size settings in the VOFS, HOFS, VWID, and HWID registers. Output signals can be synchronized by using the VEN*, HEN*, and ACT* signals. VEN* and HEN* signal output can be switched between level output or pulse output (with a width of one clock cycle) using bit 6 of the MODE register (plsen-bit). The example below uses the following values: VOFS = Voffset, HOFS = Hoffset, VWID = Vwidth, HWID = Hwidth-1.



Note 1: The VS* signal input and the first HS* signal need not be synchronized.

Note 2: The ACT* signal is not related to the value of the plsen-bit. To set this signal to width 1, set the HWID register to 0.

6.2 Comparator Output

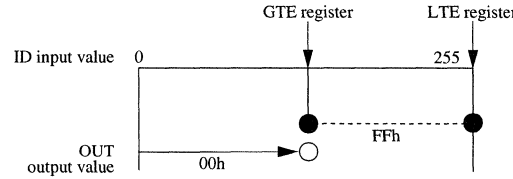
The IP90C51's two comparators can output two-value or three-value data for clipped regions of the image data input, or delay the output. The function of the comparators is set by bits b2 and b3 in register MMOD, as shown in the following table.

MMOD		Clipping region output mode
b2	b3	
0	*	Input data
1	0	Two-value output (00h, FFh)
1	1	Three-value output (01h, 02h, 04h)

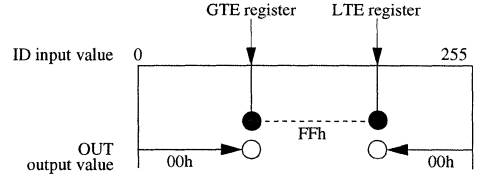
The following figures show examples of these modes.

Two-Value Mode Using a Single Comparator

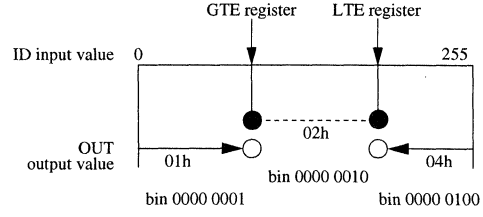
This mode uses register GTE for comparison when register LTE contains FFh, and uses register LTE for comparison when register GTE contains 00h.



Two-Value Mode Using Window Comparators



Three-Value Mode For Individually Specified Regions

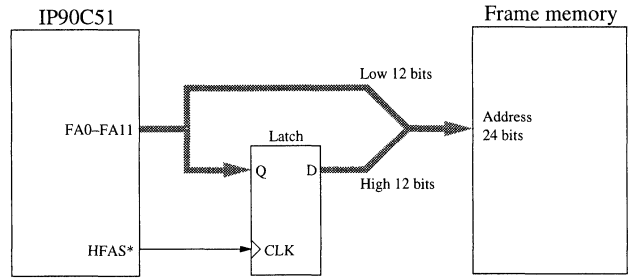


6.3 Frame Memory Addresses

The IP90C51 outputs low and high addresses on a 12-bit multiplexed address bus using the address strobe signal HFAS*, as shown in the figure below.

1) Full Use of Maximum Memory Capacity: 24-Bit Addresses

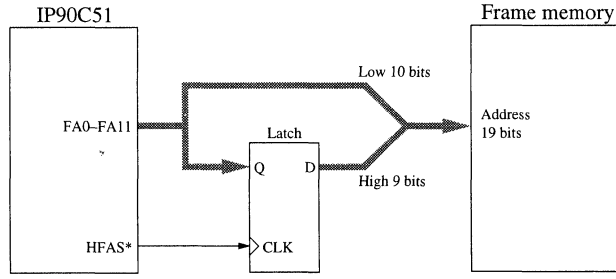
The IP90C51 can output frame memory addresses for image sizes up to 4095 x 4095 pixels, using a 24-bit address format. Though the chip generates addresses, it does not handle memory read/write timing signals (see illustration below).



Data can pass through the IP90C51 in either direction, either as data output from frame memory to the image data bus, or as data written from the image data bus to frame memory. Two IP90C51 chips are required to handle both directions simultaneously.

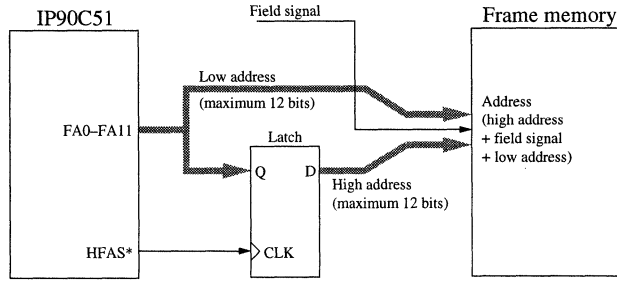
2) Using Interlaced Images and Narrower Address Space

Areas of interest defined with sizes less than 4095 pixels (12 bits) can be easily handled by reducing the value of the low address bit for the width and the value of the high address bit for the height. For example, an AOI of 640 x 480 pixels can be handled using a high address of 9 bits (FA0–FA8) and a low address of 10 bits (FA0–FA9).



Address Generation for a 640 x 480-Pixel Area of Interest

When interlaced images are written to or read from frame memory, data from odd numbered fields is merged line by line with data from even numbered fields. To access memory space, the field signal should be inserted between the high address and low address, and handled as part of the address signal.



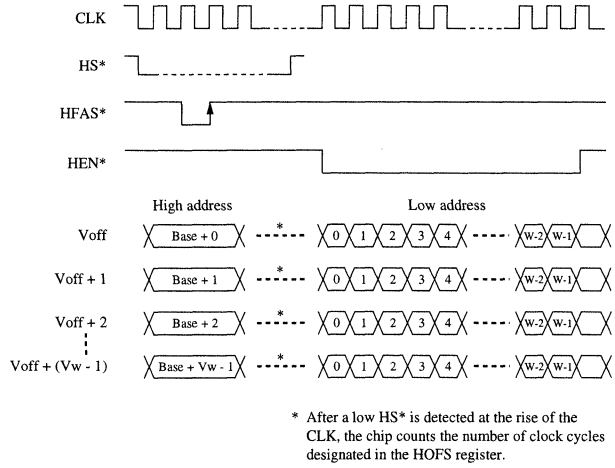
Memory Mapping of Interlaced Images by Merging Odd/Even Fields Line By Line

3) Sample Frame Memory Address Output Format

The illustration below shows the address strobe signal HFAS*, as well as the high and low address output signals. This example uses the following register settings:

- VOFS0 (vertical offset for even field data) and VOFS1 (vertical offset for odd field data) are set to Voff.
- VWID0 (vertical size for even fields) and VWID1 (vertical size for odd fields) are set to Vw.
- HOFS (horizontal offset) is set to Hoff.
- HWID (horizontal width of screen data) is set to W.
- FAOFS0 (base address value for even field data) and FAOFS1 (base address value for odd field data) are set to Base.

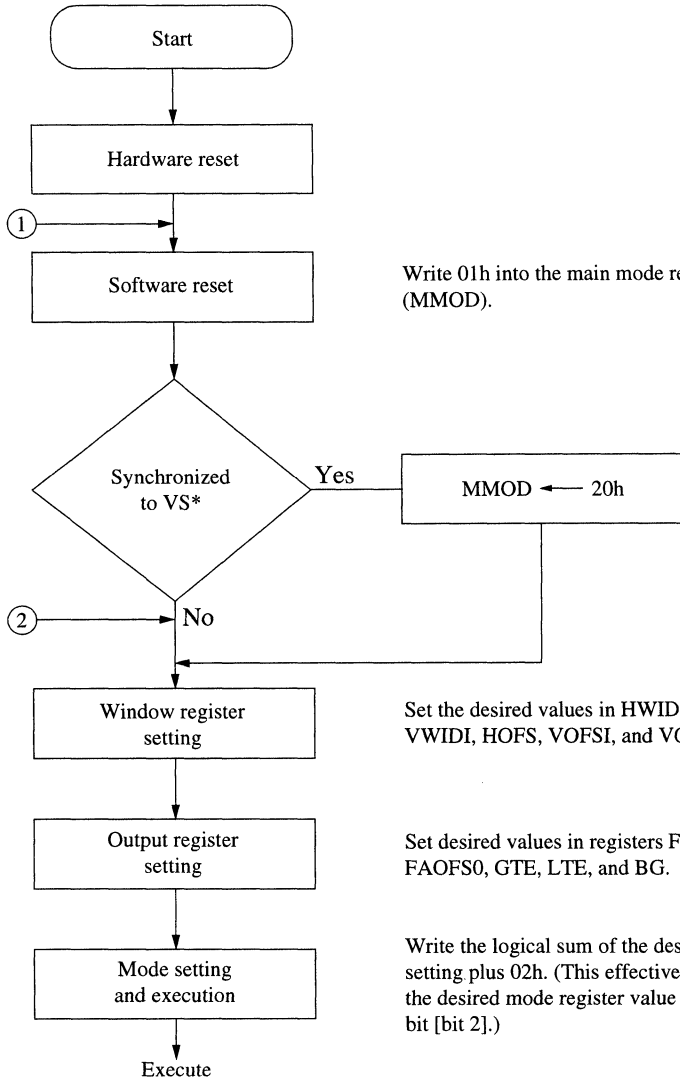
Thus, the settings for even-field data are made in the VOFS1, VWID1, and FAOFS1 registers, and settings for odd-field data are made in the VOFS0, VWID0, and FAOFS0 registers.



* After a low HS* is detected at the rise of the CLK, the chip counts the number of clock cycles designated in the HOFS register.

For details of signal timing, see Section 7.5, "AC Timing."

6.4 Control Method



Internal Register Write Sequence

Note: Change register settings as follows:

- To change the VS* synchronization mode (synchronous to asynchronous or asynchronous to synchronous), start at point ①.
- If the register VS* synchronization mode does not change, start at point ②.

Section 7: Electrical Characteristics

7.1 Absolute Maximum Ratings

Parameter	Symbol	Rating	Unit
DC supply voltage	Vdd	-0.5 to 6.5	V
Input/Output voltage	V_i/V_o	-0.5 to Vdd + 0.5	V
Output current range	I_o	20	mA
Operating temperature	T_{opt}	0 to 70	°C
Storage temperature	T_{stg}	-10 to 80	°C

7.2 Recommended Operating Conditions

(Ta = 0 to 70 °C)

Parameter		Symbol	Condition	Min.	Typ.	Max.	Unit
Power-supply voltage		Vdd		4.75	5.0	5.25	V
Input voltage		Vi	TTL level	0	—	Vdd	V
Input voltage	High level	V_{ih}	TTL level normal input	2.2	—	Vdd	V
	Low level	V_{il}		0	—	0.8	V
Voltage	Positive-trigger	V_p	TTL level inputs Schmitt-trigger	1.2	—	2.4	V
	Negative-trigger	V_n		0.6	—	1.8	V
	Hysteresis	V_h		0.3	—	1.5	V
Input rising time		T_{ri}	TTL level inputs	0	—	200	ns
Input falling time		T_{fi}		0	—	200	ns
Input rising time		T_{ri}	Schmitt-trigger	0	—	10	ms
Input falling time		T_{fi}	inputs	0	—	10	ms

7.3 DC Characteristics

Vdd = 5 V ± 5%
Ta = 0 to 70 °C

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Quiescent supply current (Note 1)	I_O	V_i = Vdd or GND	—	—	200	µA
Maximum Supply Current	I_{DD}	V_{dd} = MAX, Ta = 70°C	—	—	80	mA
Output OFF-state leakage current	I_{oz}	$V_0 = V_{dd}$ or GND	—	—	10	µA
Output OFF-state leakage current with pull-up resistor	I_{ozr}	$V_0 = V_{dd}$ or GND	-40	-100	-270	µA
Input clamping voltage	V_{ic}	I_1 = 18 mA	-1.2	—	—	V
Output short-circuit current (Note 2)	I_{os}	V_0 = 0 V	-250	—	—	mA
Input leakage current with no pull-up resistor	I_i	$V_i = V_{dd}$ or GND	—	10 pA	10	µA
Input leakage current with 50 kΩ pull-up resistor (Note 3)	I_i	V_i = GND	-40	-100	-270	µA
Low-level output current	I_{ol}	V_{ol} = 0.4 V	6.5	13	—	mA
High-level output current	I_{oh}	V_{oh} = 2.4 V	-0.5	-13	—	mA
Low-level output voltage	V_{ol}	I_{ol} = 0 mA	—	—	0.1	V
High-level output voltage	V_{oh}	I_{oh} = 0 mA	2.6	3.4	—	V

Note 1: Excluding the current through the pull-up and pull-down resistors.

Note 2: Output short-circuit current is for a duration of 1 second (max).

Note 3: Pins with have pull-up resistors are the reset pin RST* and frame memory address pins FA0–FA11.

7.4 Input/Output Pin Capacitance

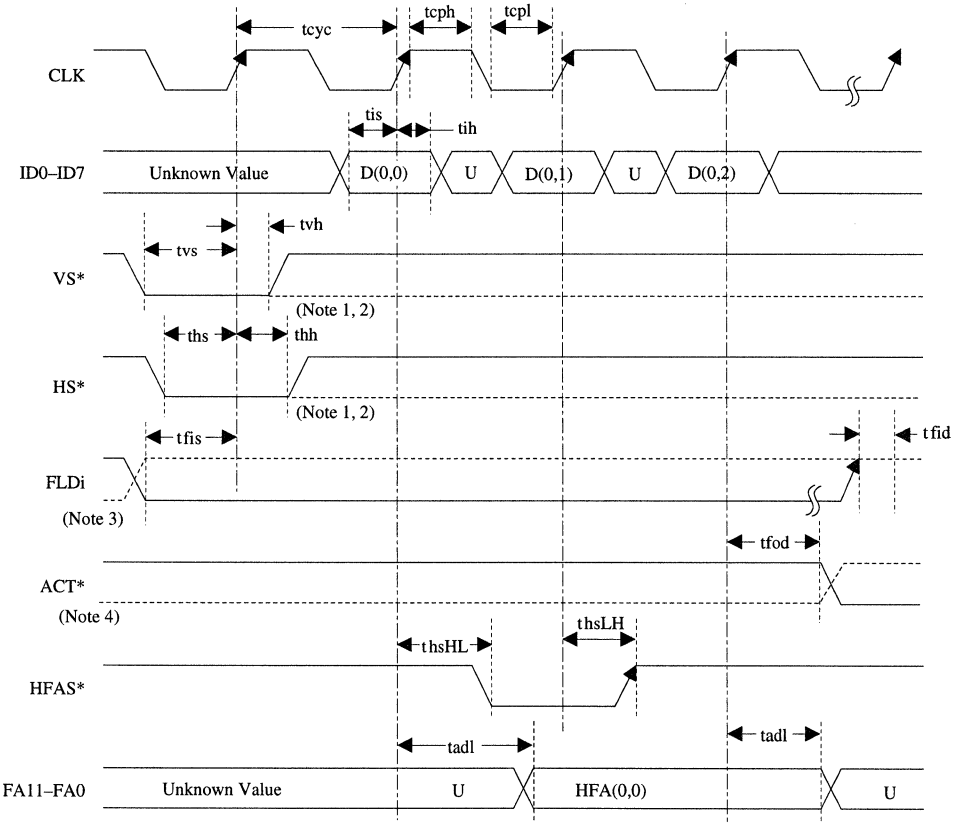
Vdd = V_i = 0 V

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Input capacitance	C_{in}	f = 1 MHz	—	10	25	pF
Output capacitance	C_{out}	f = 1 MHz	—	10	25	pF

7.5 AC Timing

(Output pin load capacity C_L = 30 pF)

1) VS Cycle Timing



Data Timing at Frame Starting Point

Note 1: VS* and HS* can be input simultaneously. Any HS* signals that precede a VS* signal are ignored.

Note 2: VS* and HS* are not necessarily pulses. They can also remain constant, as indicated by the dashed lines. However, they must go high at least two cycles before the next synchronous input.

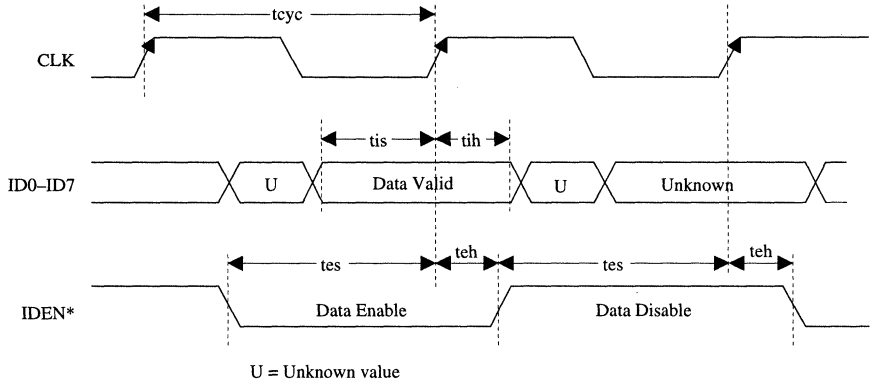
Note 3: Keep FLDi high from the time VS* is input until the data valid region finishes.

Note 4: The function of ACT* is selected by bit 0 in register SMOD. When bit 0 is 1, the field input signal on FLDi is delayed, then output synchronously on ACT*.

Unit: ns

Parameter		Symbol	Min.	Typ.	Max.
Clock	Video clock cycle period	t_{cyc}	27.7	—	—
	Video clock high-level pulse width	t_{cph}	10.0	—	—
	Video clock low-level pulse width	t_{cpl}	10.0	—	—
Input	ID0–ID7 setup time	t_{is}	2.1	—	—
	ID0–ID7 hold time	t_{ih}	2.6	—	—
	VS* setup time	t_{vs}	3.0	—	—
	VS* hold time	t_{vh}	2.0	—	—
	HS* setup time	t_{hs}	3.0	—	—
	HS* hold time	t_{hh}	2.5	—	—
	FLDi setup time	t_{fis}	4.0	—	—
	FLDi hold time	t_{fid}	2.0	—	—
Output	ACT* delay time	t_{fod}	4.0	—	12.5
	HFAS* delay time	t_{hsHL}, t_{hsLH}	4.5	—	13.0
	FA0–FA11 delay time	t_{ad1}	4.5	—	16.0

2) Data Input Timing



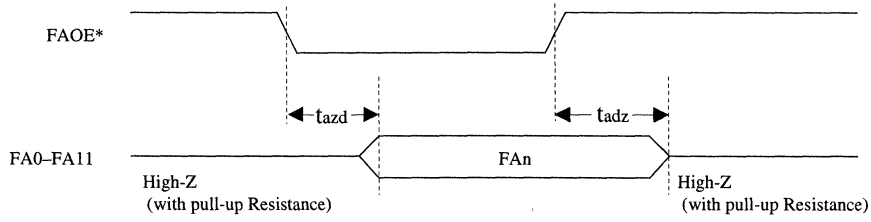
Clock and Input Data Enable Signal Timing

Unit: ns

Parameter	Symbol	Min.	Typ.	Max.
Video clock cycle period	t_{cyc}	27.7	—	—
ID0–ID7 setup time	t_{is}	2.1	—	—
ID0–ID7 hold time	t_{ih}	2.6	—	—
IDEN* setup time	t_{es}	14.5	—	—
IDEN* hold time	t_{eh}	0.0	—	—

Note: The IDEN* signal makes it possible to stop the internal counter and hold image data pixel by pixel. However, if the effective area output is in pulse output video width, the pulse cannot be extended.

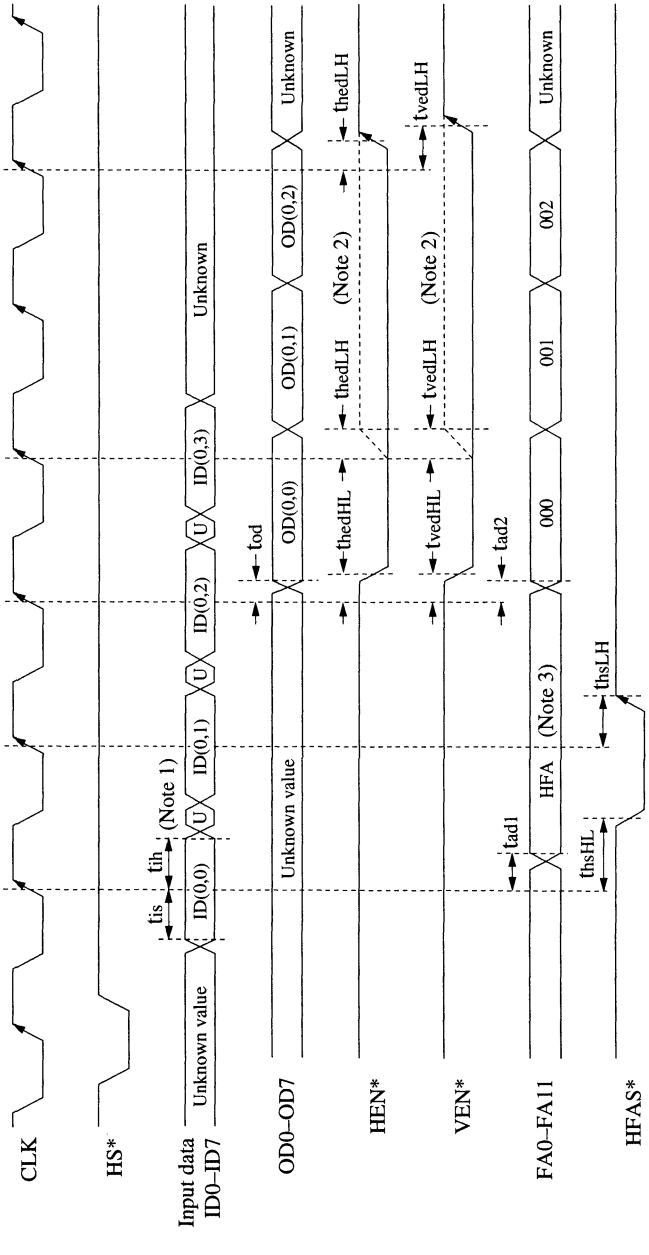
3) Frame Address Output Control Timing



Unit: ns

Parameter	Symbol	Min.	Typ.	Max.
FAOE* falling edge to data valid delay	t_{azd}	—	—	15.0
FAOE* rising edge to high-impedance delay	t_{adz}	—	—	12.0

4) Data Input/Output Cycle Timing



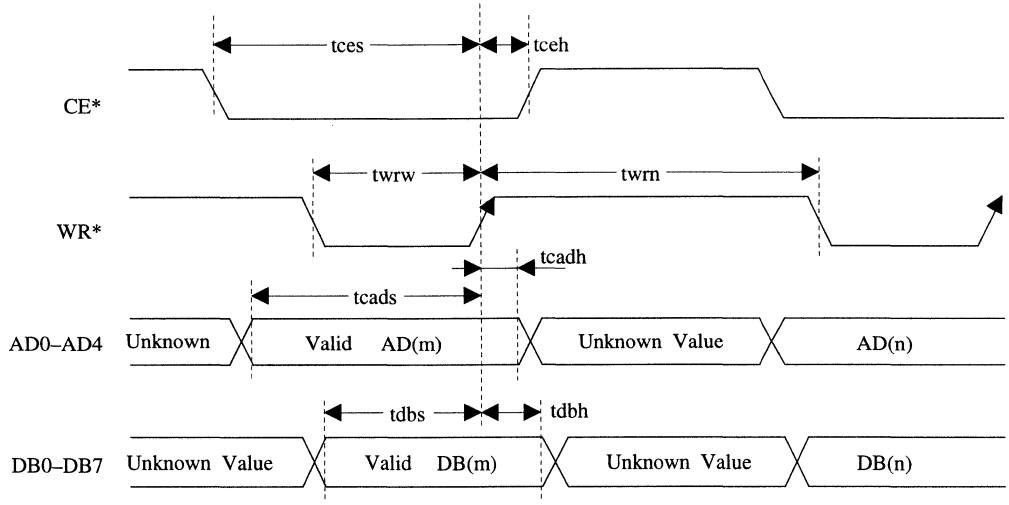
Note 1: U = Unknown value.
Note 2: Dotted lines show timing in pulse output mode.
Note 3: HFA = high address output.
Note 4: The VEN* signal is output three or four cycles after the HEN* signal.

Data Input/Output Timing for Each Line

Unit: ns

Parameter	Symbol	Min.	Typ.	Max.
ID0–ID7 setup time	t_{ts}	2.1	—	—
ID0–ID7 hold time	t_{ih}	2.6	—	—
CD0–CD7 transition delay time	t_{cd}	4.0	—	12.5
HEN* delay time	t_{hedHL} t_{hedLH}	5.0	—	15.5
VEN* delay time	t_{vedHL} t_{vedLH}	5.0	—	15.5
FA0–FA11 high-byte data delay time	t_{ad1}	4.5	—	16.0
FA0–FA11 low-byte data delay time	t_{ad2}	4.5	—	16.0
HFAS* delay time	t_{hsHL} t_{haLH}	4.5	—	13.0

5) CPU Interface Timing



Note 1: Writing to registers from the CPU does not have to be synchronized to CLK. The WR* signal pin acts as a write clock signal from the CPU, so that the written data is synchronized with the rising edge of CLK as shown.

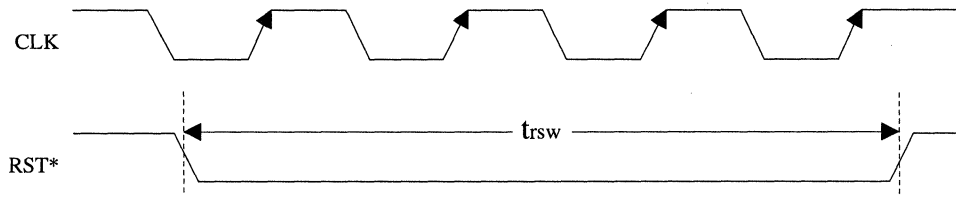
Note 2: Register data become internally valid two clock cycles after the rising edge of WR*.

Note 3: Register values can be changed during operation by writing 01h to register MMOD: this causes a soft reset, which requires all register values to be re-entered. Note that performing a soft reset before writing to a register prevents any incorrect counter operation caused by previous counter status or register values.

Unit: ns

Parameter	Symbol	Min.	Typ.	Max.
CE* setup time (from WR* rise)	t_{ces}	15	—	—
CE* hold time (from WR* rise)	t_{ceh}	0	—	—
WR* low-level pulse width (low period)	t_{wrw}	15	—	—
WR* high-level pulse interval	t_{wrm}	50	—	—
AD0–AD4 setup time from WR* rise	t_{cads}	15	—	—
AD0–AD4 hold time from WR* rise	t_{cadh}	0	—	—
DB0–DB7 setup time from WR* rise	t_{dbs}	15	—	—
DB0–DB7 hold time from WR* rise	t_{dbh}	0	—	—

6) Reset Timing



Note: The minimum reset signal pulse width is three cycles.

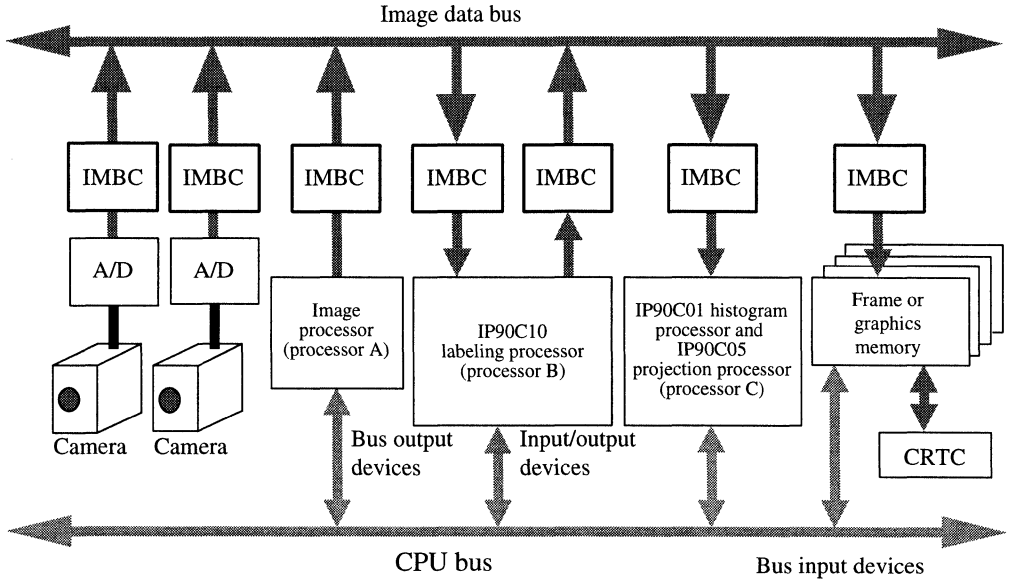
Unit: ns

Parameter	Symbol	Min.	Typ.	Max.
RST* low-level pulse width (low period)	t_{rsw}	3	—	—

Section 8: Sample Applications

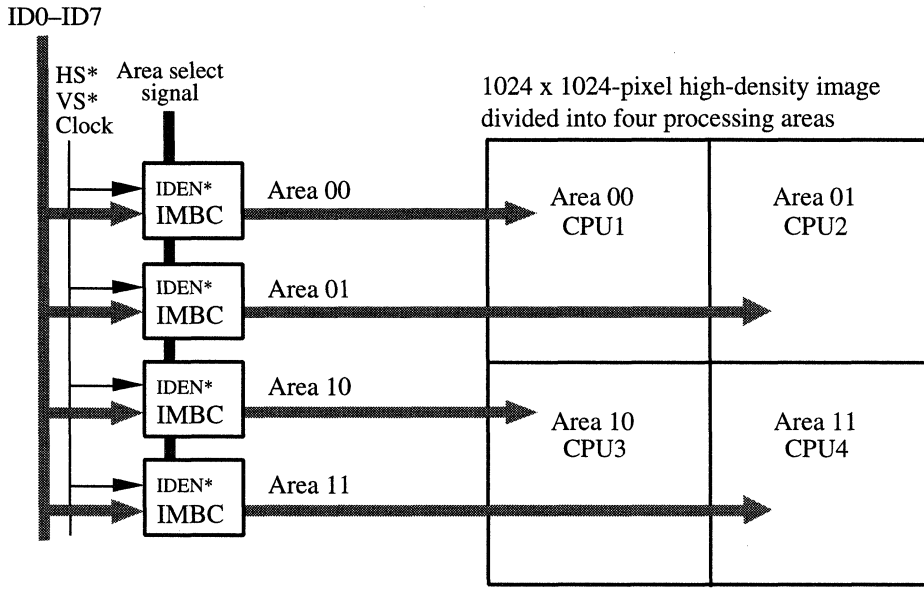
8.1 Image Data Bus Interface

1) Image Data Bus Common Input/Output Controller

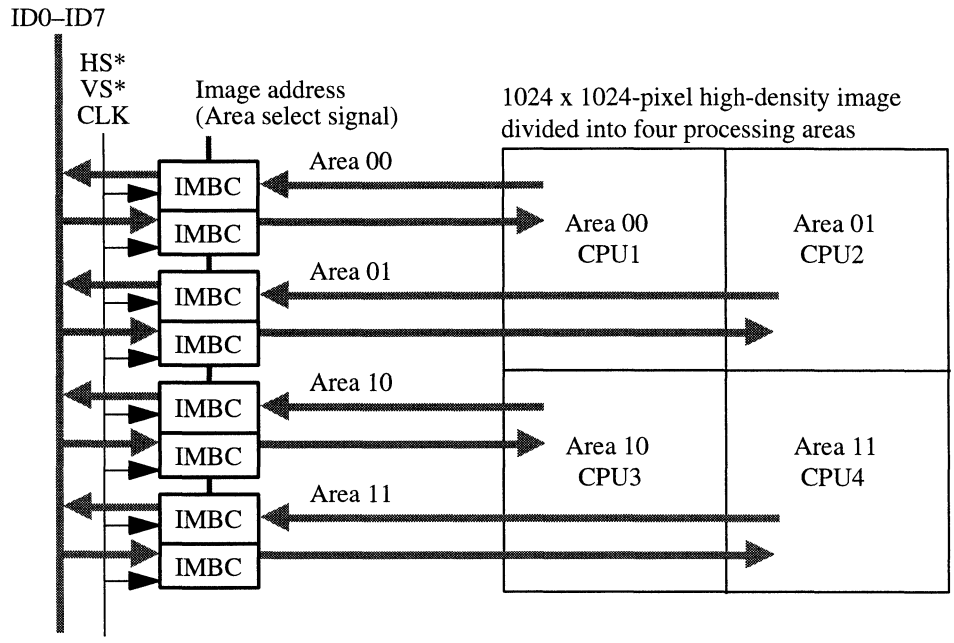


2) Image Data Transmission Using Multiple Processors

2-1) High-Speed, High-Density Parallel Image Processing

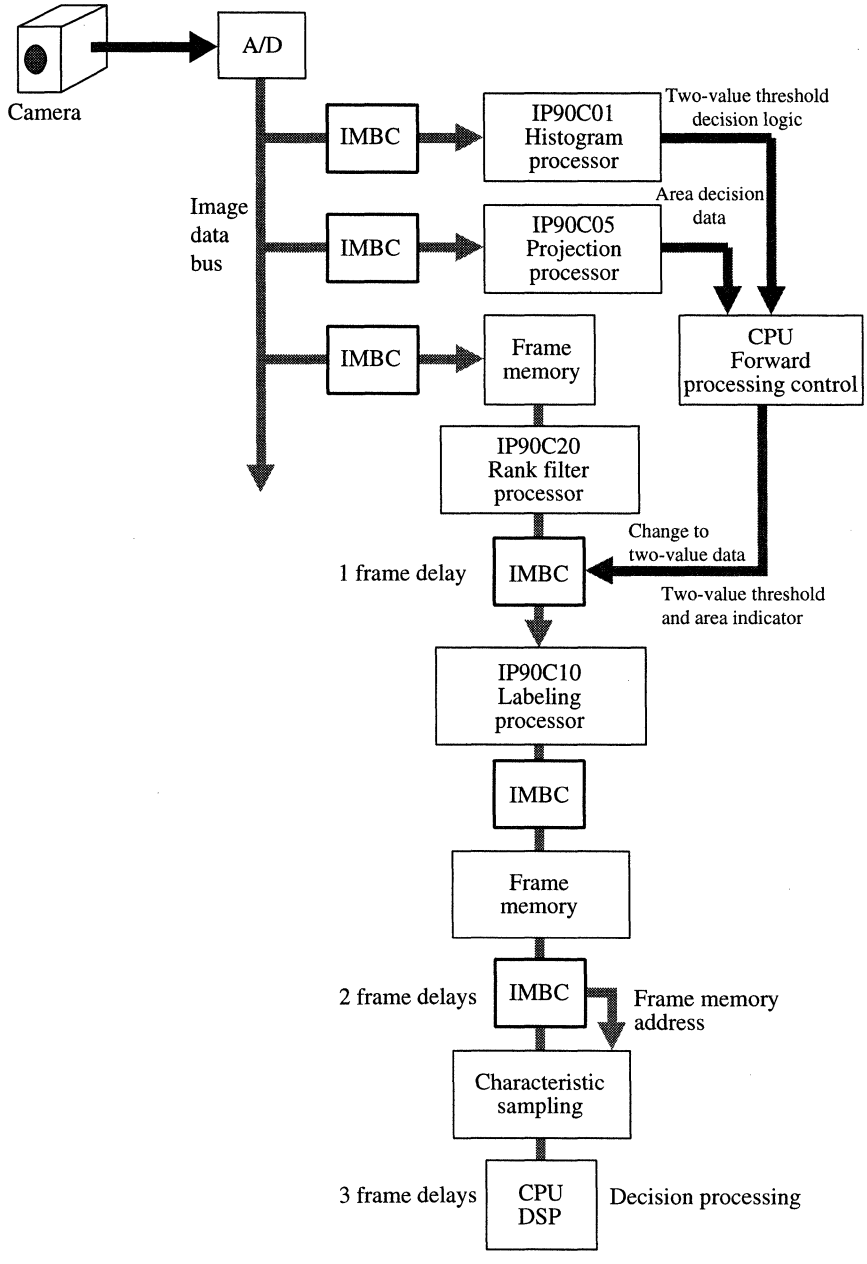


2-2) High-Speed, High-Density, Dual Bidirectional Image Processing

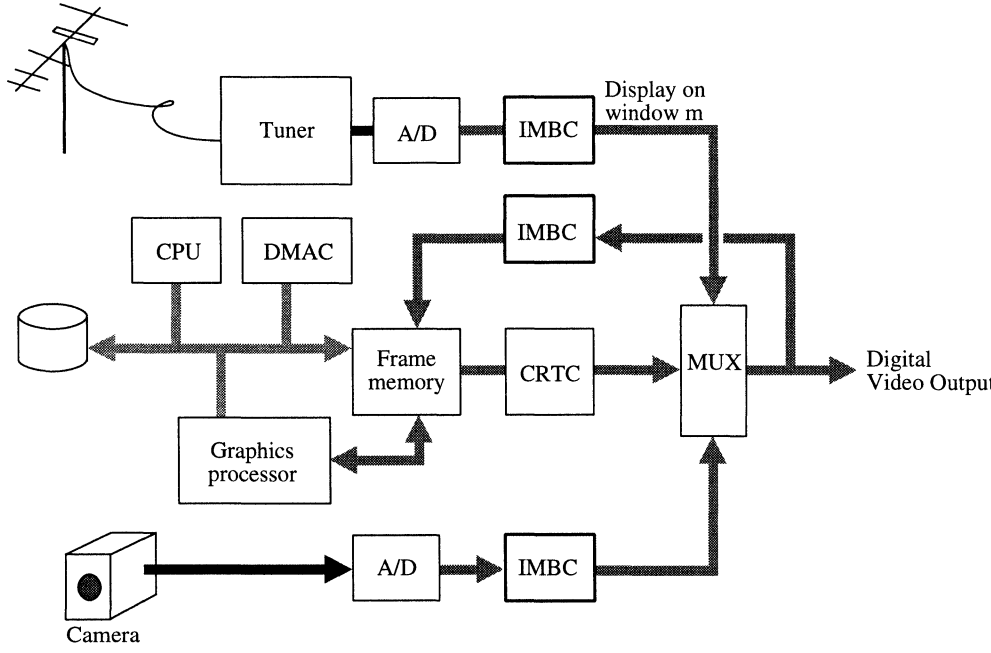


Note: Individual areas can be selected by software using the screen offsets, or by hardware connected to the data enable input.

3) Data Transmission Control for Real-Time Image Processors

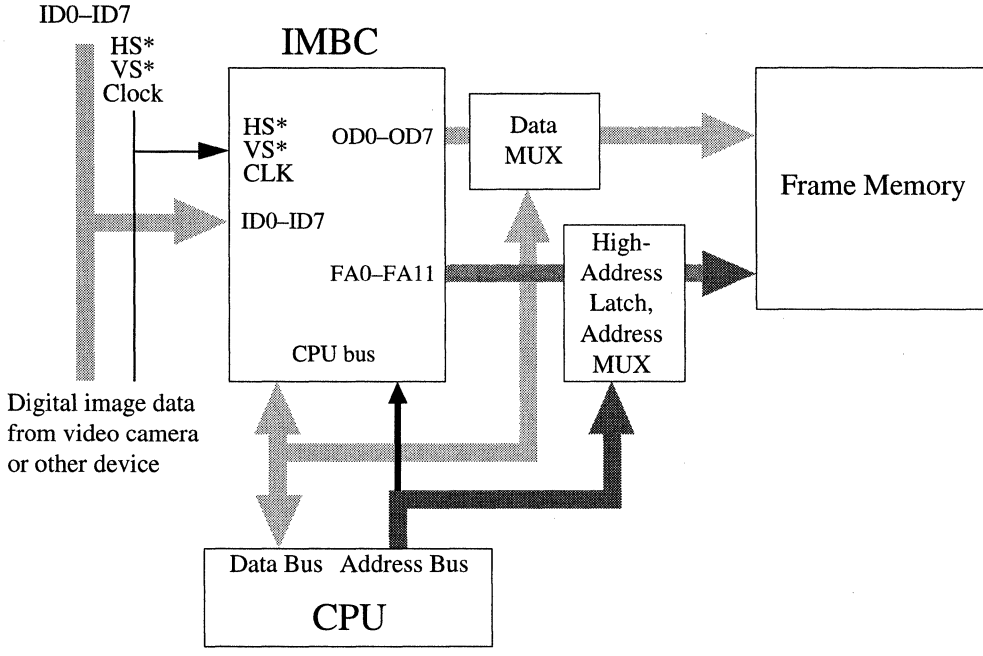


4) Multiple Window Overlap High-Speed Processor and Display

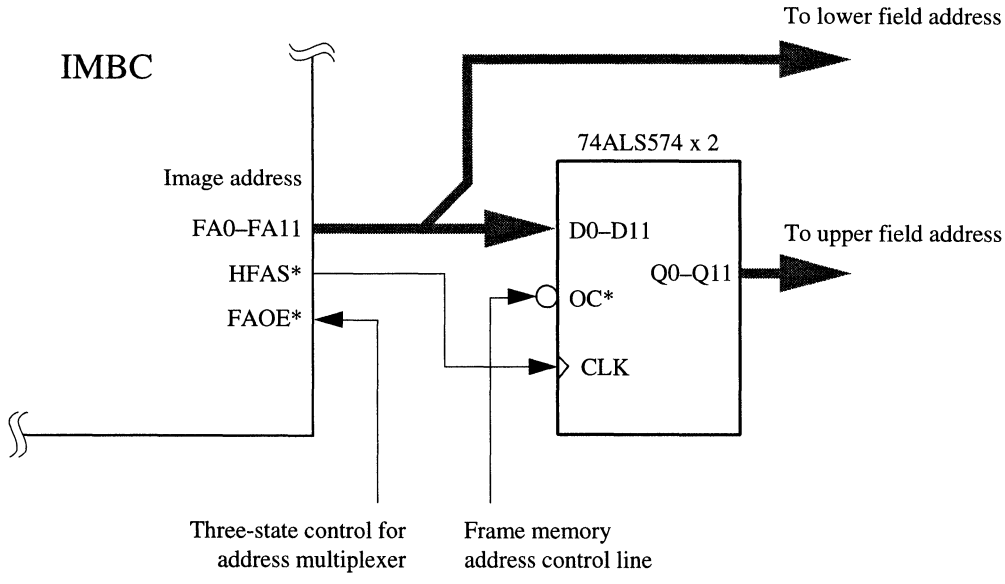


8.2. Frame Memory Interface

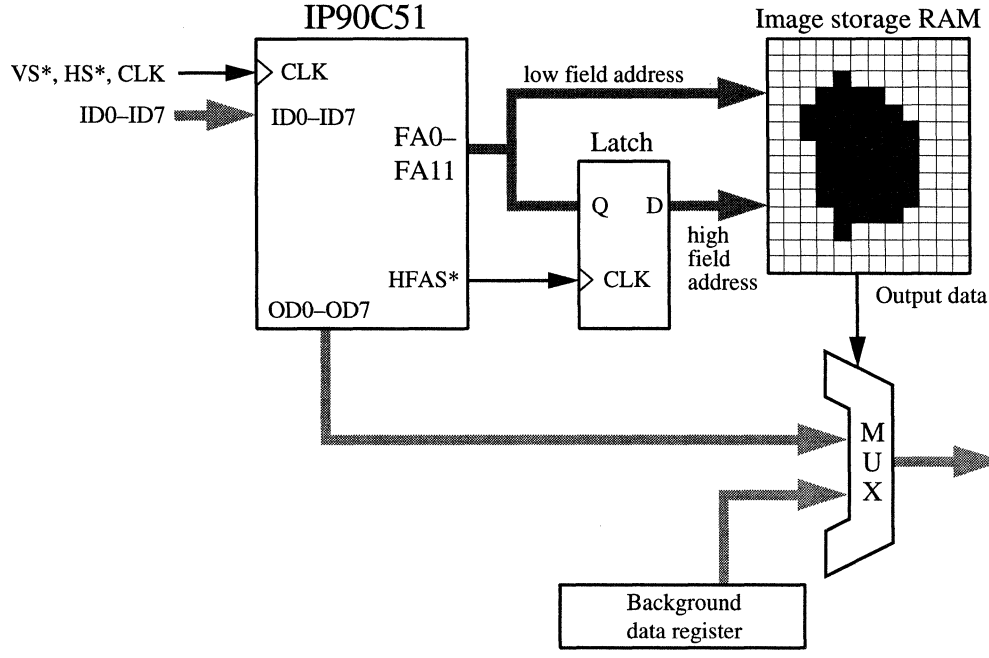
1) Frame Memory Input Controller



2) Frame Memory Address Output Upper Field Latch



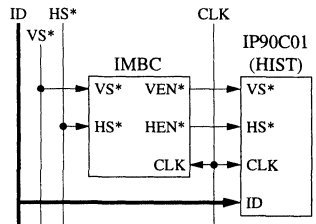
3) Data Extraction for Designated Shapes

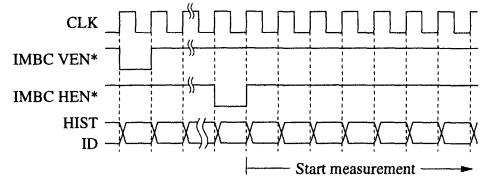


8.3 IP90C-Series Interfaces

The IMBC chip uses the standard interface for the Sumitomo image processing ASSP series. Programmable settings can be made for easy configuration of interfaces to any chip in the ASSP series.

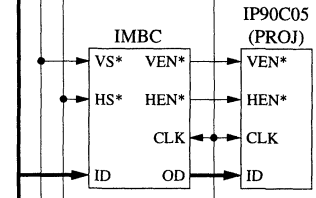
Histogram Processor

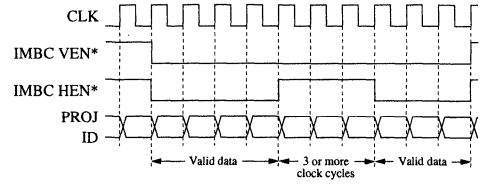




1. IP90C01 VS* and HS* signal input (IMBC VEN* and HEN* signals) should be input at intervals of one clock pulse.
2. The effective area is defined by the horizontal and vertical axis settings in effect on the IP90C01 at the rise of VS* and HS*.

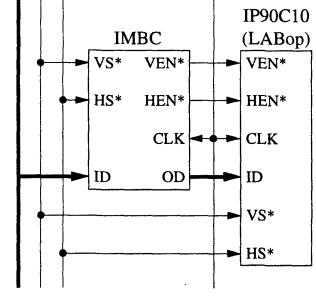
Projection Processor

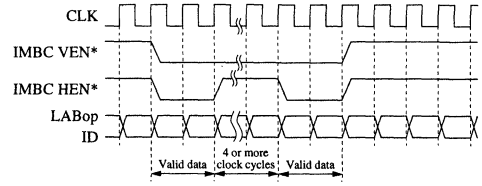




1. Set the IP90C05 exec bit while VEN* is high.
2. The IP90C05 VEN* and HEN* input should be held level.
3. The effective area is the area for which VEN* and HEN* are low.
4. The IP90C05 input ID is latched at the fall of CLK.
5. HEN* requires 3 or more clock cycles to recover.

Labeling Accelerator





1. Separate VS* and HS* signals are required.
2. The IP90C10 VEN* and HEN* input should be held level.
3. The effective area is the area for which VEN* and HEN* are low.
4. HEN* requires 4 or more clock cycles to recover.
5. The ID signal input to the IP90C10 is converted to binary form before output from the IMBC chip.

Note 1: The CLK, VEN*, and HEN* signals are controlled independently by each chip.

Note 2: When using more than one IP90C-series chip in a system, use a separate IMBC chip as an interface for each chip.

Note 3: For design information, consult the technical manuals for each ASSP chip.

SMI ASSP

Image Processing LSI Series

IP90C55

Image Data Stream Controller (IMSC)

Sumitomo Metal Industries, Ltd.
Technical Manual Ver. E 1.3

Sumitomo Metal Industries, Ltd.

Table of Contents

Section 1: Functions and Features

- Image Data Stream Switching

  The IP90C55 IMSC (Image Data Stream Controller) chip is a digital image data stream switcher that allows any desired path to be set through its 12 digital image data input/output ports (8 bits of data per port). The IP90C55 also contains delay circuits and output-enable terminals that prevent bus conflicts with peripheral LSIs when ports are switched for input or output. The chip allows switching of image paths at any time, as well as switching paths by field.

- ALU and Barrel Shifter

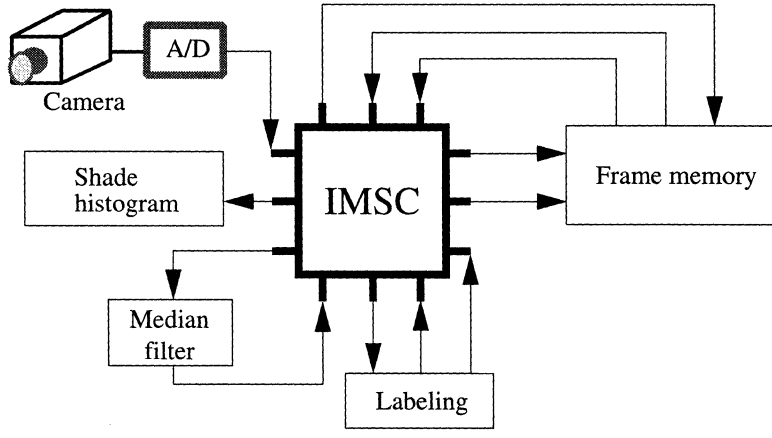
  The IP90C55 contains two pairs of 8-bit ALUs (Arithmetic Logic Units), which provide arithmetic and logic calculations between frames, a choice of large or small (MAX, MIN) calculations, and the ability to shift bits to the right. In addition, when an addition or subtraction results in an overflow or borrow condition, the ALUs' limiter function limits the calculation result to the maximum or minimum value at the bit width. Two ALUs can also be connected together to function as a 16-bit unit.

- Eight Pairs of Effective Image Area Specifying Signals

  The IP90C55 provides 8 pairs of signals for specifying effective image areas (AOI-n*, HEN-n*). These signals allow the timing delay of each pixel to be adjusted to cut out a rectangular "area of interest" (AOI) selected from within the image. The chip can therefore process image areas as large as 65,536 x 65,536 pixels (such as those produced by high-resolution-scanned images), or use a one-shot function to generate an AOI signal for a single frame of image data. The IP90C55 can be used with interlaced or non-interlaced image data.

  Furthermore, the IP90C55 has a vertical synchronization signal enable pin (VSEN*) that allows it to easily synchronize with other image processing LSIs and programmable controllers.

  Note: An asterisk (*) after a signal name denotes inverse logic.



The IP90C55 has the following specifications:

- Maximum image data transfer rate: 40 MHz
  - The transfer rate is limited by the ability of the device to dissipate heat generated by power use, which in turn depends on the device's package. For details, see Section 4, "Pin Functions and Packages."
- Maximum image area: 65,536 x 65,536 pixels (16 bits x 16 bits)
- Low-power CMOS process
- Power supply: single 5V source
- Input/output level: TTL level
- Packages:
  - 184-pin plastic QFP (used for operations at frequencies up to ~20 MHz): mold section = 32 mm^2, pin pitch = 0.65 mm
  - 181-pin ceramic PGA (used for operations at frequencies up to ~40 MHz): footprint = 40 mm^2, pin pitch = 2.54 mm

Section 2: Device Operation

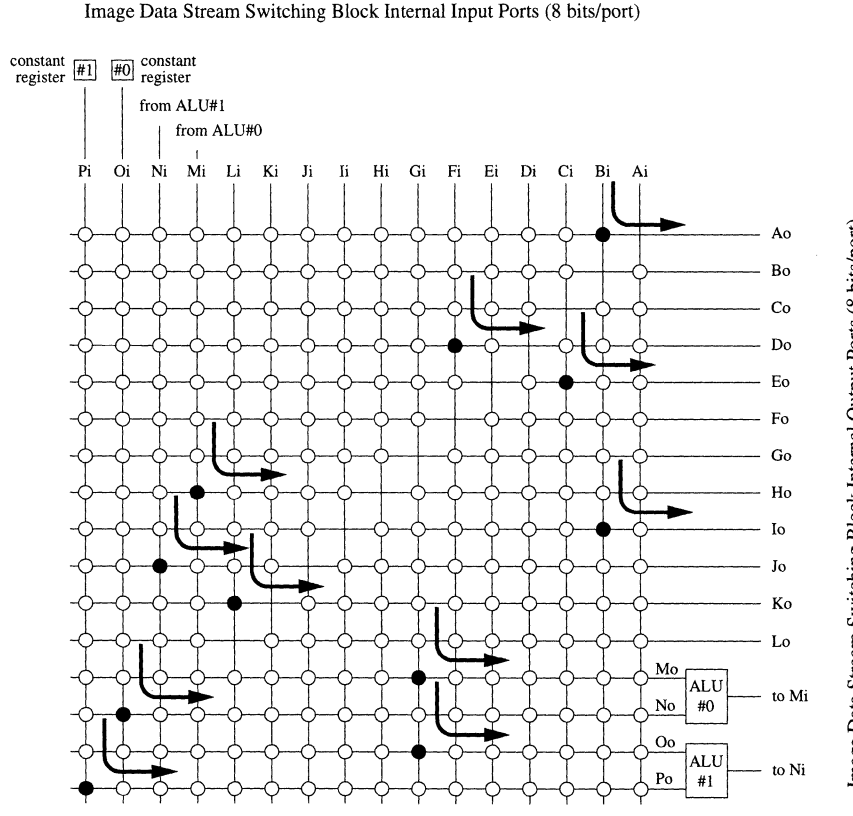
2.1 Specifying the Image Data Transfer Path

Ports PA–PL are the IP90C55's external input/output (bidirectional) terminals, and can transmit 8 bits per port. These ports are connected to input ports Ai–Li and output ports Ao–Lo in the internal image data stream switching block (see the diagram below). Ports PA–PL are set individually (in 8-bit units) for either input or output. When reset, all ports are set for input. Each of the switching block output ports Ao–Po chooses data from one of the switching block input ports Ai–Pi. When reset, data from the Ai port is selected for all output ports. Switching block input ports Oi–Pi are connected to the constant registers.

Input to ALU#0 (8 bits) is connected to internal output ports Mo and No of the image data stream switching block; its output (also 8 bits) is connected to input port Mi of the image data stream switching block.

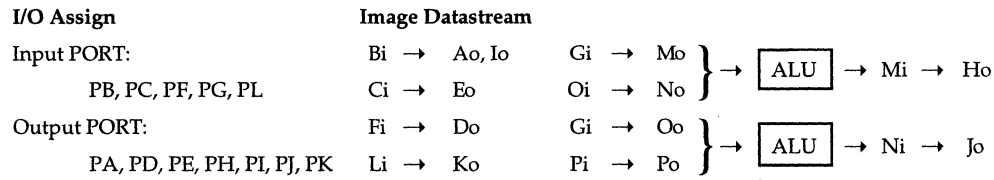
Input to ALU#1 is connected to internal output ports Oo and Po of the image data stream switching block; its output is connected to internal input port Ni of the image data stream switching block.

These connections remain unchanged even when ALU#0 and ALU#1 are connected together to function as a 16-bit unit.



Sample Setting for the Image Data Stream Switching Block

Sample Connection Settings



2.2 Connection to Internal Ports and Functional Outline of the ALUs

The ALUs are connected to the image data stream switching blocks as follows:

- The IP90C55 contains ALU#0 and ALU#1, and has basic functions equivalent to an 8- or 16-bit 74LS181 processor.
- Ports Mo and No are connected to the input ports of ALU#0 (8 bits)
- Ports Oo and Po are connected to the input ports of ALU#1 (8 bits).
- Output from ALU#0 and ALU#1 are connected to ports Mi and Ni, respectively,
- ALU#0 and ALU#1 can be connected together to function as a 16-bit ALU. This coupling can be used to perform cumulative additions.

  When using the ALUs together as a 16-bit unit (ALUEXT flag hd = 1), make sure that ALU#0's function register ALUMOD0 is set to the same value as the ALU#1 function register ALUMOD1. Also, make sure that ALUEXT flags are set with sat1 = sat0 and fix1 = fix0.

The ALUs' have these functions:

- ALUs are equivalent to an 8- or 16-bit 74LS181 processor.
- ALUs perform logic operations such as NOT, NOR, zero clear, NAND, exclusive OR, exclusive NOR, through, AND, and OR.
- ALUs perform arithmetic operations such as two's-complement addition (subtraction), decrement, increment, and others.

The ALUs also have the following extended functions of the 74LS181 processor:

- Operation to shift bits right (1-clock shift) (0–8 or 0–16 bits, including the carry bit).

  This calculation can be performed in combination with 74LS181-equivalent ALU operations, and so allows bits to be shifted 16 bits to the right after an addition to obtain the carry signal in a single path.

- MAX(A, B) calculation to choose the larger of two input values.
- MIN(A, B) calculation to choose the smaller of two input values.
- Limiter function to limit addition/subtraction results to the maximum or minimum value within the bit range.

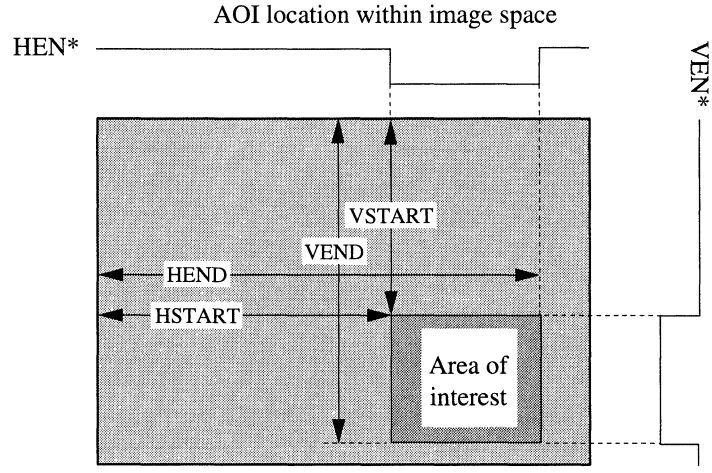
  If an add operation overflows and generates a carry, this function outputs the maximum value within the range (FFh for an 8-bit operation, FFFFh for a 16-bit operation).

  If a subtract operation underflows and generates a borrow, this function outputs the minimum value within the range (00h for an 8-bit operation, 0000h for a 16-bit operation).

Logic Operations	AND OR NOT EXOR
Arithmetic operations	+(add), - (subtract) operations with carry/borrow
Shift operation	Shift 0–8 or 0–16 bits right
Choose by comparison	MAX(A, B), MIN(A, B)

2.3 Specifying a Rectangular Image Area (AOI Function)

Image processing systems often process an area extracted from an image rather than the entire image. This extracted area is called the effective area, or the area of interest (AOI), as shown in the figure below. The IP90C55 can be programmed to clip selected rectangular AOIs from the input image. The upper-left coordinates of the AOI shown below are labeled HSTART and VSTART, and the lower-right coordinates are labeled HEND and VEND.



HSTART and HEND count image clock (CLK) pulses.
VSTART and VEND count HS* signal pulses.

As shown above, the HEN* and VEN* signals demarcate the AOI: HEN* sets the AOI's width, and VEN* sets its height. The IP90C55 sets the HSTART, HEND, VSTART, and VEND registers to output the HEN* and VEN* signals. Specifically, the IP90C55's AOI function block contains an H counter to specify the area in the horizontal direction and a V counter to specify the area in the vertical direction. The H counter is reset by an HS* pulse (derived from the HS* signal by differentiating it using ICLK) to start counting ICLK cycles. The V counter is reset by a VS* pulse (derived from the VS* signal by differentiating it using ICLK) to start counting HS* pulses. The AOI* signals (HEN*, VEN*, and ACT*) are generated by comparing these counter values against the register values. (ACT* is the logical AND product of the HEN* and VEN* signals.)

If the configuration of input image data is expressed by a two-dimensional array, as shown in Figure A, the data is transferred using the raster-scan transfer format shown by the timing chart in Figure B. For comparison, Figure C shows a field of image data with vertical and horizontal blanking periods included.

D(0,0)	D(0,1)	D(0,2)	D(0,3)		D(0,n)
D(1,0)	D(1,1)	D(1,2)	D(1,3)		D(1,n)
D(2,0)	D(2,1)	D(2,2)	D(2,3)		D(2,n)
D(m,0)	D(m,1)	D(m,2)	D(m,3)		D(m,n)

Figure A: Data Configuration With Pixels Mapped as a Two-dimensional Array

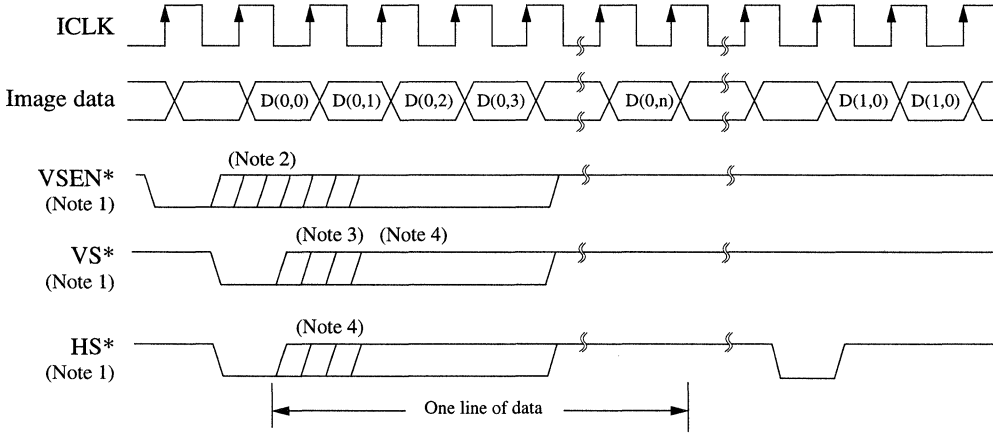


Figure B: Input Image Data Transfer Format

Note 1: Signals with names followed by an asterisk (*) are active low. VS* is a field start signal. VSEN* enables VS* input. HS* is a start signal input for each line.

Note 2: The VS* signal is enabled if the VSEN* signal is low at the leading edge of the ICLK signal one clock cycle before the clock signal at which the change of VS* to low is detected. VSEN* allows the IP90C55 to easily be synchronized with other image-processing LSIs or programmable controllers. Set VSEN* low when the IP90C55 does not need to be synchronized with other LSIs (i.e., when in normal operation).

Note 3: The VS* signal need not always be input simultaneously with HS*. HS* pulses entered before VS* is input are ignored, though HS* pulses entered simultaneously with VS* are recognized.

Note 4: Because the VS* and HS* signals are sampled by the internal clock, they need not always be pulse inputs. VS* or HS* is recognized when it is first found to be low when ICLK rises: the system interprets this as a VS* or HS* input. Once detected low in this way, VS* or HS* must go high again before the next VS* or HS* is input.

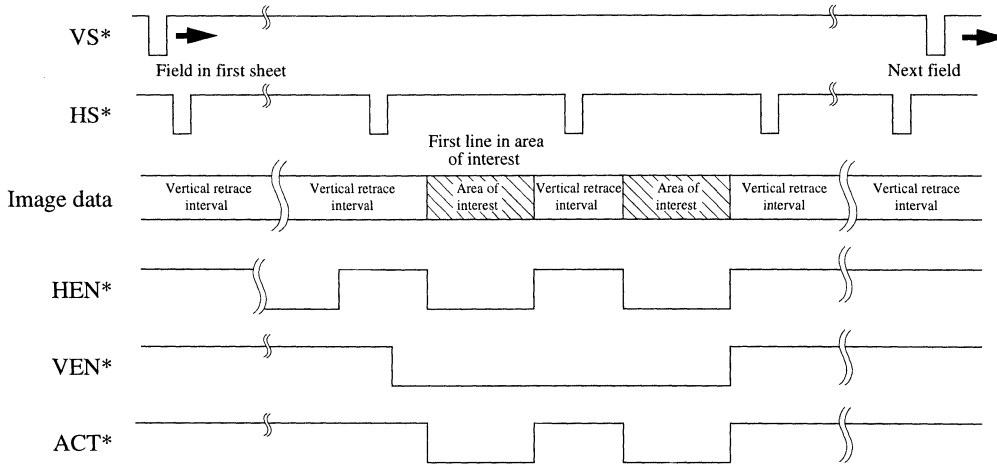


Figure C: Area of Interest on Time Axis of Image Data

The IP90C55 can output three kinds of AOI signals from its AOIn* pins: HEN*, VEN*, and ACT*. HEN* marks the AOI's width, VEN* marks its height, and ACT* is the logical AND product of HEN* and VEN*.

2.4 Example of Outputting Signals to Specify Rectangular Image Areas

The following figure shows the AOI pin output timings in various AOI output modes for the image sizes and AOI registers set values listed below.

The HEN*, VEN*, and ACT* signals each can occur in four different waveforms: Level, Single Pulse, Double Pulse, and Corner Pulse. (Corner Pulse is used only when ACT* is selected.) The Level waveform level is asserted (active) during the AOI period. The Single-Pulse waveform outputs a pulse one clock-cycle wide at the beginning of the AOI period. The Double-Pulse waveform outputs a pulse one clock-cycle wide at the beginning and the end of the AOI period. Corner Pulse generates a pulse one clock-cycle wide at the beginning of the first line of ACT*, and at the end of the last line.

Setting value:	Window size 14 x 10
HSTART - n = 0003h	VSTARTod - n = 0003h
HEND - n = 0009h	VENDod - n = 0008h

Figure 1: Level, HEN*

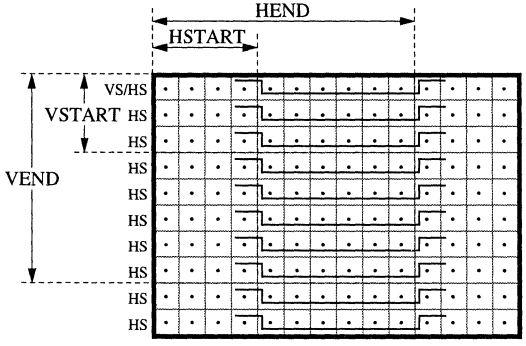


Figure 2: Level, VEN*

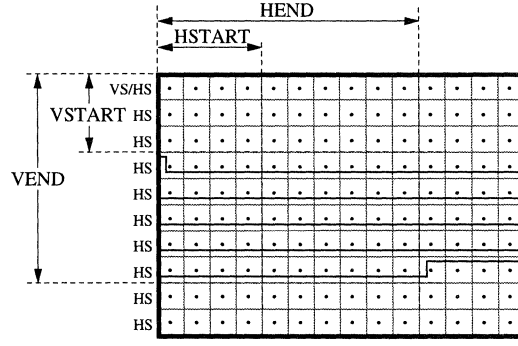


Figure 3: Level, ACT*

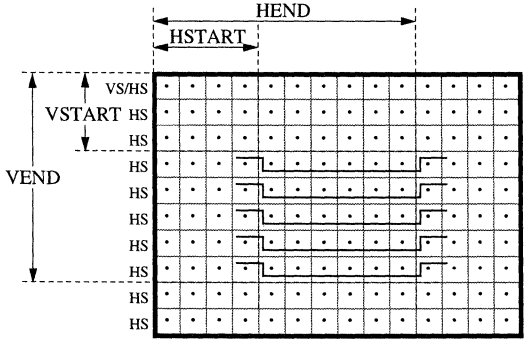


Figure 4: Single Pulse, HEN*

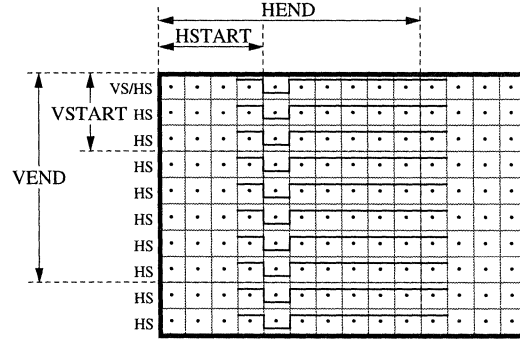


Figure 5: Single Pulse, VEN*

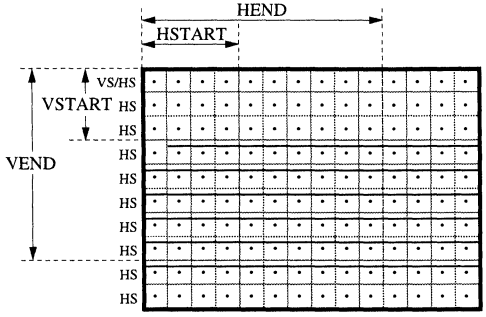


Figure 6: Single Pulse, ACT*

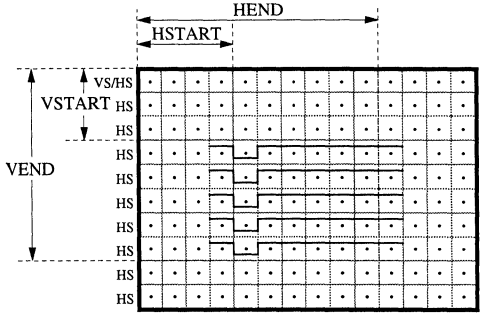


Figure 7: Double Pulse, HEN*

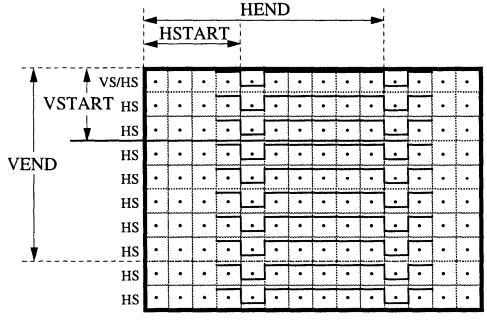


Figure 8: Double Pulse, VEN*

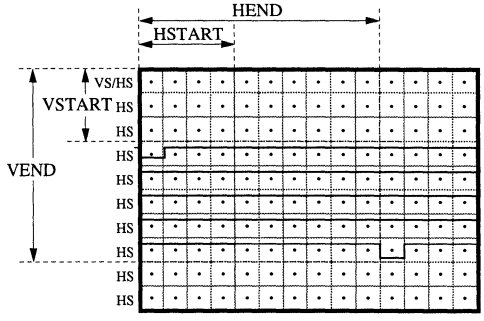


Figure 9: Double Pulse, ACT*

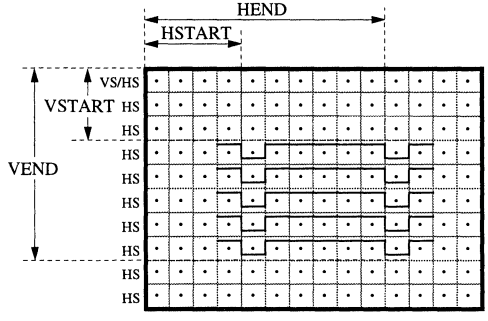
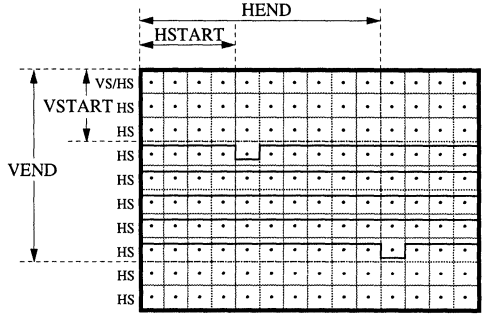


Figure 10: Corner Pulse, ACT*



Note: The vertical counter is reset at the falling edge of each VS*, regardless of whether HS* is active, and counts the number of HS* inputs. The vertical counter is also reset when VS* and HS* are asserted simultaneously. When VS* alone is asserted, the counter is set to line 0, then increments to 1 several counts later when HS* is asserted. Note that the line count is 0 if HS* and VS* are asserted simultaneously.

The figures above show examples of register settings and signal output. When HS* and VS* are asserted simultaneously, the above sample signal outputs can be obtained from these register settings:

HSTART - n = 0003h VSTARTod - n = 0002h

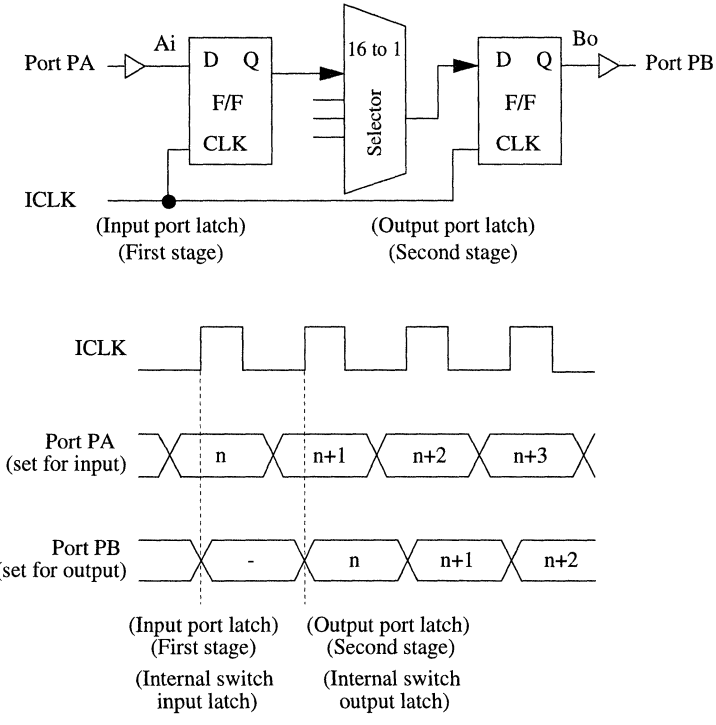
HEND - n = 0009h VENDod - n = 0007h

2.5 Delays Between Ports

2.5.1 Delays Between Image Data Input/Output Ports

The IP90C55 introduces a delay of two clock cycles when input image data is fed from the input port through to the output port.

The following figure shows this delay in image data streams. (For more information see Section 3.2, "Internal Configuration of Image Data Stream Switching Block.")

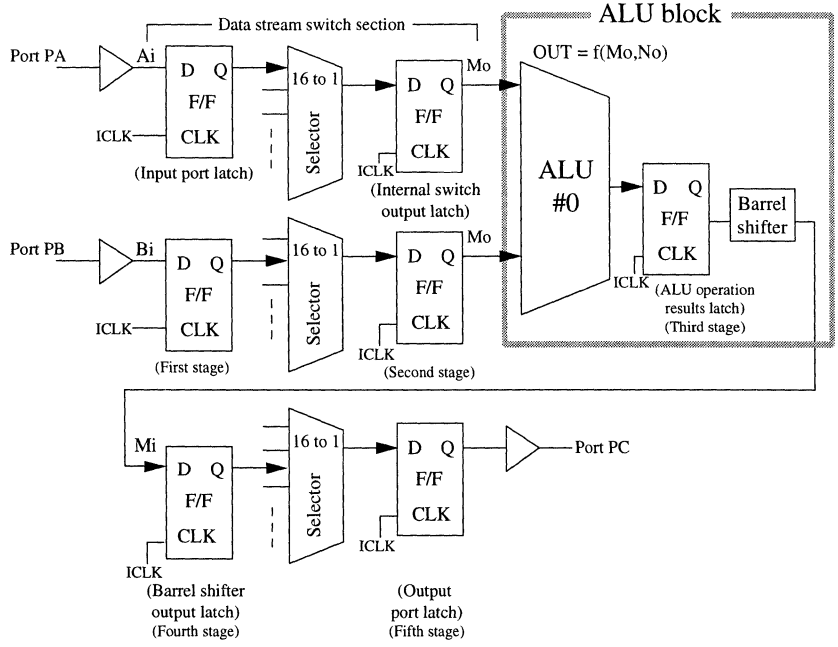


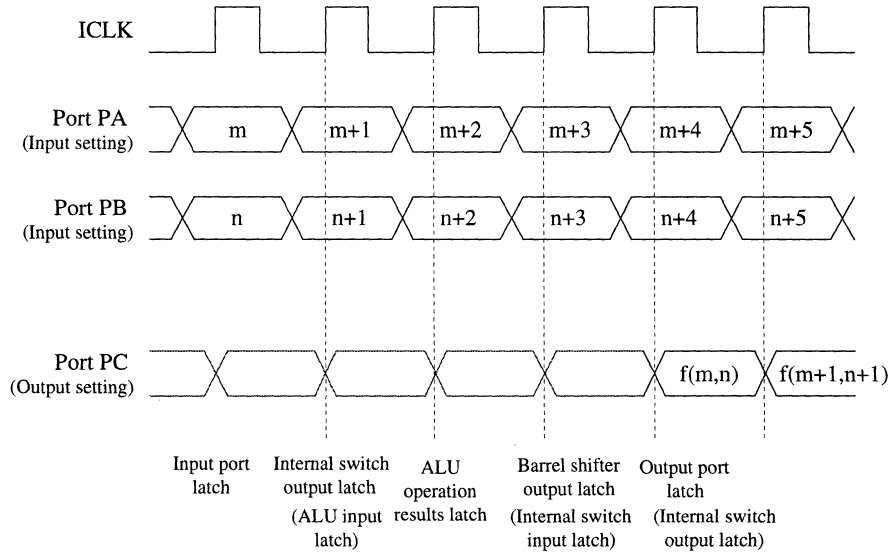
2.5.2 Delays Between Image Data Input/Output Ports When Using ALUs

The ALU blocks connected to the internal output ports of the IP90C55's image data stream switching block consist of two stages: an ALU unit and a barrel shifter section.

Outputs from these ALU blocks are connected to the internal input ports of the switching block. Consequently, a three clock-cycle delay is added here to the two-cycle delay previously described in Section 2.5.1.

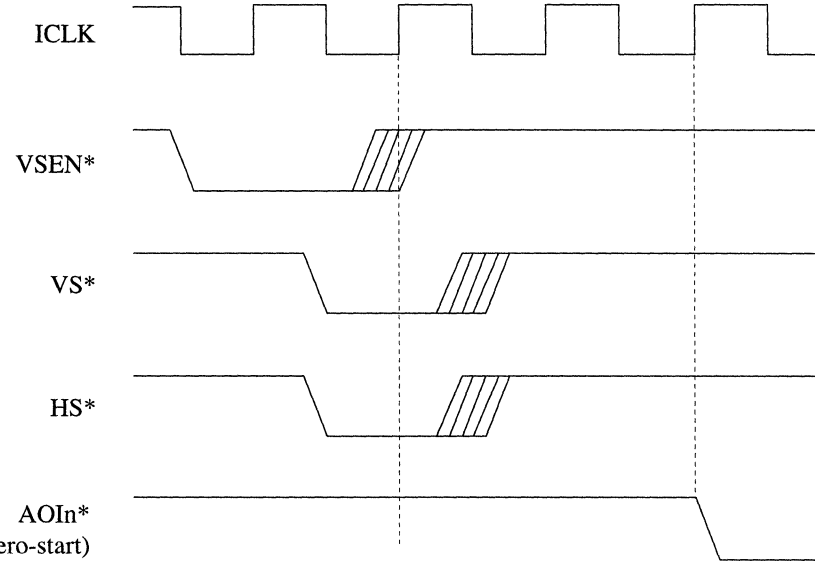
The following figure shows delays in image data streams when using the ALU units. (For more information see Section 3.2, "Internal Configuration of the Image Data Stream Switching Block.")





2.5.3 Output Delays of AOIn* Relative to HS*

If zero-start is specified (by setting HZSTART, VOZSTART, and VEZSTART to 1), AOIn* goes low two clock cycles after HS* is found to be low on the rising edge of ICLK. The output image data begins with pixel 0.



2.5.4 Processing Image Data Using AOI Signals

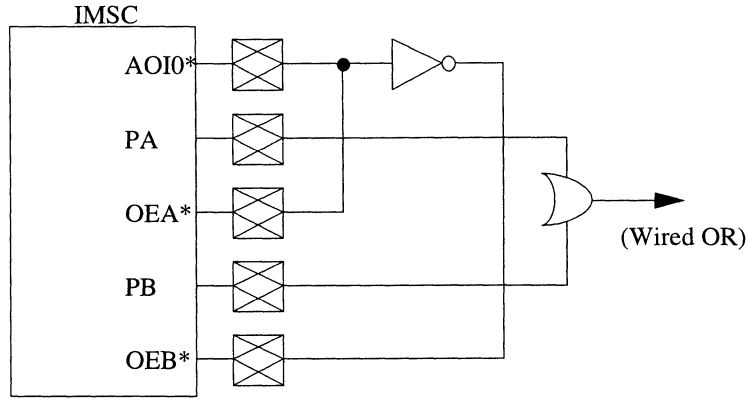
The IP90C55's AOI signals (which specify rectangular areas of interest within the image) operate independently of the IP90C55's image data switches, and so can respond flexibly to each system configuration. Therefore, control using AOI signals does not affect the image data switches or change the image data.

In addition, the chip can divide the rectangular AOI into inside and outside parts and apply image data stream path switching or image data masking to the parts individually. This can be done either of two ways:

- by entering the AOIn* signal to any of the ports PA–PL so it will be handled as image data and processed with other signals by the ALU
- by connecting the AOIn* signal directly to OEx* (where x = A, B, C, ... L) to control the output

Sample Circuits

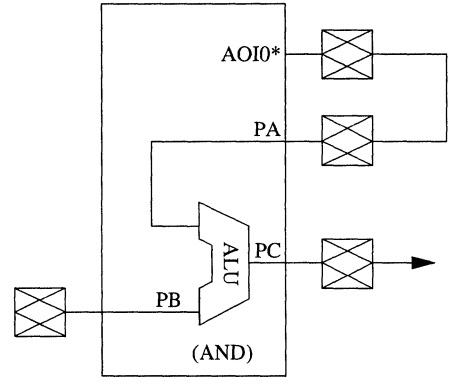
2.5.4.1 Switching ports using the AOIn* signal



PA and PB Set for Output

Image data for PA is selected in the area specified by the AOI0* signal; image data for PB is selected in other areas.

2.5.4.2 Operating on image data with the AOIn* signal (e.g., for masking)



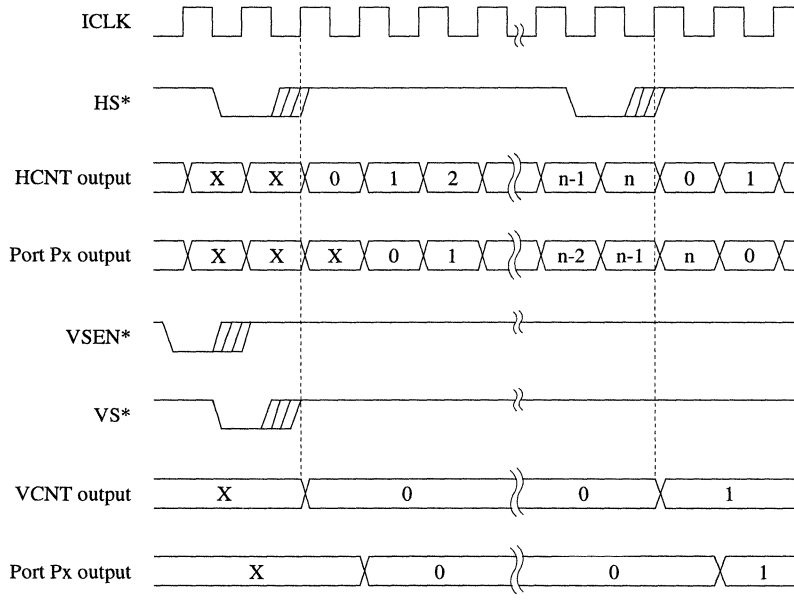
PA and PB Set for Input, PC Set for Output.

Inputs are AND'ed by the ALU, so that data is masked in the AOI-specified area and PC outputs 00h.

2.5.5 Delay from Internal Image Space Address Counter

Section 3.2, "Internal Configuration of Image Data Stream Switching Block," describes how to set flags Mm–Pm of port select registers PTSEL-M through PTSEL-P to 1 so that they output the values of the internal image space address counters (horizontal counter HCNT, vertical counter VCNT) to Mi–Pi, and also through a switch to external output ports PA–PL. (To reduce power consumption, the counters are disabled when all AOI areas are counted.)

These functions can be used to produce test signals to debug a board or system, or to generate gradation in a picture. They can also be combined with the ALUs to generate striped patterns, thus adding greater versatility to applications.



Note 1: HCNT is reset at the falling edge of HS* regardless of whether VS* is active.

Note 2: VCNT is reset at the falling edge of VS* regardless of whether HS* is active; that is, VCNT is reset when VS* and HS* are simultaneously asserted. Also, if HS* is pulsed low one or more clock cycles after VS* goes low, VCNT is incremented from 0 to 1.

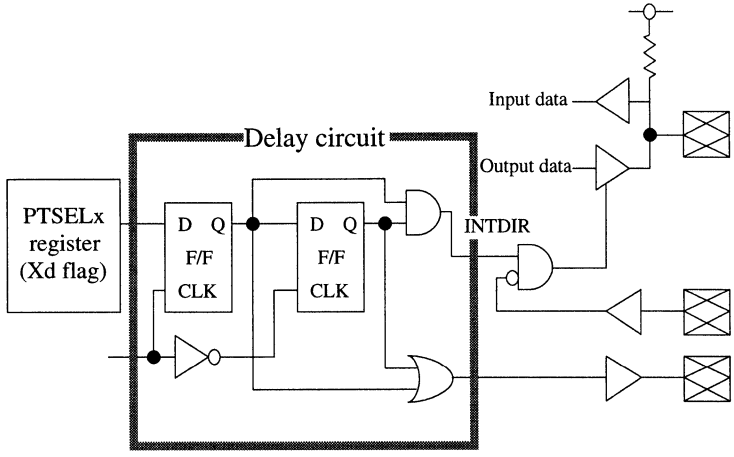
Note 3: Px is one of the image input/output ports PA–PL.

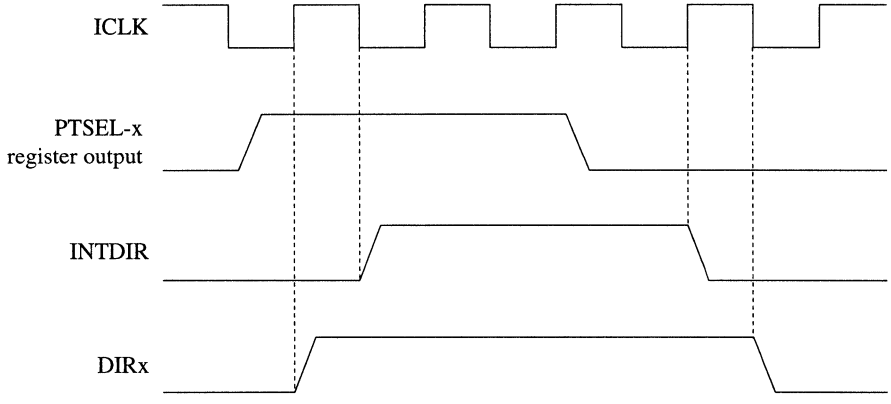
Note 4: Signal value X in the above diagram depends on the preceding status of the signal.

2.6 Controlling Image I/O Port Bidirectional Switching

2.6.1 Bus Conflict of Image Input/Output

The IP90C55's image input/output ports are protected against bus conflicts with external LSIs on the interface when the ports are switched between input and output.

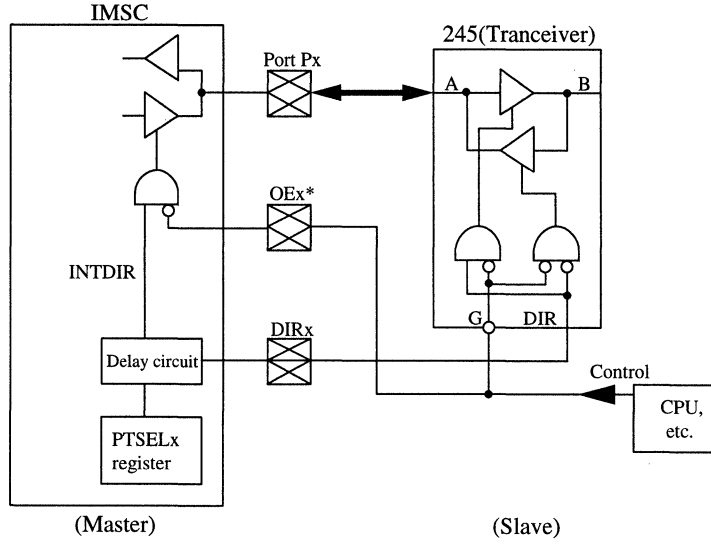




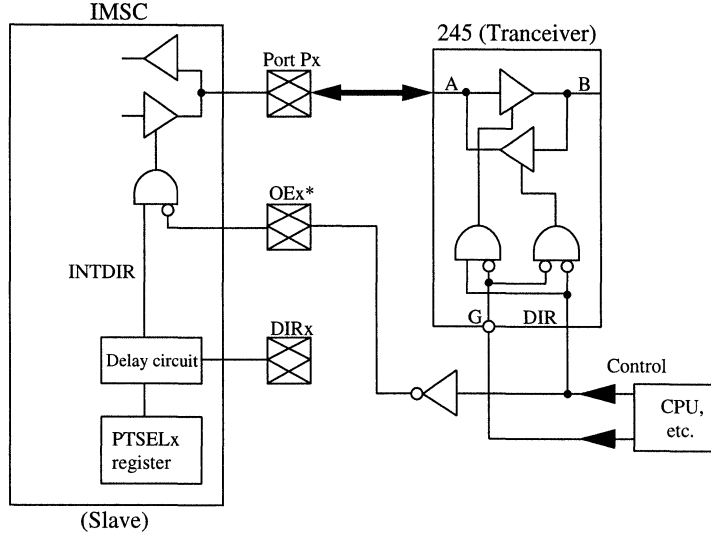
Note: Px represents one of the image input/output ports PA–PL. However, DIRx signals pins are not provided for ports PI–PL.

2.6.2 Sample Connections

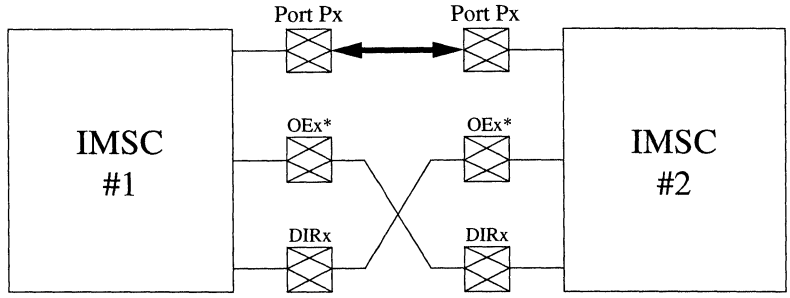
2.6.2.1 Using the IP90C55 as a Master and a Standard Logic Transceiver (74LS245) as a Slave



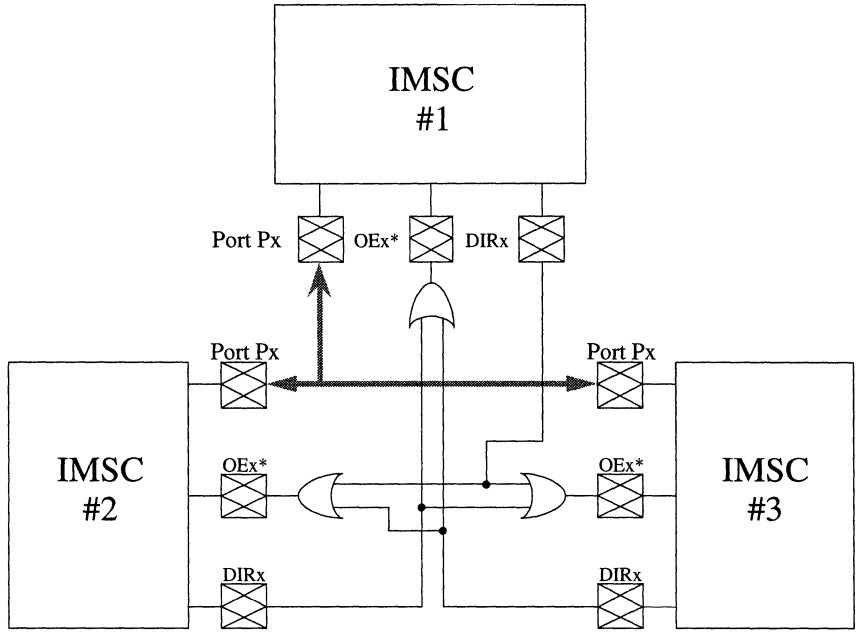
2.6.2.2 Using the IP90C55 as a Slave



2.6.2.3 Connecting Two IP90C55s Together



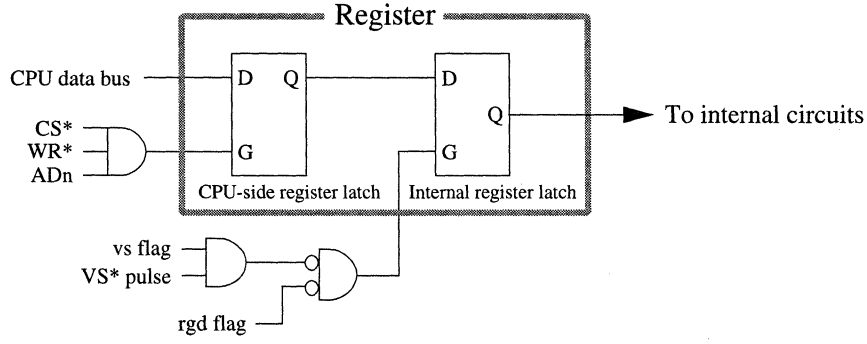
2.6.2.4 Connecting Three IP90C55s Together



2.6.3 Synchronized Port Switching Using the Vertical Synchronization Signal (VS*)

The IP90C55's registers are configured with dual latches.

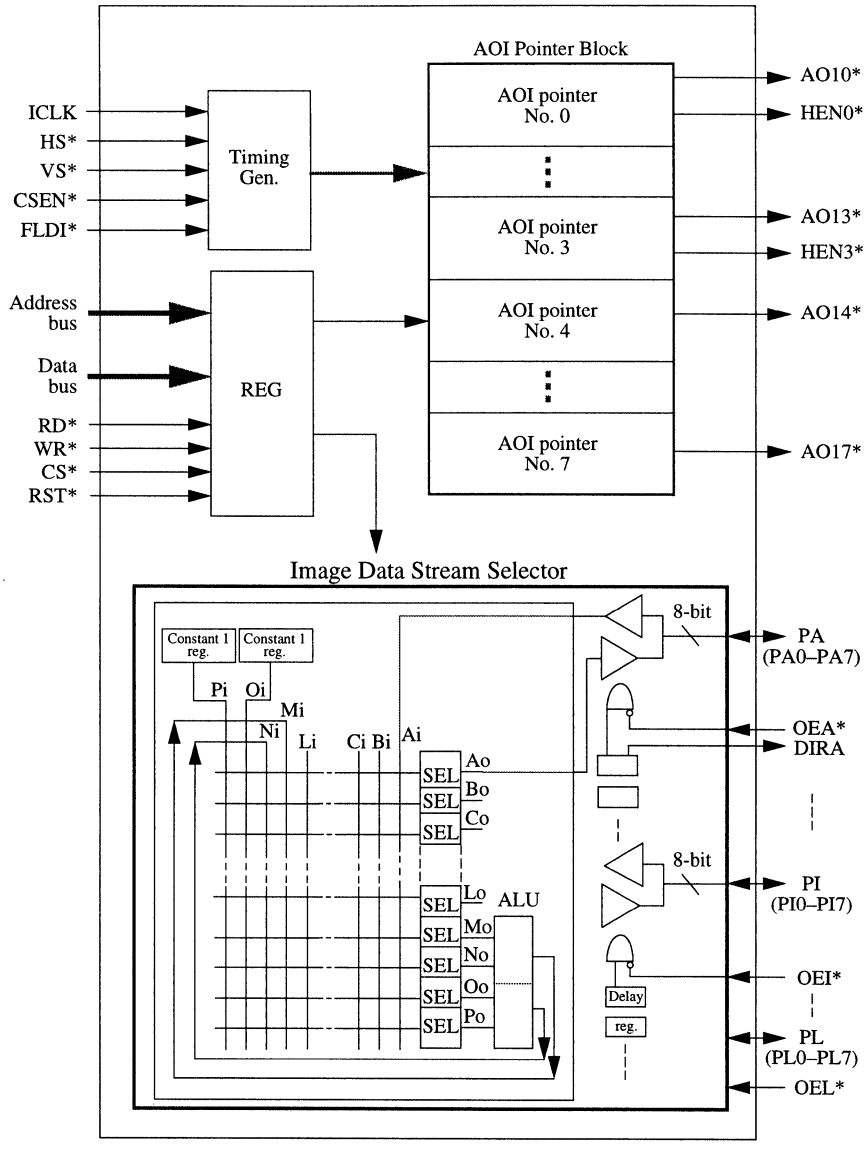
Data written from the CPU (host bus) to specify a port is first latched in the register latch on the CPU side. An internal register latch (a "through" latch) is provided between the internal circuit and the CPU-side register latch: the gate of this latch is opened (and the data latched) by a combination of three inputs: the REGVDIS register's rgd flag, the SMOD register's VS flag, and the VS* pulse (derived from the VS* field start signal by synchronizing it with the image clock).



Ports can be switched by fields by setting the REGVDIS register's RGD flag to 0 and the SMOD register's VS flag to 1. This enables the value to be set by the CPU (or PTSELO-x register for input/output port switching) at the beginning of the next field (i.e., at the rising edge of ICLK immediately after VS* is pulsed low).

Section 3: Internal Configuration

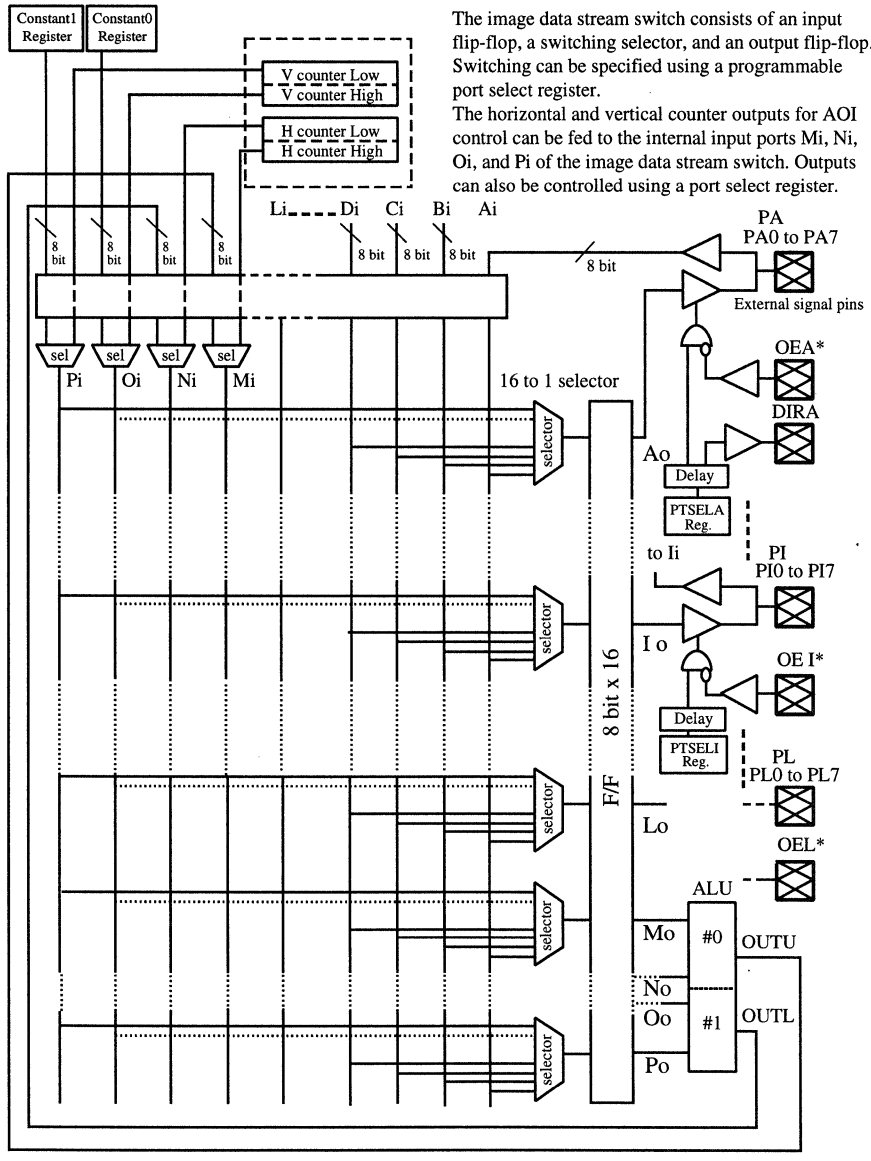
3.1 Internal Block Diagram



Note 1: An asterisk (*) following a signal name denotes inverse logic.

Note 2: This block diagram is shown only for functional description, and does not depict all of the IP90C55's functions. For details on its functions and timings, see the relevant sections in this manual.

3.2 Internal Configuration of the Image Data Stream Switching Block



The image data stream switch consists of an input flip-flop, a switching selector, and an output flip-flop. Switching can be specified using a programmable port select register.
The horizontal and vertical counter outputs for AOI control can be fed to the internal input ports Mi, Ni, Oi, and Pi of the image data stream switch. Outputs can also be controlled using a port select register.

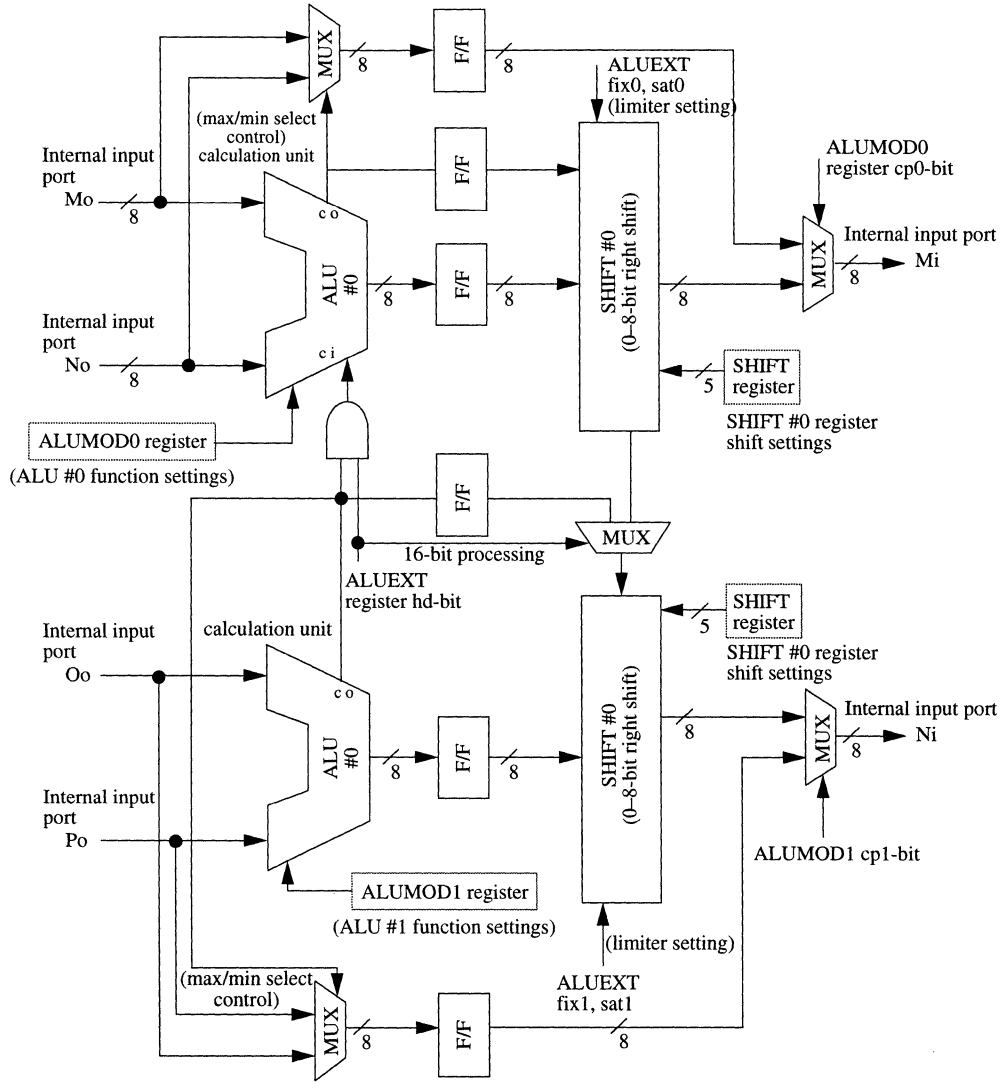
Note 1: An asterisk (*) following a signal name denotes inverse logic.

Note 2: This block diagram is shown only for functional description, and does not depict all the IP90C55's functions. For details on its functions and timings, see the relevant sections in this manual.

3.3 Internal Configuration of the ALU Block

The ALUs have these basic functions (equivalent to those of an 74LS181 processor):

- Arithmetic operations: addition, subtraction
- Logic operations: AND, OR, XOR, NOT
- Right shift, left shift, MIN/MAX selection, 8/16-bit process switching



Section 4: Pin Functions and Packages

4.1 Pin Assignment and Functional Description

Pin group	Symbol	No.	I/O	Function	Description
Image sync signals	ICLK	1	I	Image clock input	
	HS*	1	I	Horizontal sync signal input	
	VS*	1	I	Vertical sync signal input	
	VSEN*	1	I	Vertical sync enable signal input	VS* input is enabled when VSEN* is low (Note 2).
	FLDI	1	I	Field-identifying signal input	High = odd field, (used with non-interlaced) low = even field.
Image input/output bus	PA<7..0>	8	I/O	8-bit input/output port A	8-bit grayscale image data I/O
	PB<7..0>	8	I/O	8-bit input/output port B	
	PC<7..0>	8	I/O	8-bit input/output port C	These image data port pins have pull-up resistors.
	PD<7..0>	8	I/O	8-bit input/output port D	
	PE<7..0>	8	I/O	8-bit input/output port E	
	PF<7..0>	8	I/O	8-bit input/output port F	
	PG<7..0>	8	I/O	8-bit input/output port G	
	PH<7..0>	8	I/O	8-bit input/output port H	
	PI<7..0>	8	I/O	8-bit input/output port I	
	PJ<7..0>	8	I/O	8-bit input/output port J	
	PK<7..0>	8	I/O	8-bit input/output port K	
	PL<7..0>	8	I/O	8-bit input/output port L	
	OEA*	1	I	PA output enable	I/O pins (ports) are enabled for output in 8-bit units. Pins are set low for output or high for high-impedance status.
	OEB*	1	I	PB output enable	
	OEC*	1	I	PC output enable	
	OED*	1	I	PD output enable	
	OEE*	1	I	PE output enable	
	OEF*	1	I	PF output enable	
	OEG*	1	I	PG output enable	
	OEH*	1	I	PH output enable	
	OEI*	1	I	PI output enable	
	OEJ*	1	I	PJ output enable	
	OEK*	1	I	PK output enable	
	OEL*	1	I	PL output enable	
	DIRA	1	O	PA input/output direction	These indicate the I/O status of I/O signal pins (ports) in 8-bit units: a high signal indicates output, and a low signal indicates input. These signals can be used as external bus buffers or transceivers, or for bidirectional communication between IP90C55 chips. These pins represent ports PA–PH (no pins for PI–PL).
	DIRB	1	O	PB input/output direction	
	DIRC	1	O	PC input/output direction	
	DIRD	1	O	PD input/output direction	
	DIRE	1	O	PE input/output direction	
	DIRF	1	O	PF input/output direction	
	DIRG	1	O	PG input/output direction	
	DIRH	1	O	PH input/output direction	

Pin group	Symbol	No.	I/O	Function	Description
Image timing	AOI0*	1	O	AOI* output for AOI No. 0	Selects VEN0*, HEN0*, or ACT0*.
	HEN0*	1	O	HEN* output for AOI No. 0	
	AOI1*	1	O	AOI* output for AOI No. 1	Selects VEN1*, HEN1*, or ACT1*.
	HEN1*	1	O	HEN* output for AOI No. 1	
	AOI2*	1	O	AOI* output for AOI No. 2	Selects VEN2*, HEN2*, or ACT2*.
	HEN2*	1	O	HEN* output for AOI No. 2	
	AOI3*	1	O	AOI* output for AOI No. 3	Selects VEN3*, HEN3*, or ACT3*.
	HEN3*	1	O	HEN* output for AOI No. 3	
	AOI4*	1	O	AOI* output for AOI No. 4	AOI signal types can be HEN*, VEN*, or ACT*.
	AOI5*	1	O	AOI* output for AOI No. 5	
	AOI6*	1	O	AOI* output for AOI No. 6	
	AOI7*	1	O	AOI* output for AOI No. 7	
CPU bus	RD*	1	I	Read signal	
	WR*	1	I	Write signal	
	CS*	1	I	Chip select	
	AD<7..0>	1	I	Address bus	Register selection address.
	DB<7..0>	8	I/O	Data bus	
	RST*	1	I	Reset	Must be three clock cycles long or more (Note 3).
Power supply	Vdd	8	PW	5V	
GND	GND	23 (19)	PW	Ground	(19) is for the PGA 181.
Total number of pins		184 (180)			(180) is for the PGA 181. Pin 1 on the 181 is a locating (EXTRA) pin.

Packages: 184-pin QFP (mold section 32 x 32 mm^2, pin pitch 0.65 mm).
181-pin PGA (floor area 40 x 40 mm^2, pin pitch 2.54 mm)

Note that the EXTRA pin is a locating pin.

Note 1: An asterisk following a pin name indicates inverse logic.

Note 2: VS* can only be detected as low when the VSEN* signal is low one or more clock cycles before VS* goes low. Use VSEN* to synchronize fields using control signals (e.g., processing a start signal from a DSP) that are not synchronized with the image clock, or to establish field synchronization between image-processing circuit blocks. VSEN* must be held low in normal operation (i.e., when the pin's function is not required).

Note 3: Schmitt-trigger input, with a pull-up resistor.

4.2 Selecting a Package

The IP90C55 is available in 184-pin plastic QFP and 181-pin ceramic PGA packages. When deciding which package to use in a particular device, keep in mind the IP90C55's junction temperature limit. The device must be able to dissipate enough of the heat it generates to keep its internal temperature below the IP90C55's junction temperature limit. Otherwise, the device will not operate reliably.

As an approximate guide to selecting a package, the amount of heat the device generates can be estimated assuming the device operates as follow:

1. The heat generated (and power consumed) by CPU access to the internal registers is negligible compared to the heat generated by driving image data signals.
2. The average rate of change for image data is 50%.
3. Heat radiates under normal conditions (the device mounted on a board with no special heat sink).
4. Approximately 8 of the 12 ports on the image data bus are driven when image data is output (that is, eight ports consume power).
5. The output load capacitance is about 30 pF.

Under these assumptions, the plastic QFP can be used for image clock speeds up to 20 MHz, while the ceramic PGA can be used for clock speeds from about 20 MHz to 40 MHz. If the device will be used outside the above conditions, it must include a heat sink or air-cooling unit to increase heat radiation; even with a heat sink or cooling unit, however, the device still may not operate reliably.

If operating conditions differ from the above assumptions, see Section 4.3, "Estimating the Approximate Amount of Power Consumption." If the output pins have a large load capacitance, for example, the device's power consumption increases, and so must use a lower clock frequency. Conversely, if fewer output ports are driven, the device can operate at a higher clock frequency.

The ceramic PGA package has better heat radiation characteristics than the plastic QFP, and can therefore be used in high-performance, high-reliability systems. If the system requires wiring changes in prototype production, or if its power consumption or heat-radiation characteristics cannot be estimated, use the ceramic PGA.

Verify power consumption after the prototype is complete and mass production is under way, and then use the plastic QFP if the device meets the above conditions, or if the system will always operate at clock speeds of 20 MHz or less.

[References]

The IP90C55's operational limits with respect to temperature are actually determined by the temperature (technically known as the "junction temperature") at which the transistors can operate safely without reducing their performance and reliability. This temperature is estimated from the heat generated by the device's power consumption and its heat radiation characteristics, which are affected by the package material and heat radiation design features.

The heat generated is the sum of various heat-generating sources: power consumed for internal operation, which is proportional to the clock frequency and operating rate; the drive current on each output pin; the through current that flows when an intermediate voltage is applied to an input pin; the power consumed in each pull-up or pull-down resistor; and each transistor's leakage current.

The device's heat radiation characteristics are affected by the ambient temperature, air-cooling flow rate, presence of a heat sink, ability to radiate heat to the circuit board, and the shape and thermal resistance of the package material used.

4.3 Estimating the Approximate Amount of Power Consumption

A device's power consumption is determined by such factors as the number of output ports, operating frequency of the image clock (ICLK), load capacitance of output pins, and operating rate (change rate of image data signal).

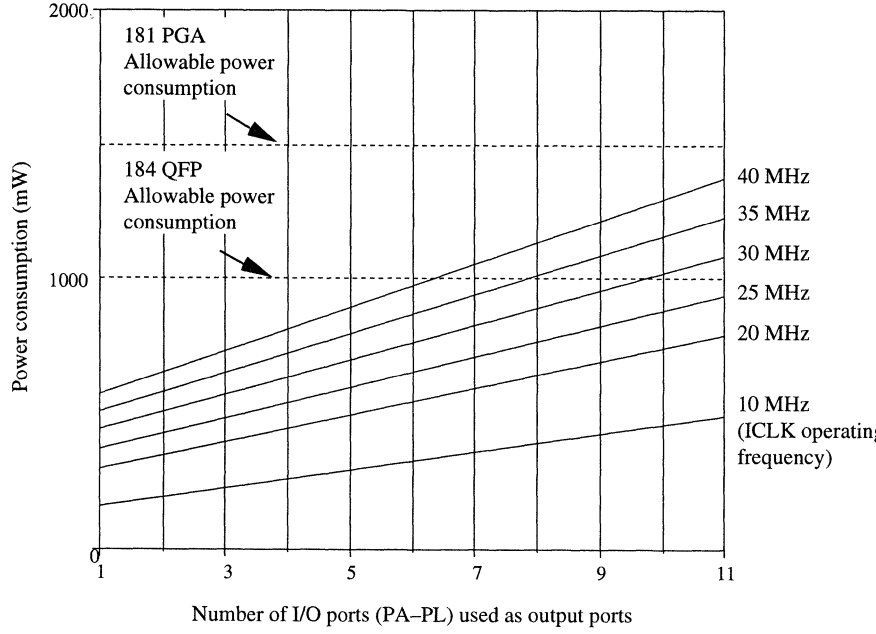
The graphs on the following pages show the approximate amount of power consumed by the IP90C55 under different output pin load capacitances: a) 30 pF, b) 40 pF, c) 50 pF, and d) 60 pF. Note, however, that this does not include the device's own pin capacitance of 10 pF. These approximations assume that the device is driven at 50% of the operating rate.

In cases where the blanking period is long and the average operating rate is low, or where the average operating rate is higher than 50% (as occurs with a clock's frequency-divided waveform), power consumption must be recalculated accordingly. The power consumed by access to the CPU buses is minimal and can be included in the calculation errors because the access frequency is much lower than the operating rate.

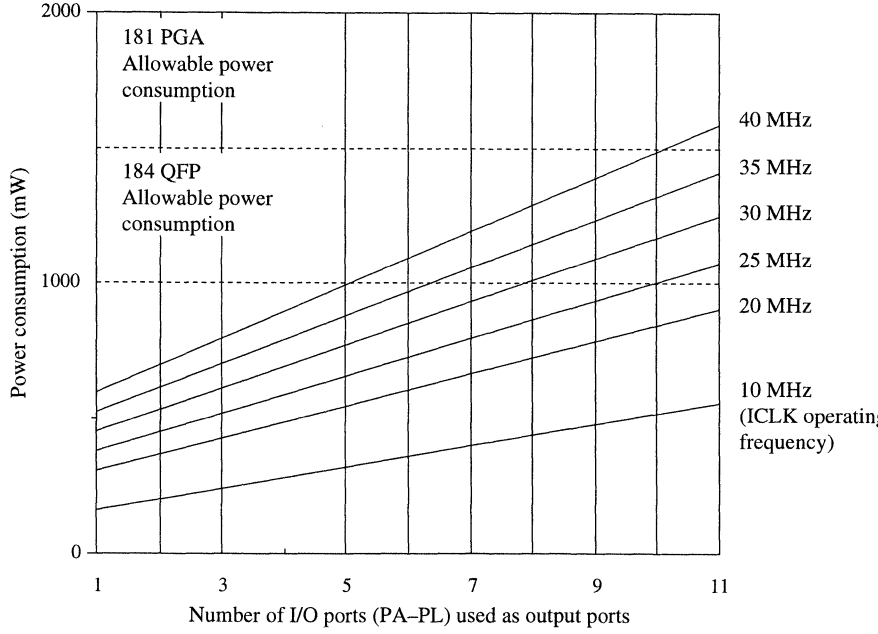
In special applications where registers are accessed to perform reads almost constantly and the CPU buses are frequently driven, the power consumed by CPU access can be considered equivalent to that of one image data port.

The 181-pin ceramic PGA and 184-pin plastic QFP packages have allowable limits of 1500 mW and 1000 mW, respectively. When using the devices, make sure these limits are not exceeded.

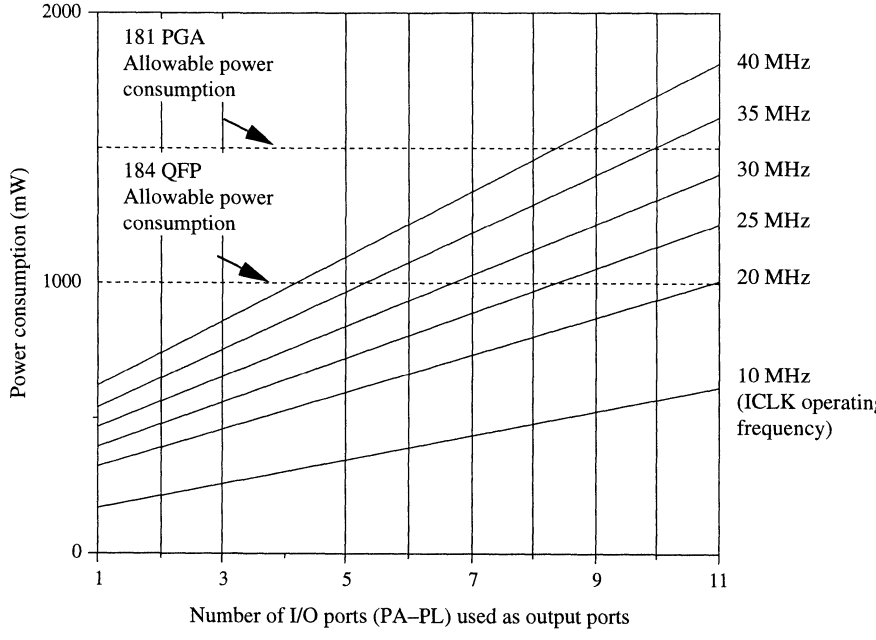
a) Output Pin Load Capacitance = 30 pF



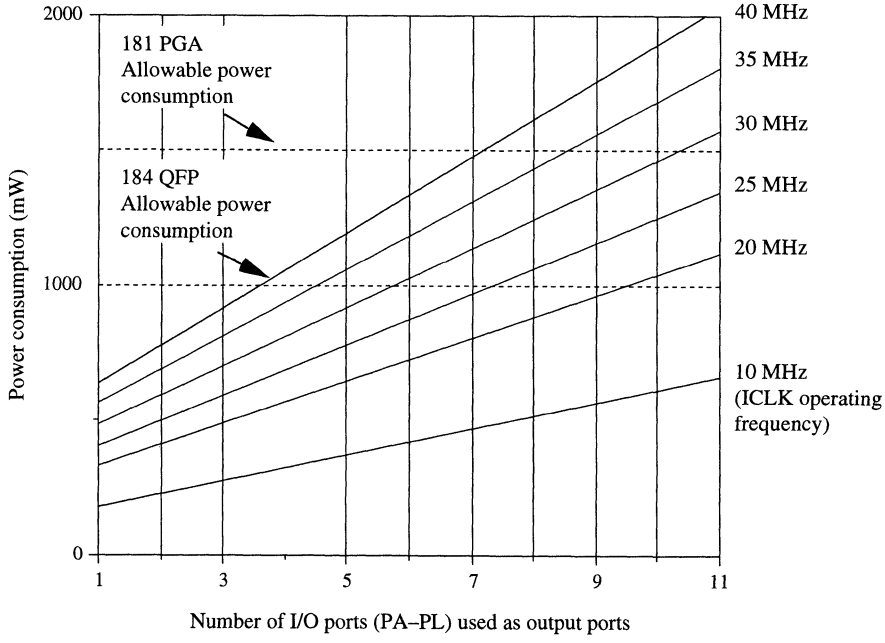
b) Output Pin Load Capacitance = 40 pF



c) Output Pin Load Capacitance = 50 pF



d) Output Pin Load Capacitance = 60 pF



4.4 Table of Pin Assignments for the 184-Pin Plastic QFP

No.	Symbol	I/O
1	GND	PW
2	OEA*	I
3	DIRA	O
4	PA0	I/O
5	PA1	I/O
6	PA2	I/O
7	PA3	I/O
8	PA4	I/O
9	PA5	I/O
10	PA6	I/O
11	PA7	I/O
12	GND	PW
13	OEB*	I
14	DIRB	O
15	PB0	I/O
16	PB1	I/O
17	PB2	I/O
18	PB3	I/O
19	PB4	I/O
20	PB5	I/O
21	PB6	I/O
22	GND	PW
23	Vdd	PW
24	GND	PW
25	PB7	I/O
26	OEC*	I
27	DIRC	O
28	PC0	I/O
29	PC1	I/O
30	PC2	I/O
31	PC3	I/O
32	PC4	I/O
33	PC5	I/O
34	PC6	I/O
35	GND	OW
36	PC7	I/O
37	OED*	I
38	DIRD	O
39	PD0	I/O
40	PD1	I/O

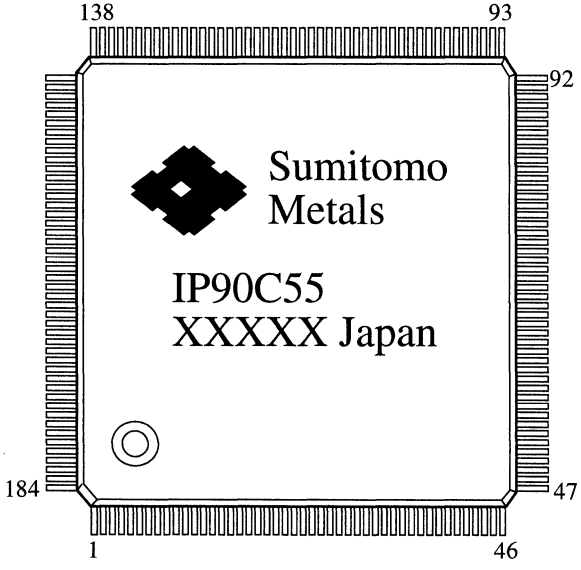
No.	Symbol	I/O
41	PD2	I/O
42	PD3	I/O
43	PD4	I/O
44	PD5	I/O
45	PD6	I/O
46	Vdd	PW
47	GND	PW
48	PD7	I/O
49	OEE*	I
50	DIRE	O
51	PE0	I/O
52	PE1	I/O
53	PE2	I/O
54	PE3	I/O
55	PE4	I/O
56	PE5	I/O
57	PE6	I/O
58	GND	PW
59	PE7	I/O
60	OEF*	I
61	DIRF	O
62	PF0	I/O
63	PF1	I/O
64	PF2	I/O
65	PF3	I/O
66	PF4	I/O
67	PF5	I/O
68	GND	PW
69	Vdd	PW
70	GND	PW
71	PF6	I/O
72	PF7	I/O
73	OEG*	I
74	DIRG	O
75	PG0	I/O
76	PG1	I/O
77	PG2	I/O
78	PG3	I/O
79	PG4	I/O
80	PG5	I/O

No.	Symbol	I/O
81	GND	PW
82	PG6	I/O
83	PG7	I/O
84	OEH*	I
85	DIRH	O
86	PH0	I/O
87	PH1	I/O
88	PH2	I/O
89	PH3	I/O
90	PH4	I/O
91	PH5	I/O
92	Vdd	PW
93	GND	PW
94	PH6	I/O
95	PH7	I/O
96	OEI*	I
97	PI0	I/O
98	PI1	I/O
99	PI2	I/O
100	PI3	I/O
101	PI4	I/O
102	PI5	I/O
103	GND	PW
104	PI6	I/O
105	PI7	I/O
106	OEJ*	I
107	PJ0	I/O
108	PJ1	I/O
109	PJ2	I/O
110	PJ3	I/O
111	PJ4	I/O
112	PJ5	I/O
113	PJ6	I/O
114	GND	PW
115	Vdd	PW
116	GND	PW
117	PJ7	I/O
118	OEK*	I
119	PK0	I/O
120	PK1	I/O

No.	Symbol	I/O
121	PK2	1/O
122	PK3	I/O
123	PK4	I/O
124	PK5	I/O
125	PK6	I/O
126	PK7	I/O
127	OEL*	I
128	GND	PW
129	PL0	I/O
130	PL1	I/O
131	PL2	I/O
132	PL3	I/O
133	PL4	I/O
134	PL5	I/O
135	PL6	I/O
136	PL7	I/O
137	GND	PW
138	Vdd	PW
139	GND	PW
140	AOI0*	O
141	HEN0*	O
142	AOI1*	O

No.	Symbol	I/O
143	HEN1*	O
144	GND	PW
145	AOI2*	O
146	HEN2*	O
147	AOI3*	O
148	HEN3*	O
149	GND	PW
150	AOI4*	O
151	AOI5*	O
152	AOI6*	O
153	AOI7*	O
154	GND	PW
155	VSEN*	I
156	VS*	I
157	HS*	I
158	FLDI	I
159	GND	PW
160	ICLK	I
161	Vdd	PW
162	GND	PW
163	RST*	I
164	CS*	I

No.	Symbol	I/O
165	WR*	I
166	RD*	I
167	AD0	I
168	AD1	I
169	AD2	I
170	AD3	I
171	AD4	I
172	AD5	I
173	AD6	I
174	AD7	I
175	GND	PW
176	DB0	I/O
177	DB1	I/O
178	DB2	I/O
179	DB3	I/O
180	DB4	I/O
181	DB5	I/O
182	DB6	I/O
183	DB7	I/O
184	Vdd	PW



4.5 Table of Pin Assignments for the 181-Pin Ceramic PGA

No.	Symbol	I/O
1	Vdd	PW
2	GND	PW
3	PD7	I/O
4	OEE*	I
5	DIRE	O
6	PE0	I/O
7	PE1	I/O
8	PE2	I/O
9	PH0	I/O
10	PH1	I/O
11	PH2	I/O
12	PH3	I/O
13	PH4	I/O
14	PH5	I/O
15	Vdd	PW
16	OEI*	I
17	PI0	I/P
18	PI1	I/O
19	PI2	I/O
20	PI3	I/O
21	PI4	I/O
22	PL2	I/O
23	PL3	I/O
24	PL4	I/O
25	PL5	I/O
26	PL6	I/O
27	PL7	I/O
28	GND	PW
29	Vdd	PW
30	GND	PW
31	AOI0*	O
32	HEN0*	O
33	AOI1*	O
34	HEN1*	O
35	AOI2*	O
36	HEN2*	O
37	DB2	I/O
38	DB3	I/O
39	DB4	I/O
40	DB5	I/O

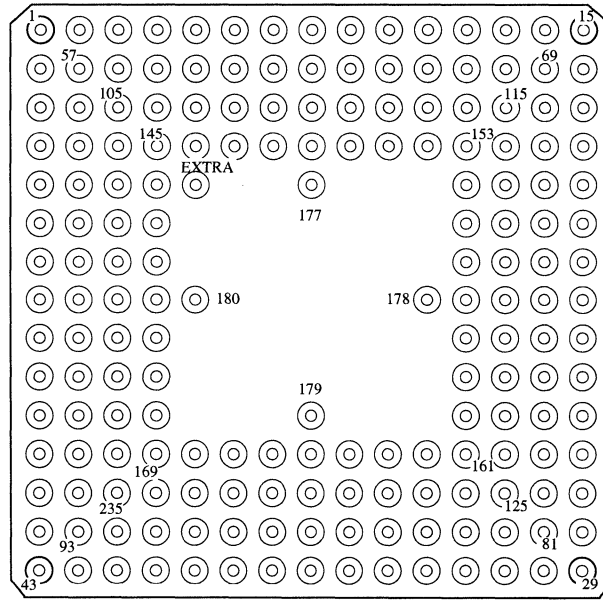
No.	Symbol	I/O
41	DB6	I/O
42	DB7	I/O
43	Vdd	PW
44	PA0	I/O
45	PA1	I/O
46	PA2	I/O
47	PA3	I/O
48	PA4	I/O
49	PA5	I/O
50	PD0	I/O
51	PD1	I/O
52	PD2	I/O
53	PD3	I/O
54	PD4	I/O
55	PD5	I/O
56	PD6	I/O
57	DIRD	O
58	PE3	I/O
59	PE4	I/O
60	PE5	I/O
61	PE6	I/O
62	GND	PW
63	PE7	I/O
64	PG5	I/O
65	GND	PW
66	PG6	I/O
67	PG7	I/O
68	OEH*	I
69	DIRH	O
70	PI5	I/O
71	GND	PW
72	PI6	I/O
73	PI7	I/O
74	OEJ*	I
75	PK5	I/O
76	PK6	I/O
77	PK7	I/O
78	OEL*	I
79	GND	PW
80	PL0	I/O

No.	Symbol	I/O
81	PL1	I/O
82	AOI3*	O
83	HEN3*	O
84	GND	PW
85	AO14*	O
86	AOI5*	O
87	AOI6*	O
88	AD5	I
89	AD6	I
90	AD7	I
91	GND	PW
92	DB0	I/O
93	DB1	I/O
94	PA6	I/O
95	PA7	I/O
96	GND	PW
97	OEB*	I
98	DIRB	O
99	PC4	I/O
100	PC5	I/O
101	PC6	I/O
102	GND	PW
103	PC7	I/O
104	OED*	I
105	PC3	I/O
106	OEF*	I
107	DIRF	O
108	PF0	I/O
109	PF1	I.O
110	PF2	I/O
111	PG0	I/O
112	PG1	I/O
113	PG2	I/O
114	0G3	I/O
115	PG4	I/O
116	PJ0	I/O
117	PJ1	I/O
118	PJ2	I/O
119	PJ3	I/O
120	OEK*	I

No.	Symbol	I/O
121	PKO	1/O
122	PK1	I/O
123	PK2	I/O
124	PK3	I/O
125	PK4	I/O
126	AOI7*	O
127	GND	PW
128	VSEN*	I
129	VS*	I
130	HS*	I
131	AD0	I
132	AD1	I
133	AD2	I
134	AD3	I
135	AD4	I
136	PB0	I/O
137	PB1	I/O
138	PB2	I/O
139	PB3	I/O
140	OEC*	I
141	DIRC	O

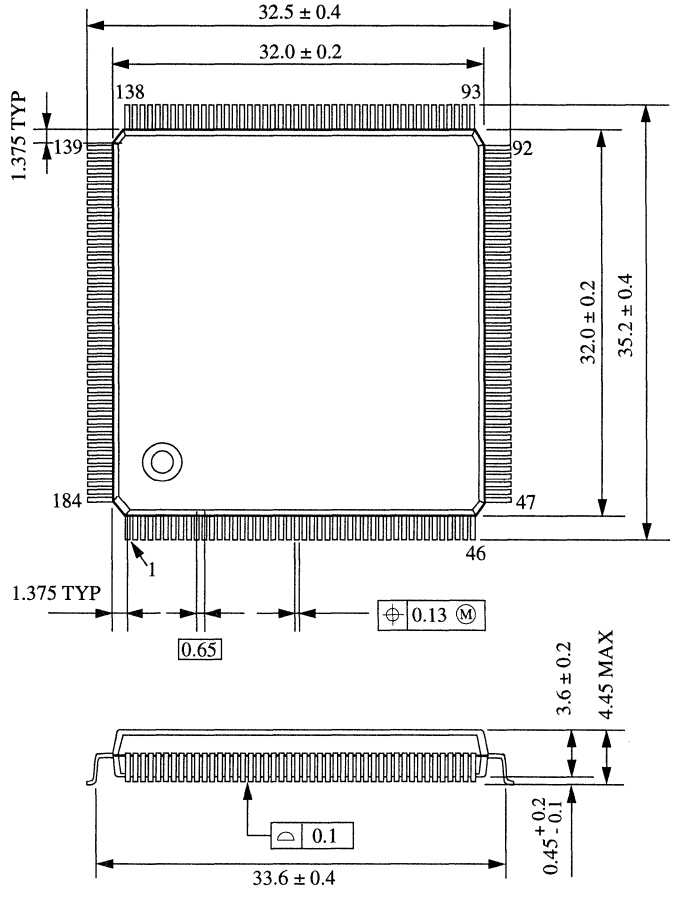
No.	Symbol	I/O
142	PC0	I/O
143	PC1	I/O
144	PC2	I/O
145	PB7	I/O
146	PF3	I/O
147	PF4	I/O
148	PF5	I/O
149	Vdd	PW
150	PF6	I/O
151	PF7	I/O
152	OEG*	I
153	DIRG	O
154	GND	PW
155	PH6	I/O
156	PH7	I/O
157	PJ4	I/O
158	PJ5	I/O
159	PJ6	I/O
160	Vdd	PW
161	PJ7	I/O

No.	Symbol	I/O
162	FLDI	I
163	GND	PW
164	ICLK	I
165	Vdd	PW
166	RST*	I
167	CS*	I
168	WR*	I
169	RD*	I
170	GND	PW
171	OEA*	I
172	DIRA	O
173	PB4	I/O
174	PB5	I/O
175	PB6	I/O
176	Vdd	PW
177	GND	PW
178	GND	PW
179	GND	PW
180	GND	PW
Pin 181 is an extra pin.		

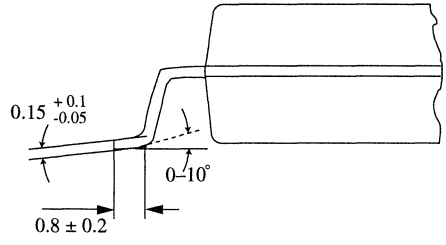


Bottom View

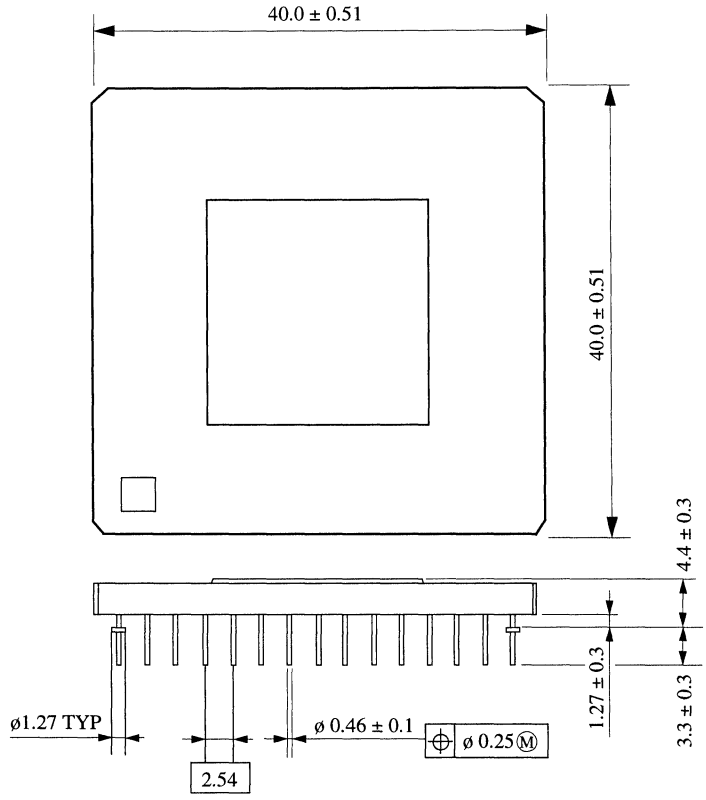
4.6 Outline Dimensions of the 184-Pin Plastic QFP



units: mm



4.7 Outline Dimension of the 181-Pin Ceramic PGA



Bottom View

unit: mm

Section 5: Registers

5.1 Table of Registers (Address Map)

All registers are read/write except the wrf flag of the SMOD register.

Address (hex)	Abbreviation		Width in bits	Functional outline
00 01	AOI-0	HSTART-n (Low) (High)	8 8	AOI-n odd/even image data Start point of area of interest in horizontal direction
02 03		HEND-n (Low) (High)	8 8	AOI- odd/even image data End point of area of interest in horizontal direction
04 05		VSTARTod-n (Low) (High)	8 8	AOI-n odd image data Start point of area of interest in vertical direction
06 07		VENDod-n (Low) (High)	8 8	AOI-n odd image data End point of area of interest in vertical direction
08 09	where n = 0	VSTARTev-m (Low) (High)	8 8	AOI-n even image data Start point of area of interest in vertical direction
0A 0B		VENDev-n (Low) (High)	8 8	AOI-even image data End point of area of interest in vertical direction
0C-0E		(Reserved)		AOI-n even image data
0F	(Note 1)	AOIMOD-n	7	Chooses the AOI-n signal and output waveform
10–1F	AOI-1	Similar to AOI-0, where n = 1	—	AOI-1 pointer register set
20–2F	AOI-2	Similar to AOI-0, where n = 2	—	AOI-2 pointer register set
30–3F	AOI-3	Similar to AOI-0, where n = 3	—	AOI-3 pointer register set
40–4F	AOI-4	Similar to AOI-0, where n = 4	—	AOI-4 pointer register set
50–5F	AOI-5	Similar to AOI-0, where n = 5	—	AOI-5 pointer register set
60–6F	AOI-6	Similar to AOI-0, where n = 6	—	AOI-6 pointer register set
70 -7F	AOI-7	Similar to AOI-0, where n = 7	—	AOI-7 pointer register set
80–82	ZEROSTART		—	Sets the AOI zero-start flag
83–8F	(Reserved)		—	(Unused address)
90–9F	PTSEL-A to P		5	Port-select register
A0, A1	CONST0, CONST1		8	Constant registers (Oi, Pi ports)
A2, A3	ALUMOD0, ALUMOD1		8	ALU function registers
A4	ALUEXT		5	ALU extended function register
A5, A6	SHIFT0, SHIFT1		5	Right bit shifter shift-setting registers
A7–FB	(Reserved)		—	(Unused address)
FC	TESTM		—	Test the device's internal logic
FD	REGVDIS		1	Register data disable
FE	SMOD		2	Sub-mode register
FF	MMOD		2	Main mode register

Note 1: The IP90C55 has eight pairs of AOI register sets: these are represented above by AOI-0, though the information also applies to register sets AOI-1 through AOI-7.

5.2 AOI Pointer Register Sets

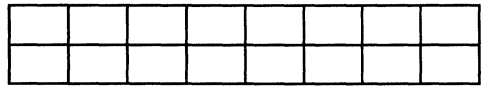
These register sets specify the coordinates of the area of interest. Coordinates in the vertical and horizontal directions are each specified with 16 bits. The IP90C55 has eight pairs of AOI pointer register sets.

Note: The AOI pins (HEN-n*, AOI-n*) corresponding to each of the AOI pointer register sets have a numerical suffix 0–7. In the following descriptions, a given AOI pointer register set is represented by a suffix "n," where n equals 0, 1, 2, 3, 4, 5, 6, or 7. The suffix "n" is attached to the corresponding AOI pin (HEN-n*, AOI-n*) for the given AOI pointer register set. For example, when specifying coordinates for AOI-3 (HEN3*, AOI3*), the starting coordinate in the horizontal direction is HSTART-3, and is allocated at addresses 30h and 31h.

HSTART-n odd/even screen

Horizontal area of interest start-point setting register.

Specifies the AOI's first horizontal coordinate value-1.



Reg. name	Address
HSTART-n low	n0h
HSTART-n high	n1h

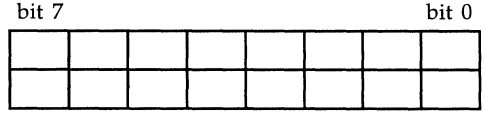
The data entering at the rising edge of ICLK immediately after HS* is detected low by a rising edge of ICLK is named D(0), and successive data is sequentially named D(1), D(2), D(3), ..., D(m). If N-1 is set in the HSTART-n register, the AOI begins with data D(N).

However, if the area of interest to is to begin from data D(0), set bit-n in HZSTART (80h) to 1. In this case, the value of the HSTARET-n register is ignored.

HEND-n odd/even screen

Horizontal area of interest end-point setting register.

Specifies the AOI's last horizontal coordinate.



Reg. name	Address
HEND-n low	n2h
HEND-n high	n3h

The data entering at the rising edge of ICLK immediately after HS* is detected low by a rising edge of ICLK is named D(0), and successive data is sequentially named D(1), D(2), D(3), ..., D(m). If N is set in the HEND-n register, the AOI is valid up to data D(N) and becomes invalid beginning with data D(N+1).

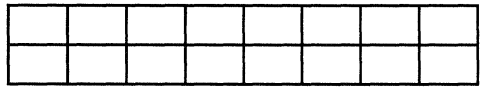
VSTARTod-n odd screen

Vertical area of interest start-point setting register.

For interlaced operation, this register specifies the AOI's first vertical coordinate value-1 in an odd field.

For non-interlaced operation, this register specifies the AOI's first vertical coordinate value-1.

This register value is used when the FLDI input is high.



bit 7							bit 0	Reg. name	Address
								VSTARTod-n low	n4h
								VSTARTod-n high	n5h

The first vertical coordinate determines the number of high-to-low transitions of HS* pulses counted by the rising edge of ICLK after VS* is asserted low. For example, if M-1 is set in the VSTARTod-n register, low HS* pulses occurring after VS* is asserted low are counted as the 1st, 2nd, 3rd, ..., and M-1th lines, with the AOI beginning from the Mth line of data.

If VS* and HS* are simultaneously asserted low at the rising edge of ICLK, the HS* pulses are counted as the 0th, 1st, 2nd, 3rd, ..., and M-1th.

If the AOI is to begin from the 0th line of data, set bit-n in the VOZSTART register (81h) to 1.

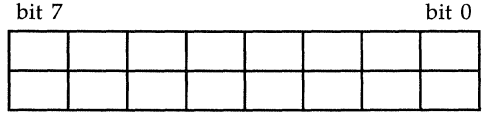
VENDod-n odd screen

Vertical AOI end-point setting register.

For interlaced operation, this register specifies the AOI's last vertical coordinate in an odd field.

For non-interlaced operation, this register specifies the AOI's last vertical.

This register is used when the FLDI input is high.



bit 7							bit 0	Reg. name	Address
								VENDod-n low	n6h
								VENDod-n high	n7h

The last vertical coordinate determines the number of high-to-low transitions of HS* pulses counted by the rising edge of ICLK after VS* is asserted low. For example, if M is set in the VENod-n register, low HS* pulses occurring after VS* is asserted low are counted as the 1st, 2nd, 3rd, ..., and M-1th lines, with the AOI becoming invalid beginning from the M+1th line of data.

If VS* and HS* are simultaneously asserted low at the rising edge of ICLK, the HS* pulses are counted as the 0th, 1st, 2nd, 3d, ..., and M-1th.

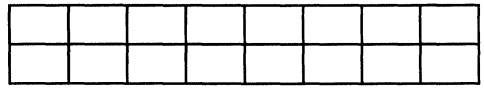
VSTARTev-n even screen

Vertical AOI start-point setting register.

For interlaced operation, this register specifies the AOI's first vertical coordinate value-1 in an even field.

For non-interlaced operation, this register is ignored.

This register is used when the FLDI input is low.



bit 7							bit 0	Reg. name	Address
								VSTARTev-n low	n8h
								VSTARTev-n high	n9h

If VS* and HS* are simultaneously asserted low at the falling edge of ICLK, the HS* pulses are counted as the 0th, 1st, 2nd, 3rd, ..., and M-1th lines. To start the AOI from the 0th line of data, set bit-n in VEZSTART (82h) to 1.

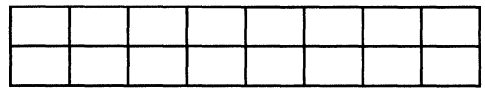
VENDev-n even screen

Vertical AOI end-point setting register.

For interlaced operation, this register specifies the AOI's last vertical coordinate in an odd field.

For non-interlaced operation, this register is ignored.

This register value is used when the FLDI input is low.



bit 7							bit 0	Reg. name	Address
								VENDev-n low	nAh
								VENDev-n high	nBh

AOI mode register

AOIMOD-n (n = 0, 1, 2, ..., 7)

This bit selects the waveform (Level, Single, Double, or Corner) of the AOI signal output from the AOIn* signal pin, as well as the type of the AOI signal (HEN*, VEN*, ACT*). The setting also controls whether AOI signal output is enabled, or output remains at high level.

The HENn* (n=0, 1, 2, 3) signal pins carry only those output signals that can be output from the AOIn* (n=0, 1, 2, 3) signal pins.

bit 7	bit 6	bit 5	bit 2	bit 0	Reg. name	Address
0	AOIexe-n	AOIp-n		AOIs-n	AOIMOD-n	nFh, 1Fh, 2Fh, 3Fh, 4Fh, 5Fh, 6Fh, 7Fh

AOIs-n: AOI signal-select flag

When n = 0, 1, 2, 3:

This flag chooses VEN*, HEN*, or ACT* for output from the AOIn* pin. For output from the HENn* pin, however, HEN* is selected directly. Thus, the HENn* pins output HEN* regardless of whether VEN* or ACT* is selected as output from the AOIn* pins.

When n = 4, 5, 6, 7:

This flag chooses HEN*, VEN*, or ACT* for output from the AOIn* pin.

AOIp-n: AOI waveform-select flag

The waveform of the AOI signal output from the AOIn* signal pins. This flag chooses the AOI waveform from Level, Single, Double, and Corner. It also chooses between 1-field[[-only?]] or 2-field-only output. The waveform of the HEN* signal output from the HENn* (n=0, 1, 2, 3) pins is the wave form selected for the AOIn* (n=0, 1, 2, 3) pins. Also, the "Corner pulse" waveform is only enabled when ACT* is selected. Thus, if the Corner pulse waveform is selected for the AOIn* pins, the output from the HENn* pins is undefined.

AOIexe-n: AOI output enable flag

This flag enables AOI signal output from the AOIn* signal pins. When output from the AOIn* pins is enabled or held at high level, output from the HENn* pins is also either enabled or held at high level, respectively.

If AOIexe-n = 0, AOI is not output. It remains high.

If AOIexe-n = 1, AOI is output beginning with the field next to the one in which this flag bit is set to 1.

0: Do not use this bit. Leave it set to 0.

AOIs-n		AOI signals output	
bit 1	bit 0	(Note 1)	Remarks
0	0	Inactive	
0	1	HEN*	Horizontal AOI signal
1	0	VEN*	Vertical AOI signal
1	1	ACT* (Note 2)	AOI signal

Note 1: The HENn* signal pins output only the HEN* signal.

Note 2: ACT = (HEN • VEN).

AOIp-n					Output	
bit 5	bit 4	bit 3	bit 2	Output mode	waveform	Remarks
0	X	0	0	Continuous every field	Level	The output waveform described on the left is output every field
0	X	0	1		Single Pulse	
0	X	1	0		Double Pulse	
0	X	1	1		Corner Pulse (Note 2)	
1	0	0	0	2-field only	Level	Use this setting to output AOI for only one screen (1 or 2 fields) immediately after the field in which AOIexe-n is set to 1. For interlaced operation, one screen (i.e., one frame) consists of two fields.
1	1	0	0	1-field only		
1	0	0	1	2-field only	Single Pulse	
1	1	0	1	1-field only		
1	0	1	0	2-field only	Double Pulse	
1	1	1	0	1-field only		
1	0	1	1	2-field only	Corner Pulse	
1	1	1	1	1-field only	(Note 2)	

After 1- or 2-field output is complete, AOI can be output every field or for either one or two fields by clearing AOIexe-n to 0 before setting it to 1 again.

Combinations of AOI Output Signals and Waveforms

Figures 1–10 are in Section 2.4, "Example of Outputting Signals to Specify Rectangular Image Areas."

Output waveform (bit 3, bit 2)		Level	Single Pulse	Double Pulse	Corner Pulse
Output signal (bit 1, bit 0)		00	01	10	11
None	00	High	Undefined (Note 3)	Undefined (Note 3)	Undefined (Note 3)
HEN*	01	Figure 1	Figure 4	Figure 7	Undefined (Note 3)
VEN*	10	Figure 2	Figure 5	Figure 8	Undefined (Note 3)
ACT* (Note 2)	11	Figure 3	Figure 6	Figure 9	Figure 10 (Note 4)

Note 1: An asterisk (*) at the end of a signal name indicates inverse logic.

Note 2: Only when ACT* is selected as the AOI signal.

Note 3: ACT = (VEN • HEN).

Note 4: If described as undefined, values are not guaranteed.

Note 5: This pulse is output at the first falling edge of HEN* after VEN* is asserted, and again at the first rising edge of HEN* after VEN* is deasserted.

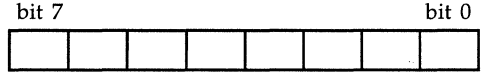
5.3 Setting AOI Zero-Start Flag

Set the eight sets of start coordinates for an AOI to begin from the 0th pixel or 9th line.

HZSTART odd/even screen data

Horizontal AOI start point zero-setting flag set.

Specifies 0 as the AOI's first horizontal coordinate.



Reg. name	Address
HZSTART	80h

bit-n = 0: The first horizontal coordinate of AOI-n (AOI) is the value set by the HSTART-n register.

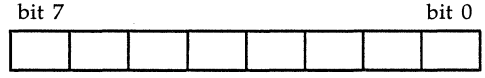
bit-n = 1: The first horizontal coordinate of AOI-n (AOI) begins from the 9th pixel, and the value of the HSTART-n register is ignored.

VOZSTART odd screen

Vertical AOI start point zero-setting flag set.

For interlaced operation, this specifies 0 as the AOI's first vertical coordinate in an odd field.

For non-interlaced operation, this specifies 0 as the AOI's first vertical coordinate.



Reg. name	Address
VOZSTART	81h

bit-n = 0: The first vertical coordinate of AOI-n (AOI) is the value set by the VSTARTod-n register.

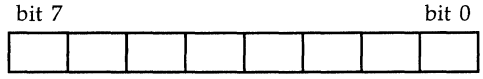
bit-n = 1: If VS* and HS* are simultaneously asserted low at the rising edge of ICLK, the HS* pulses are counted as the 0th, 1st, 2nd, ..., and M-1th lines, and the first vertical coordinate of AOI-n (AOI) begins from the 0th line.

VEZSTART even screen

Vertical AOI start point zero-setting flag set.

For interlaced operation, specifies 0 as the AOI's first vertical coordinate in an even field.

For non-interlaced operation, this register is ignored.



Reg. name	Address
VEZSTART	82h

bit-n = 0: The first vertical coordinate of AOI-n (AOI) is the value set by the VSTARTev-n register.

bit-n = 1: If VS* and HS* are simultaneously asserted low at the falling edge of ICLK, the HS* pulses are counted as the 0th, 1st, 2nd, ..., and M-1th lines, and the first vertical coordinate of AOI-n (AOI) begins from the 0th line.

5.4 Port-Select Registers

PTSEL-x (x = A, B, ..., P)

The IP90C55's internal output ports Ao–Po (8 bits each) are connected to its internal input ports Ai–Pi (also 8 bits each). These registers specify the internal port and output ports through which data is fed.

Note: PA represents external I/O pins PA0–PA7 (similarly for ports PB–PL).

Ai represents internal input ports Ai0–Ai7 (similarly for ports Bi–Pi).

Ao represents internal output ports Ao0–Ao7 (similarly for ports Bo–Po).

Each 8-bit input port Ai–Li is connected to the input side of the 8-bit external input/output pins PA–PL (that is, input port #i is connected to the input side of external I/O pin p#, where # denotes A, B, C, ..., L.) Similarly, each 8-bit output port Ao–Lo is connected to the output side of the 8-bit external input/output pins A–L.

bit 7			bit 4	bit 0	Reg. name	Address
0	0	0	Ad	PSELAo	PTSEL-A	90h
0	0	0	Bd	PSELBo	PTSEL-B	91h
0	0	0	Cd	PSELCo	PTSEL-C	92h
0	0	0	Dd	PSELDo	PTSEL-D	93h
0	0	0	Ed	PSELEo	PTSEL-E	94h
0	0	0	Fd	PSELFo	PTSEL-F	95h
0	0	0	Gd	PSELGo	PTSEL-G	96h
0	0	0	Hd	PSELHo	PTSEL-H	97h
0	0	0	Id	PSELIo	PTSEL-I	98h
0	0	0	Jd	PSELJo	PTSEL-J	99h
0	0	0	Kd	PSELKo	PTSEL-K	9Ah
0	0	0	Ld	PSELLo	PTSEL-L	9Bh
Mm	0	0	0	PSELMo	PTSEL-M	9Ch
Nm	0	0	0	PSELNo	PTSEL-N	9Dh
Om	0	0	0	PSELOo	PTSEL-O	9Eh
Pm	0	0	0	PSELPo	PTSEL-P	9Fh

PSELxo: Port-select flag

Bit 4 specifies the internal input port from which data is output.

xd: External image I/O pin input/output mode setting flag

Sets the Px0–Px7 8-bit external input/output pins for input or output.

xd = 0: Px0–Px7 are set for input

xd = 1: Px0–Px7 are set for output

where x denotes A, B, ..., L.

For example, if set to Gd = 0 and Hd = 1, external image pins PG0–PG7 would be set for input, and pins PH0–PH7 would be set for output.

The flag is cleared to 0 when reset, and so all image data I/O pins are set for input. This is a protective measure to prevent signal collisions at output pins when the device is reset.

Mm to Pm: Input port data-select flags

This flag selects the input data for internal input ports Mi, Ni, Oi, and Pi:

Mm = 0: Sets the output of ALU#0 to the Mi input port.

Mm = 1: Sets the value of the upper byte (HCNTH) of the horizontal pixel counter HCNT to the Mi input port.

Nm = 0: Sets the output of ALU#1 to the Ni input port.

Nm = 1: Sets the value of the lower byte (HCNTL) of the horizontal pixel counter HCNT to the Ni input port.

Om = 0: Sets the CONST0 of the constant register to the Oi input port.

Om = 1: Sets the value of the upper byte (VCNTH) of the vertical pixel counter VCNT to the Oi input port.

Pm = 0: Sets the CONST1 of the constant register to the Pi input port.

Pm = 1: Sets the value of the lower byte (VCNTL) of the vertical pixel counter VCNT to the Pi input port.

Note: Mo and No are connected to the input of ALU#0; Oo and Po are connected to the input of ALU#1.

Internal I/O ports Mi, Mo, Ni, No, Oi, Oo, Pi, and Po are not connected to the IP90C55's external input/output pins.

0: Not used. Write the value 0.

PSELxo bits 0–3	Selected input port
0000	Input signal from Ai is selected.
0001	Input signal from Bi is selected.
0010	Input signal from Ci is selected.
0011	Input signal from Di is selected.
0100	Input signal from Ei is selected.
0101	Input signal from Fi is selected.
0110	Input signal from Gi is selected.
0111	Input signal from Hi is selected.
1000	Input signal from Ii is selected.
1001	Input signal from Ji is selected.
1010	Input signal from Ki is selected.
1011	Input signal from Li is selected.
1100	Input signal from Mi is selected.
1101	Input signal from Ni is selected.
1110	Input signal from Oi is selected.
1111	Input signal from Pi is selected.

However, if ports Ao–Lo are specified so that the input-to-output path specification is looped, data FFh is set to the output port.

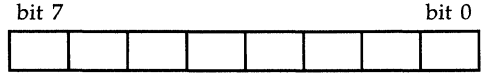
For example, if input port Ai is specified for output port Ao, Ao is set to FFh.

5.5 Constant Registers (Oi and Pi ports)

These registers set constants to the Oi and Pi ports.

CONST0 Oi port constant register

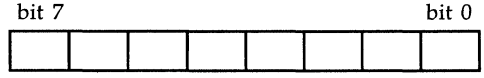
This register sets the Oi input port to a constant value.



Reg. name	Address
CONST0	A0h

CONST1 Pi port constant register

This register sets the Pi input port to a constant value.



Reg. name	Address
CONST1	A1h

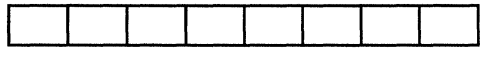
5.6 ALU Function Registers

These registers set the functions of the two 8-bit ALU (arithmetic/logic) units. The ALUs functions are equivalent to those of the 74LS181 processor.

ALUMOD0 ALU#0 function register

ALU#0 receives input from the Mo and No output ports, and sends its output to the Mi input port (when PTSEL-M bit 7: Mm = 0) through a right bit shifter (SHIFT#0). (If cp0 = 1, i.e., selection between large and small, the ALU#0 output is not routed through SHIFT#0.)

This register sets the arithmetic processing function of ALU#0.

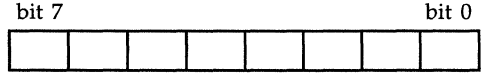


Reg. name	Address
ALUMOD0	A2h

ALUMOD1 ALU#1 function register

ALU#1 receives input from the Oo and Po output ports, and sends its output to the Ni input port (when PTSEL-Nbit7: Nm = 0) through a right bit shifter (SHIFT#1). (If cp1 = 1, i.e., selection between large and small, the ALU#1 output is not routed through SHIFT#1.)

This register sets the arithmetic processing function of ALU#1.



Reg. name	Address
ALUMOD1	A3h

fs0, 1: ALU function flag
Sets the basic function.

fm0, 1: Arithmetic/logic switching flag
Specifies whether arithmetic or logic operation.

ci0, 1: Carry input

cp0, 1: MAX, MIN AB function execution flag
Executes large or small (MIN AB, MAX AB) select function.
In this case, the values set for fs0, 1 and fm0, 1 are ignored.

mg0, 1: MAX, MIN select flag
Specifies select large or select small in large/small (MIN AB, MAX AB) select function.

The truth table for ALU#0 and ALU#1 is shown below.

Input						Output		
		fs0, 1				fm0, 1 = 1	fm0, 1 = 0 (ARITHMETIC operation)	
mg0, mg1	cp0, cp1	bit 3	bit 2	bit 1	bit 0	(LOGIC function)	ci0, 1 = 1 (no carry)	ci0, 1 = 1 (with carry)
X	0	0	0	0	0	Q = U*	Q = U	Q = U plus 1
X	0	0	0	0	1	Q = (U+V)*	Q = U+V	Q = (U+V) plus 1
X	0	0	0	1	0	Q = U*•V	Q = U+V*	Q = (U+V*) plus 1
X	0	0	0	1	1	Q = 0	Q = minus 1 (2'compl)	Q = ZERO
X	0	0	1	0	0	Q = (U•V*)	Q = U plus (U•V*)	Q = U plus (U•V*) plus 1
X	0	0	1	0	1	Q = V*	Q-(U+V) plus (U•V*)	Q = (U+V) plus (U•V*) plus 1
X	0	0	1	1	0	Q = UxorV	Q = U minus V minus 1	Q = U minus V
X	0	0	1	1	1	Q = U•V*	Q = (U•V*) minus 1	Q = U•V*
X	0	1	0	0	0	Q = U*+V	Q = U plus (U•V)	Q = U plus (U•V) plus 1
X	0	1	0	0	1	Q = (UxorV)*	Q = U plus V	Q = U plus V plus 1
X	0	1	0	1	0	Q = V	Q = (U+V*) plus (U•V)	Q = (U•V*) plus (U•V) plus 1
X	0	1	0	1	1	Q = U•V	Q = (U•V) minus 1	Q = U•V
X	0	1	1	0	0	Q = 1	Q = U plus U	Q = U plus U plus 1
X	0	1	1	0	1	Q = U = V*	Q = (U+V) plus U	Q = (U+V) plus U plus 1
X	0	1	1	1	0	Q = U+V	Q = (U+V*) plus U	Q = (U+V*) plus U plus 1
X	0	1	1	1	1	Q = U	Q = U minus 1	Q = U
0	1	X	X	X	X	Q = MIN UV		
1	1	X	X	X	X	Q = MAX UV		

- • : AND
- + : OR
- xor : EXCLUSIVE OR
- * : Denotes negative logic.
- MIN : Compares U and V to choose the smaller.
- MAX : Compares U and V to choose the larger.
- X : This bit is ignored.
- U : Denotes Mo for ALU#0; denotes Oo for ALU#1.
- V : Denotes No for ALU#0; denotes Po for ALU#1.

5.7 ALU Extended Function Register

This register limits the calculation results and perform 16-bit operations. However, if the ALU units are combined to function as a 16-bit ALU (when ALUEXT's hd flag = 1), ALU#0 function register ALUMOD0 must be set to the same value as ALU#1 function register ALUMOD1, and the ALUEXT flags must be set so that sat1 = sat0, and fix1 = fix0.

ALUEXT ALU extended function register

If the result of an ALU addition or subtraction exceeds 255 (FFh) or is smaller than 0, this register sets the calculation result to 255 (FFh) or 0, respectively. It also allows the two ALU units to be combined to perform 16-bit operations.

bit 7							bit 0	Reg. name	Address
0	0	0	hd	sat1	fix1	sat0	fix0	ALUEXT	A4h

fix0: ALU#0 arithmetic operation result limit enable flag

- fix0 = 0: The result of ALU#0's arithmetic operation is output directly to SHIFT#0 without modification.
- fix0 = 1: If the result of ALU#0's arithmetic operation exceeds 255 (FFh) (carry output = 1), the value is fixed at 255. If the result is smaller than 0 (carry output = 0), the value is fixed at 0.

sat0: ALU#0 arithmetic operation result limit value setting flag

- sat0 = 0: When fix0 = 1, if ALU#0 performs a subtraction and the result is smaller than 0 (carry output = 0), the result is fixed at 0.
- sat0 = 1: When fix0 = 1, if ALU#0 performs an addition and the result is greater than 255 (FFh) (carry output = 1), the result is fixed at 255.

fix1: ALU#1 arithmetic operation result limit enable flag

- fix1 = 0: The result of ALU#1's arithmetic operation is output directly to SHIFT#1 without modification.
- fix1 = 1: If the result of ALU#1's arithmetic operation exceeds 255 (FFh) (carry output = 1), the value is fixed at 255. If the result is smaller than 0 (carry output = 0), the value is fixed at 0.

sat1: ALU#1 arithmetic operation result limit value setting flag

- sat1 = 0: When fix1 = 1, if ALU#1 performs a subtraction and the result is smaller than 0 (carry output = 0), the result is fixed at 0.
- sat1 = 1: When fix1 = 1, if ALU#1 performs an addition and the result is greater than 255 (FFh) (carry output = 1), the result is fixed at 255.
- Note: If hd = 1, the above value 255 (FFh) is changed to 65535 (FFFFh).

hd: ALU 16-bit-processing enable flag

This flag connects ALU#0 and ALU#1 together to perform 16-bit operations. The two bit shifts SHIFT#0 and SHIFT#1 (9 bits each) are also connected together to become a 17-bit shifter to handle 16-bit operation results and the carry bit.

hd = 0: 8-bit processing mode (normal mode)

hd = 1: 16-bit processing mode

Operations are performed for Mo (upper byte) and Oo (lower byte), and for No (upper byte) and Po (lower byte).

The operation result is output through the bit shifter to Mi for the upper byte, and to Ni for the lower byte. When operating in this mode, make sure ALUMOD0 = ALUMOD1, ALUEXT flags sat1 = sat0 and fix1 = fix0, and SHIFT0 = SHIFT1.

Note: If cp0 in ALUMOD0 is 1, the operation result is not routed through the right bit shifter (SHIFT#0), and the data chosen by large/small comparison (i.e., when PTSEL-M bit 7: Mm = 0) is output to Mi (upper byte).

If cp1 in ALUMOD1 is 1, the operation result is not routed through the right bit shifter (SHIFT#1), and the data chosen by large/small comparison (i.e., when PTSEL-N bit 7: Nm = 0) is output to Ni (lower byte).

5.8 Right Bit Shifter Shift-Setting Registers

These registers set the amount of shift for the two right bit shifters (9 bits each).

SHIFT0 SHIFT#0 shift-setting register

This register specifies how much the values in ALU#0 (including the carry output in MSB) are shifted to the right.

The lower byte of the operation result is output to the Mi input port (when PTSEL-M bit 7: Mm = 0).

bit 7							bit 0	Reg. name	Address
0	0	0						SHIFT0	A5h

SHIFT1 SHIFT#1 shift-setting register

This register specifies how much the values in ALU#1 (including the carry output in MSB) are shifted to the right.

The lower byte of the operation result is output to the Ni input port (when PTSEL-N bit 7: Nm = 0).

bit 7			bit 4				bit 0	Reg. name	Address
0	0	0						SHIFT1	A6h

0: Do not use this bit. Leave it set to 0.

Note: If hd in the ALUEXT register is set to 1 (16-bit operation), the two bit shifts (9 bits each) can be connected together to function as a 17-bit shifter (including the carry output in ALU's 16-bit operations). In this case, make sure the registers are set so SHIFT0 = SHIFT1.

Also, when processing is completed (if PTEL-M, N bit 7: Mm = Nm = 0), the upper byte of the result is output to Mi and the lower byte is output to Ni.

Of the bits 0–16, the MSB in bit 16 is ignored.

5.9 Mode Register Set

REGVDIS Register Data Disable

bit 7							bit 0	Reg. name	Address
0	0	0	0	0	0	0	rgd	REGVDIS	FDh

rgd: Register data disable flag

This flag disables the data written to the register as it is written or when VS* is asserted.

rgd = 0: Enable the data written to the register.

Enables the data written to the register as it is written or when VS* is asserted.

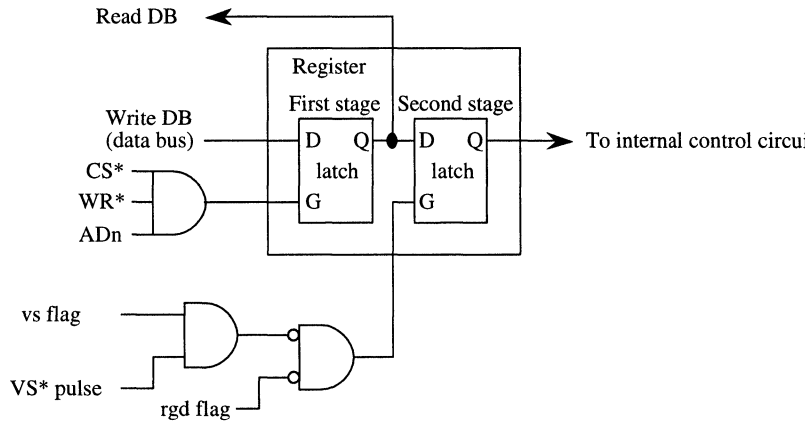
Use the SMOD register vs flag to specify whether data is enabled when written to the register, or when VS* is asserted.

rgd = 1: Disables the data written to the register

The data written to the register is disabled until this flag is cleared to 0.

0: Do not use this bit. Leave it set to 0.

Register configuration



The VS* pulse is a negative signal pulse. When a low VS* pulse occurs following a low VSEN* signal, one pulse is output at the next rising edge of ICLK.

Note: The SMOD register vs flag and this rgd flag are enabled once the register is written to.

Note: For a read from the register, the data in the first-stage latch is read.

SMOD Sub-mode register

bit 7							bit 0	Reg. name	Address
0	0	0	0	0	0	wrf	vs	SMOD	FEh

vs: Register write enable VS* sync flag

This vs flag is set only when the rgd flag = 0. This flag specifies the timing priority and determines whether the value written to the register is enabled when written to the register, or when the next VS* is asserted.

vs = 0: Field asynchronous mode.

Values entered are enabled when written to the register.

vs = 1: Field synchronous mode.

Values entered are enabled when VS* is asserted immediately after the field in which the value is located is written to the register (i.e., when the next field begins).

wrf: Register data enable/disable monitor flag (read-only)

The IP90C55's internal registers are constructed with dual latches. When any register is written to (i.e., CS* and WR* are pulsed low), the data on DB is latched in the first-stage latch. The wrf flag is then set to 1, and is cleared when the gate to the second-stage latch is opened. The gate to the second-stage latch opens if VS* is pulsed low when rgd = 0 and vs = 1, or if rgd = 0 and vs = 0.

Clearing has priority so that the wrf flag can be cleared even during write operations.

wrf = 0; The written data is valid.

wrf = 1: Writing in progress.

MMOD Main mode register

bit 7							bit 0	Reg. name	Address
0	0	0	0	0	0	fg	r	MMOD	FFh

r: Software reset bit

- r = 0: Cancels a software reset. Resets remain canceled.
- r = 1: When 1 is written to this bit, all registers are reset to 0 within three clock cycles beginning with the next clock cycle. At the same time, this bit is also reset to 0.

fg: Field identifying signal latch timing

- fg = 0: The field identifying signal (FLDI) is always referenced in the valid field.
- fg = 1: The field identifying signal (FLDI) is latched at the beginning of each field.

  (The FLDI level detected at the rising edge of ICLK immediately after VS* is driven low and latched.)

0: Do not use this bit. Leave it set to 0.

5.10 Device Test Register

TESTM

This register is used to test the internal logic of the device before it is shipped from the factory. Do not use this register.

The chip is placed in test mode when this register is set to 1. When in test mode, the IP90C55 may not operate normally.

Section 6: Electrical Characteristics

6.1 Absolute Maximum Ratings

Parameter	Symbol	Rating	Unit
Supply voltage	Vdd	-0.3 to 6.5	V
Input voltage	Vi	-0.3 to Vdd + 0.3	V
Input current	Ii	±10	mA
Output current	Io	10	mA
Operating temperature	Topt	0 to 70	°C
Storage temperature	Tstg	-10 to 80	°C

6.2 Recommended Operating Conditions

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Supply voltage	Vdd		4.75	5.0	5.25	V
HIGH level input voltage	Vih	TTL level	2.2	—	Vdd	V
LOW level input voltage	Vil	Normal input	0	—	0.8	V
Input rise time	Tri	TTL level	0	—	100	ns
Input fall time	Tfi	Normal input	0	—	100	ns
Input rise time	Tri	Schmitt trigger input	0	—	1000	ns
Input fall time	Tfi	Note 1	0	—	1000	ns

6.3 Input/Output Pin Capacitance

Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Input pin	Cin	f = 1 MHz	—	10	25	pF
Output pin	Cout	f = 1 MHz	—	10	25	pF

6.4 DC Characteristics

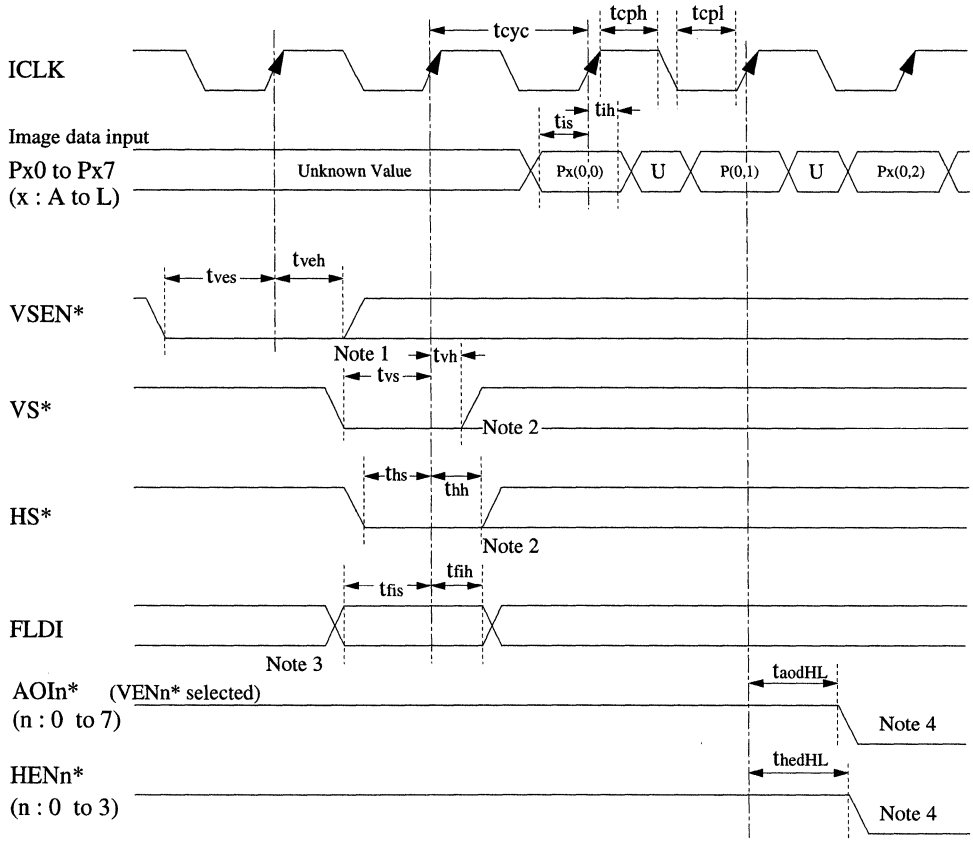
Parameter	Symbol	Conditions	Min.	Typ.	Max.	Unit
Static current consumption (Note 1)	Il	Vi = Vdd or GND			200	µA
Off-state output leakage current (ports A–L)	Ioz	Vi = Vdd or GND	-10		-200	µA
Output shorting current (Note 2) (all output and input/output pins)	Ios	Vo = GND	-250			mA
LOW level input leakage current						
Normal input pins (Except RST* and ports A to L)	Iil	Vi = GND	-10		10	µA
Input/output and inputs with pull-up resistors (RST* and ports A to L)	Iipl	Vi = GND	-10		-200	µA
HIGH level input leakage current (all input, all input/output pins)	Iih	Vi = Vdd	-10		10	µA
LOW level output voltage	Vol1	Iol = 8 mA			0.4	V
HIGH level output voltage (DB0–7)	Voh1	Ioh = -8 mA	2.4			V
LOW level output voltage	Vol2	Iol = 8 mA			0.4	V
HIGH level output voltage (output and input/output pins except DB0–7)	Voh2	Ioh = -8 mA	2.4			V
Schmitt hysteresis voltage	Vsch			0.5		V

Note 1: This value does not include static current consumption drawn in pull-up and pull-down resistors.

Note 2: Output shorting current is for shorting of less than 1 second, in only one pin on the device.

6.5 AC Characteristics

a) Data timing at the beginning of a frame



Note 1: VS* is enabled when a low VSEN* is detected at the rise of ICLK immediately preceding the cycle that follows the low VS*. VSEN* allows the IP90C55 to be easily synchronized with other image processing LSIs or programmable controllers. If the IP90C55 need not be synchronized with other LSIs (i.e., in normal operation), VSEN* must be held low.

Note 2: VS* and HS* need not always be pulses; they can be level-sensitive inputs, as shown above by the broken lines. In this case, however, they must be held high for at least two clock cycles before they can be driven low again.

Note 3: The FLDI signal must be held low or high from the time VS* is asserted until the AOI is finished. However, this restriction applies when the MMOD register's fg = 0. If fg = 1, the input level on the FLDI pin is latched at the rising edge of ICLK immediately after VS* is driven low, and is retained until VS* is driven low next time.

Note 4: This applies when bit-n in registers HZSTART, VOZSTART, and VEZSTART are set to 1 when specifying an AOI.

Units: ns

Parameter	Symbol	Pin load capacitance					
		30 pF			60 pF		
		Min.	Typ.	Max.	Min.	Typ.	Max.
ICLK period	tcyc	25.0	—	—	25.0	—	—
HIGH level period of ICLK	tcph	10.0	—	—	10.0	—	—
Low level period of ICLK	tcpl	10.0	—	—	10.0	—	—
Ports A to L data setup time	tis	5.0	—	—	5.0	—	—
Ports A to L data hold time	tih	2.0	—	—	2.0	—	—
VSEN* setup time	tves	9.0	—	—	9.0	—	—
VSEN* hold time	tveh	2.0	—	—	2.0	—	—
VS* setup time	tvs	9.0	—	—	9.0	—	—
VS* hold time	tvh	2.0	—	—	2.0	—	—
HS* setup time	ths	9.0	—	—	9.0	—	—
HS* hold time	thh	2.0	—	—	2.0	—	—
FLDI setup time	tfis	10.5	—	—	10.5	—	—
FLDI hold time	tfih	2.0	—	—	2.0	—	—
AOIn* (n = 0 to 7) delay time	taodHL	3.0	—	16.0	4.0	—	20.0
HENn* (n = 0 to 3) delay time	thedHL	3.0	—	16.0	4.0	—	20.0

b) Data input/output timing on each line

Note 1: PA(m, n) = PB(m,n).

Note 2: PD(m, n) = F [PA(m, n) PC(m, n)],
where F[] represents any ALU calculation function.

Note 3: To make the horizontal effective area setting, set HZSTART register bit-n = 1
For output level setting, HENDn register = 0003h.

Note 4: The dotted line represents first-line or last-line timing.

Note 5: The above timing chart is for the following data path setting:
PA –> Ai (internal port) –> Bi (internal port) –> PB

Note6: PA –> Ai (internal port) –> Bi (internal port) –> [ALU#0] –> Mi (internal port) –> Do (internal port) –> PD
PA –> Ai (internal port) –> Bi (internal port) –>

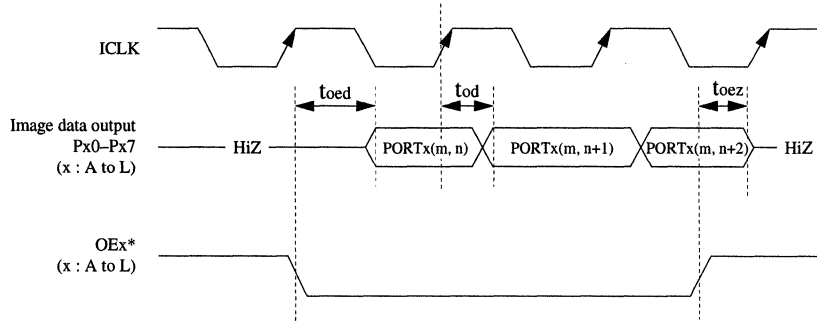
Note 7: The use of the ALU function creates a delay of three clock cycles
relative to normal input/output timings.

Units: ns

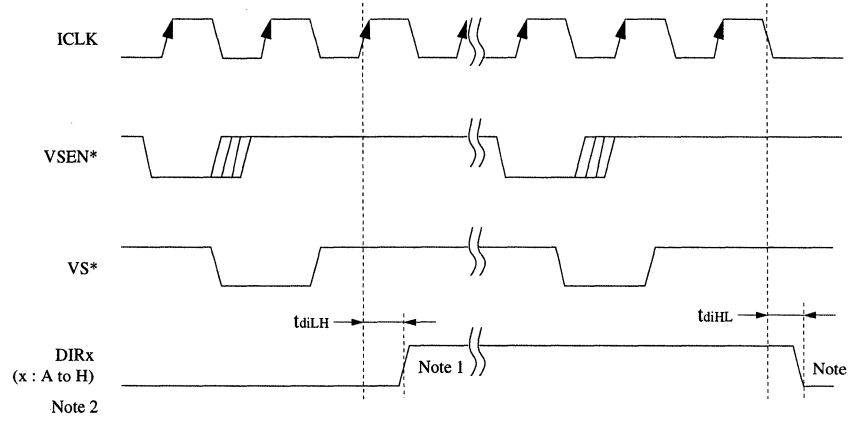
Parameter	Symbol	Pin load capacitance					
		30 pF			60 pF		
		Min.	Typ.	Max.	Min.	Typ.	Max.
Ports A to L data setup time	tis	5.0			5.0		
Ports A to L data hold time	tih	2.0			2.0		
Ports A to L delay time	tod	3.0		17.0	4.0		20.0
AOIn* (n = 0 to 7) delay time	taodHL toadLH	3.0 3.0		16.0 16.0	4.0 4.0		20.0 20.0
HENn* (n = 0 to 3) delay time	thedHL thedLH	3.0 3.0		16.0 16.0	4.0 4.0		20.0 20.0

c) Image data output control-related timing

Image data output enable



Port input/output flag output



Note 1: This applies when the SMOD register vs flag (bit 0) is set to 1. (The signal becomes valid when VS* is detected low by the rising edge of ICLK immediately after the field in which the value is located is written to the register.)

Note 2: Ports PI to PL do not have DIR pins.

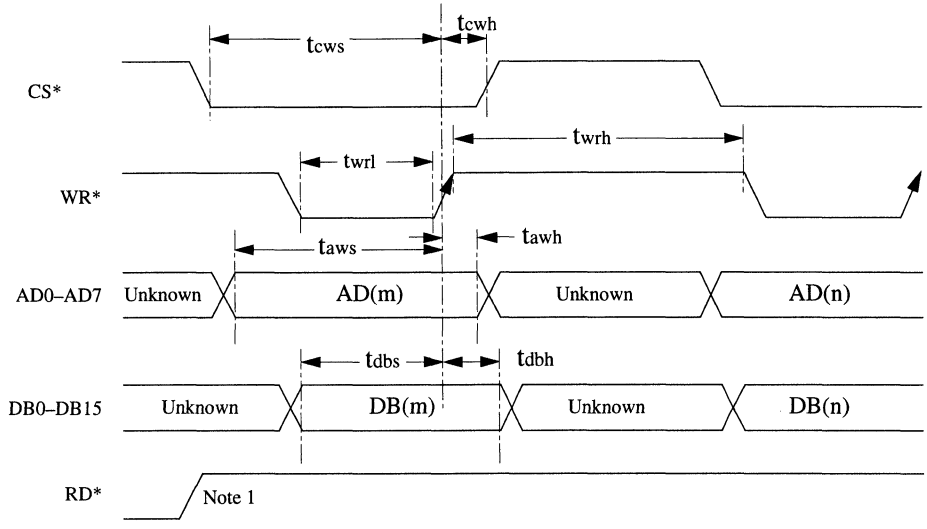
Units: ns

Parameter	Symbol	Pin load capacitance					
		30 pF			60 pF		
		Min.	Typ.	Max.	Min.	Typ.	Max.
Delay time from fall of OEx* (x = A to L) to port x enabled	toed	1.0		13.0	2.0		15.0
Delay time from rise of OEx* (x = A to L) to port x placed in Hi-Z state	toez	1.0		14.0	2.0		15.0
DIRx (x = A to H) delay time	tdiLH tdiHL	3.0 3.0		1 cycle Note	3.0 3.0		1 cycle Note

Note: Synchronized with ICLK to avoid signal collisions.

d) CPU interface-related timing

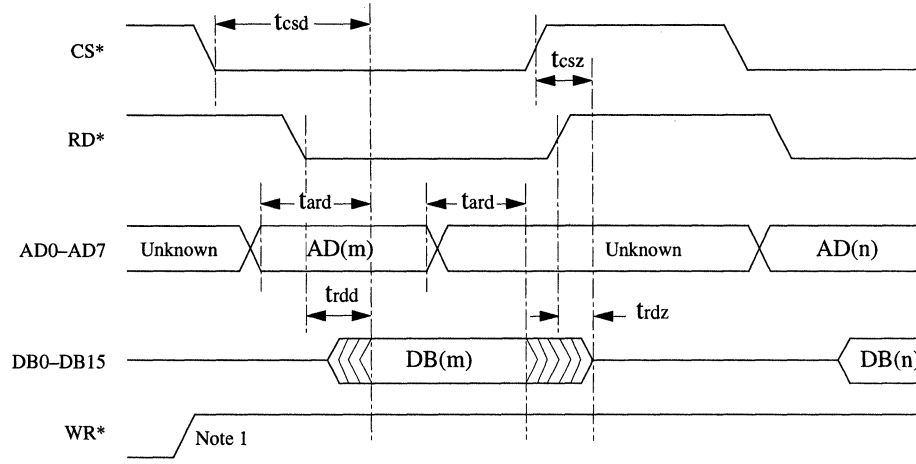
Write Cycle



Units: ns

Parameter	Symbol	Pin load capacitance					
		30 pF			60 pF		
		Min.	Typ.	Max.	Min.	Typ.	Max.
CS* setup time	tcws	40.0			40.0		
CS* hold time	tcwh	4.0			4.0		
Low pulse duration of WR*	twrl	40.0			40.0		
High pulse duration of WR*	twrh	60.0			60.0		
AD0–AD7 set up time	taws	40.0			40.0		
AD0–AD7 hold time	tawh	4.0			4.0		
DB0–DB7 setup time	tdbs	40.0			40.0		
DB0–DB7 hold time	tdbh	4.0			4.0		

Read Cycle

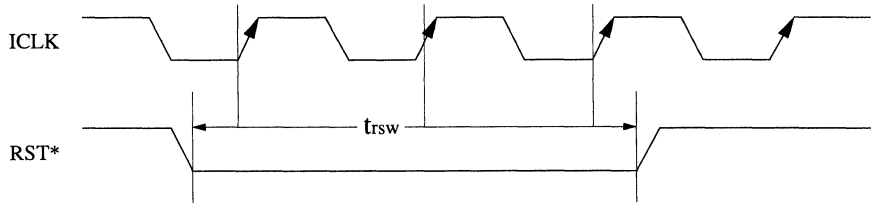


Note 1: When the WR* pin is low, hold the RD* pin high to ensure that both pins are never driven low simultaneously. Similarly, when the RD* pin is low, make sure the WR* pin is high.

Units: ns

Parameter	Symbol	Pin load capacitance					
		30 pF			60 pF		
		Min.	Typ.	Max.	Min.	Typ.	Max.
Delay time from valid AD0–AD7 until DB0–DB7 enabled	tard	2.0		35.0	2.0		35.0
Delay time until DB0–DB7 enable (relative to fall of CS*)	tcsd	2.0		35.0	2.0		35.0
Delay time until DB0–DB7 enable (relative to fall of RD*)	trdd	2.0		35.0	2.0		35.0
Delay time until DB0–DB7 placed in Hi-Z state (relative to rise of CS*)	tcsz	2.0		20.00	2.0		2.0
Delay time until DB0–DB7 placed in Hi-Z state (relative to rise of RD*)	trdz	2.0		20.00	2.0		2.0

e) Reset-related timing



Note: The reset signal must be held low at least three clock cycles.

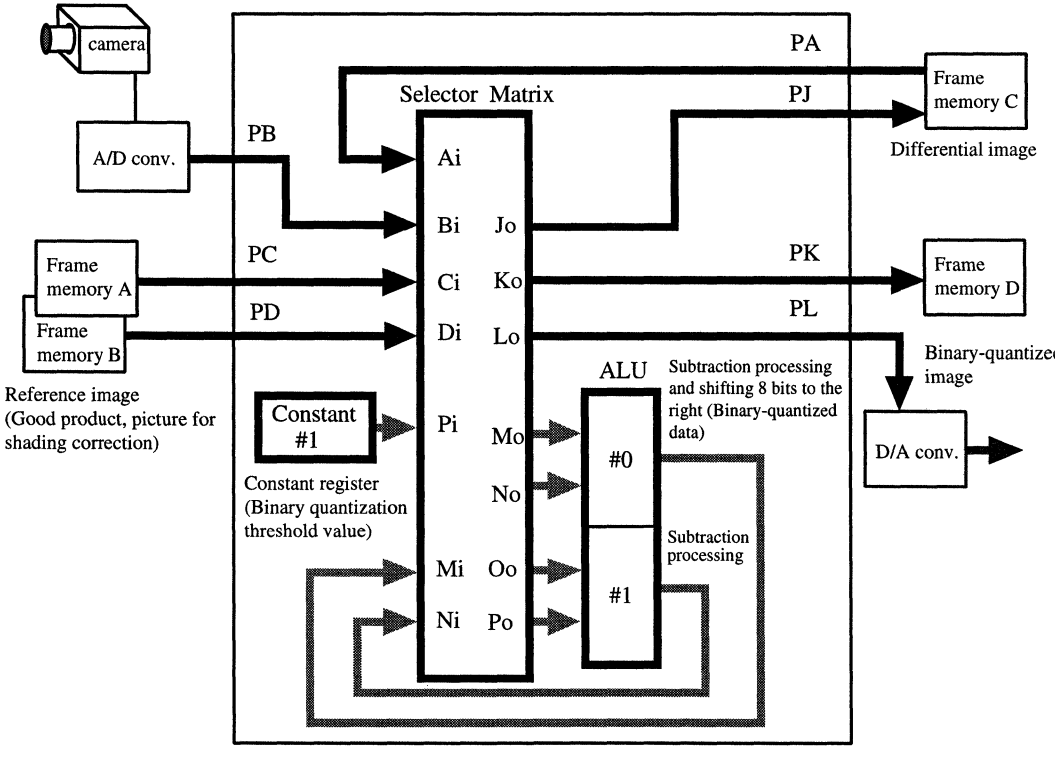
Units: clock cycles

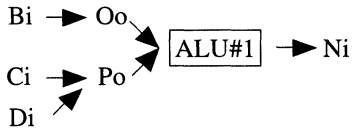
Parameter	Symbol	Pin load capacitance					
		30 pF			60 pF		
		Min.	Typ.	Max.	Min.	Typ.	Max.
Low pulse duration of RST*	trsw	3.0			3.0		

Section 7: Sample Applications and Connection Configurations

7.1 Example of Frame-to-Frame Calculation Processing

(differential shading correction, binary conversion)

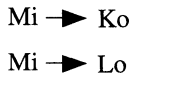




1. ALU#1 performs image differentiation between the camera image and reference image A (good image) or B (image for shading correction).

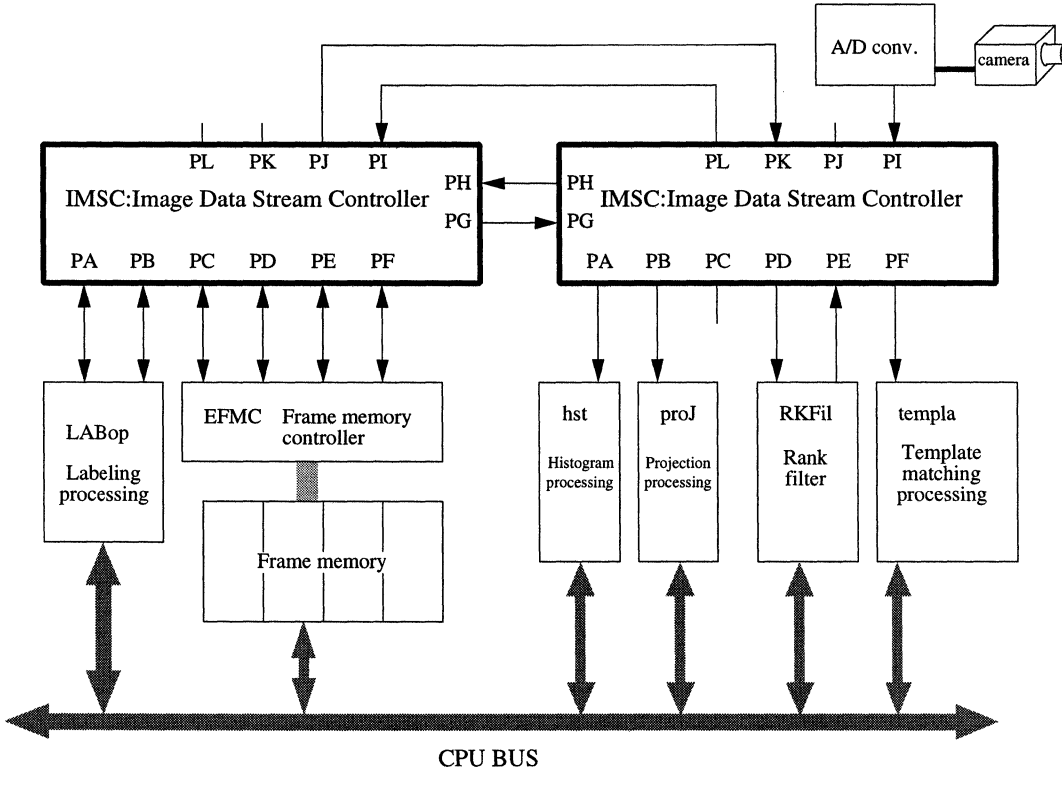
2. Results are transmitted from Ni to Jo and stored as a differential image.

3. ALU#0 then subtracts the binary threshold value from constant register CONST#1 (reference value for binary conversion) from this differential image, and then converts the results—including the carry bit—to binary format using an 8-bit right shift.



4. The binary data is sent from Ko and stored in frame memory D as a binary image. At the same time, the image is output from Lo in analog form.

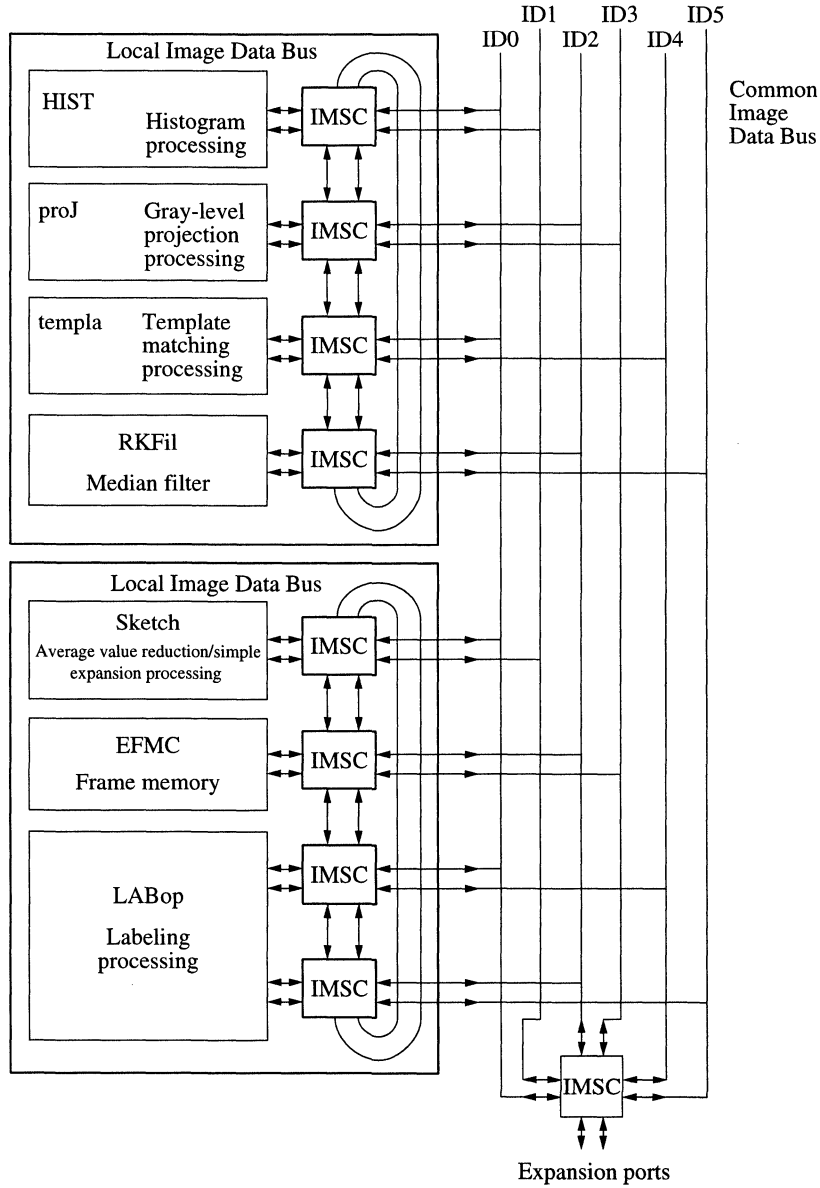
7.2 Switching Image Data Streams Between Image Processing Functional Blocks



7.3 Typical Configuration of Image Network

The following shows an example of a flexible network configuration that connects image-processing blocks using multiple IP90C55s.

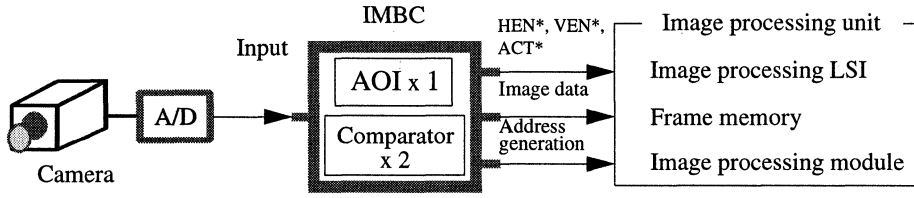
Each image-processing module is constructed with IMSC chips, which are mounted in ring form on each board and in bus form between boards.



Appendix

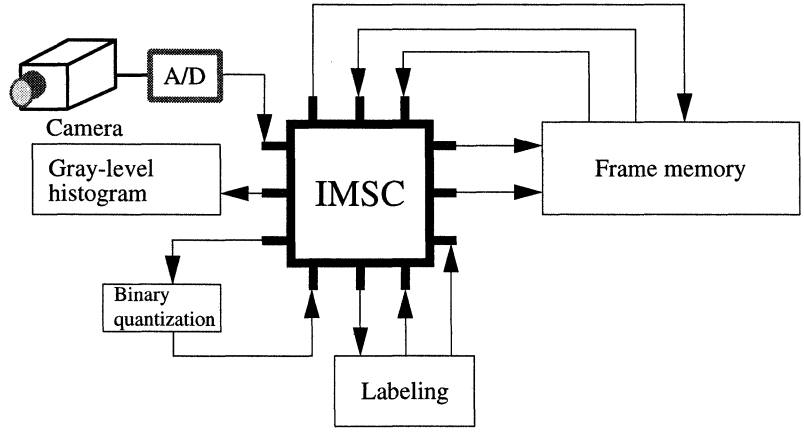
Differences from the IP90C51 (IMBC)

- The IP90C51 (IMBC) controls image data for a single image-processing module or chip.



The IP90C51 (IMBC) controls image data (by clipping, binary-quantizing, and generating addresses) for one image processing unit, while at the same time generating frame memory addresses.

- The IP90C55 (IMSC) dynamically controls (i.e., modifies) the paths of image data flows for multiple image-processing modules or chips.



The IP90C55 (IMSC) performs frame-to-frame arithmetic/logic operations between image processing units, between image processing functional modules, or between frame memories using its internal ALU units.

SIDIP Image Processing Modules

IP90MD08

Template Matching Module

Product Manual

Ver. E 1.1

Sumitomo Metal Industries, Ltd.

SIDIP Two-Dimensional Grayscale Template Matching Module IP90MD08

FEATURES AND FUNCTIONS

- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Industries
- Based on the IP90C08 LSI chip; provides two-dimensional grayscale (8-bit) template matching, including difference, absolute value, and summation processing using local image areas and templates
- Outputs matched filter results (sum-of-differences values)
- Stores up to four sets of coordinates in order of best match
- Maximum template size of 16 x 7 pixels (also handles one-dimensional templates up to 112 x 1 pixels)
- Two sets of template registers with external switching capability for continuous processing
- Individual enable/disable settings for each pixel in the template
- Input: 8-bit grayscale image
- Output: 16/8-bit matched filter output
- Maximum operating frequency of 25 MHz (image input rate of 4 ns per pixel)
- Maximum processing area of 4095 pixels by 4095 lines
- Input image can be converted to binary format
- Any rectangular area from the input image can be selected for template matching
- 5120-pixel x 8-bit x 6-line memory
- Module ID can be read out
- Operates on a single 5V power source
- All terminal signal levels are TTL level
- Module size of 48.8 x 127.0 mm (SIDIP single-width size)
- Control software reference source code list (C language) included

APPLICATIONS

- Option boards for general-purpose image analysis devices
- Built-in image processing devices for factory automation (FA) and office automation (OA) equipment
- Built-in security devices
- Evaluating IP90C08 (templa) and IP90C51 (IMBC) chips

PRODUCT DESCRIPTION

The core of the IP90MD08 module is the Sumitomo Metal Industries IP90C08 two-dimensional grayscale template matching LSI chip. The module also contains the peripheral circuitry necessary for interfacing with the templa chip. The IP90MD08's bus configuration is compatible with the SIDIP standard proposed by Sumitomo Metal Industries, and also with other boards and bus interfaces in the Sumitomo image processing module series.

The IP90MD08 module performs two-dimensional grayscale template matching in exactly the same way as the IP90C08 chip. (For further information on this processing, see "Algorithms," below.)

The IP90MD08 performs template matching processing on the pixels of binary or grayscale (8-bit) raster images. These images are input along with a pixel clock synchronized with the image pixels, and the horizontal and vertical synchronization signals of the image. The module outputs the results as matched filter (sum-of-differences) data over a 16-bit bus line, with the same pixel clock and horizontal and vertical synchronization signals as the input signal. (For further description of input and output images, see "Structure and Operation," below.) In addition, the IP90MD08 can store the four best-matching sets of coordinates, in order of closeness of fit.

The IP90MD08 stores template data in two sets of template registers. While template matching is being applied from one register, the other register can receive the next set of template data, thus enabling processing to continue without interruption. The IP90MD08 can handle template data up to 16 x 7 pixels in size, and can mask any pixels in the template to exclude them from processing.

The largest allowable input image is 4095 pixels by 4095 lines. The maximum pixel clock frequency is 25 MHz.

The IP90MD08 module can accommodate binary template matching by using a built-in comparator to convert incoming grayscale images to binary format. The host computer controls the threshold value for binary conversion.

Matched filter output is in 16-bit format. However, the output can also be separated into its lower 8 bits and upper 8 bits. When only the lower 8 bits are output, the module can clip output values that exceed 8 bits (overflow values) to a value of 255 (FFh).

The IP90MD08 has an individual module ID number, which the host computer can use to verify which modules are in which slots in the base board. This provides an efficient way to configure the system automatically.

The IP90MD08 module's board is the SIDIP single-width size (48.8 x 127.0 mm).

The IP90MD08 provides an easy way to perform two-dimensional grayscale template matching processing. It replaces previous low-speed software systems and bulky hardware with a space-saving, high-speed hardware module. A real-time grayscale template matching system can be configured easily and simply by installing the IP90MD08 in a SIDIP-compatible base board. Such a system can be used in a wide variety of applications, including those requiring position detection, image alignment, or vector pursuit functions, as well as for image searching in databases and electronic filing systems.

Other uses include general purpose image processing equipment, particularly built-in processing equipment in factory automation (FA) applications requiring high-speed processing, as well as office automation (OA) equipment, and security equipment.

ALGORITHMS

This section describes the template matching algorithms used by the IP90MD08 module's IP90C08 chip.

Template matching is the process of searching among raster scan input image areas for the areas that most closely match the pixel values stored in the IP90C08's template registers. The IP90C08 does this by determining pixel-to-pixel matching for binary images, and expands this method to provide template matching of grayscale image data.

The process involves calculating the sum of the absolute values of pixel-by-pixel differences between the incoming grayscale image and the template stored in the chip. This sum of differences indicates how closely the input image matches the template.

Figure 1 illustrates the processing of a local area of M x N pixels, having its lower right corner at (a, b), and a template of the same size. A pixel at (m, n) of the extracted local area is compared to (that is, subtracted from) the value of the pixel at (m, n) in the template. The absolute value of the difference indicates the closeness of the match.

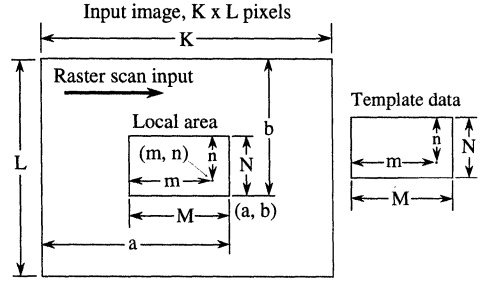


Figure 1: Algorithm

Therefore, the degree of mismatch between the local image area with its lower right corner at (a, b) and the template can be expressed by this equation:

$$S(a,b) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} \left| I(a+m-M, b+n-N) - T(m,n) \right|$$

where:

- (a, b) is the lower right corner of the local area within the input image.
- S(a, b) is the sum of differences with relation to the local area of M x N pixels having its lower right corner at (a, b).
- I(a + m – M, b + n – N) is the pixel grayscale value of (m, n) within the local area of the input data.
- T(m, n) is the template data for the corresponding point (m, n) within the template.

The IP90C08 calculates the sum of differences with respect to the input image, and outputs the result. The lower the sum of differences is, the less difference there is between the template and the image area of the input data, and so the closer the fit. The output value can be further augmented by supplementary functions, including addition of bias values, or division by means of a shift operation. This enables the IP90C08 to function as a matched filter or average smoothing filter, and to create an output image composed of closeness-of-fit values.

In addition, the IP90C08 can determine which sets of coordinates have the lowest sum of differences (corresponding to the positions with the best fit), and save the four lowest in its matching coordinates register. The chip can also rank these coordinates in order of closeness-of-fit.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD08 module can handle binary-coded or 8-bit grayscale raster images (non-interleaved). The IP90MD08 requires as a minimum a system that includes a pixel clock synchronized with the pixels of the raster image, as well as the horizontal and vertical synchronization signals necessary to construct the raster image. These signals are synchronized with the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

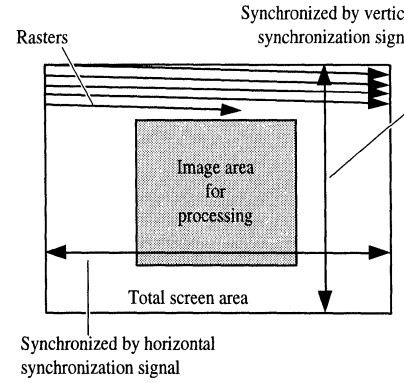


Figure 2: Definition of Image Area

The IP90MD08 can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded rectangle in Figure 2. The host computer controls the size and position of this area. In addition, areas outside this designated area can be considered background and processed as gradient value 0 (binary value [0]).

The output image is a matched filter (sum-of-differences) image, and, like the input image, is a raster image. The output also has the same pixel clock and horizontal and vertical synchronization signals as the input image.

Figure 3 shows an example of an actual application.

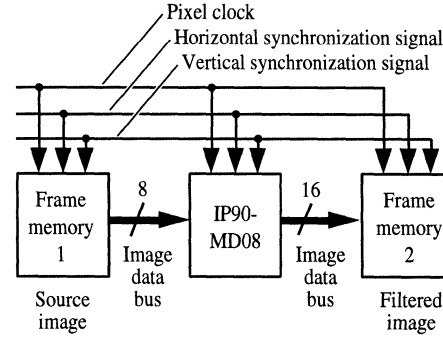


Figure 3: Sample Application

(2) Image Input Block (IP90C51)

Figure 4 shows a block diagram of the IP90MD08 module.

Inverted image data enters through the input image bus, located at IN00*–IN07* in Figure 4. This input data is inverted again in the image input block. The input block also designates the area of the input image to be processed as described earlier, and, if binary processing is to be used, converts 8-bit grayscale data to binary format.

(3) 1H Line Delay

To perform two-dimensional grayscale template matching processing, the IP90MD08 requires seven successive horizontal lines of horizontal image data. The incoming image data enters one line at a time, so the module obtains seven lines by applying six successive line delays of one horizontal (1H) line each.

(4) Core Block (IP90C08)

This is the block in which the IP90C08 performs the necessary functions for two-dimensional grayscale template matching.

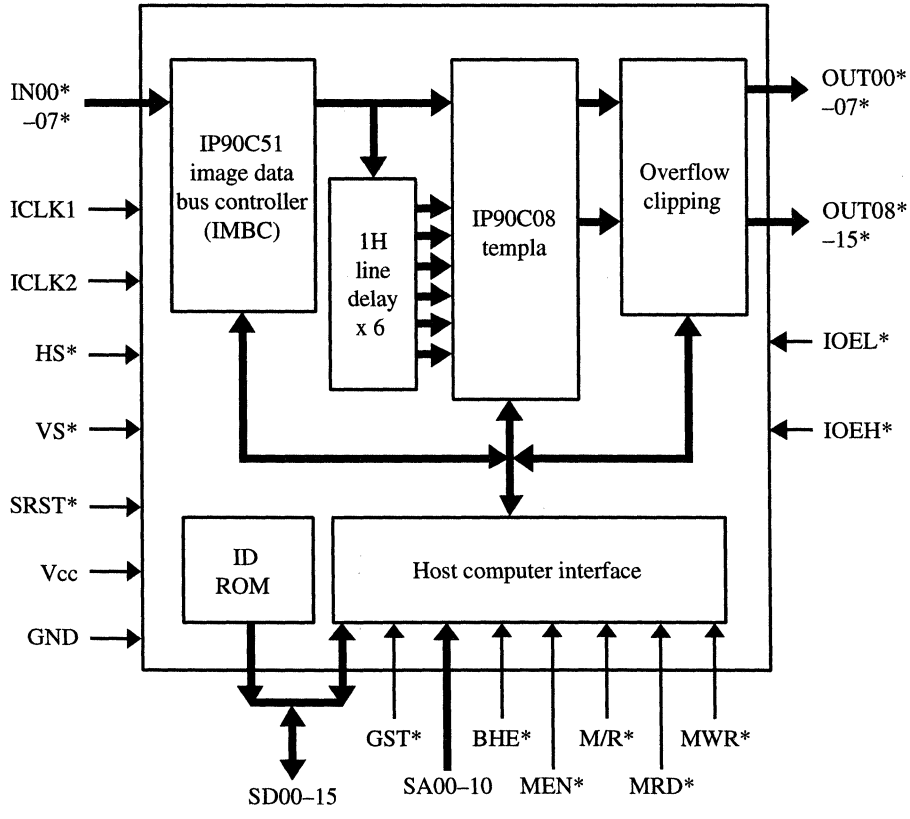


Figure 4: Block Diagram

(5) Overflow Clipping Block

The results of sum-of-differences processing are output to the image bus with a maximum width of 16 bits (the IP90C08 can also be set for 8-bit operation). The lower 8-bits and upper 8-bits of the output have three possible states, enabling independent control of the connection. The output is in inverted logic format.

If 8-bit output is used, output values in excess of 8 bits are considered overflow and are clipped to the maximum value of 255 (FFh). This function can be turned on or off by software.

(6) Timing Control Block

The timing control block is the point of entry for the pixel clocks (ICLK1, ICLK2), horizontal synchronization signal (HS*), and vertical synchronization signal (VS*).

When the IP90MD08 uses a pixel clock frequency higher than 20 MHz, a two-phase clock input is created using both ICLK1 and ICLK2. At frequencies of 20 MHz or lower, only ICLK1 is used.

(7) Host Computer Interface Block

The host interface block contains the interface through which the host computer controls the IP90MD08. The data bus (SD00–15) is 16 bits wide, and the address bus (SA00–10) is 11 bits wide. Write and read operations are controlled by the MWR* and MRD* signals, respectively. Byte access or cord access are enabled by a combination of the address A0 setting and the BHE* signal. Although memory space and register space are defined in all SIDIP modules, the IP90MD08 allocates all functions to register space. The IP90MD08 is also designed for either 8-bit or 16-bit data bus width.

1) ID Number Reading

When GST* is taken to active level, the IP90MD08's module ID number (28h) is outputted using the lower 8 bits of the data bus.

2) System Reset

When the SRST* signal is set to active level, a hardware reset is initiated. This resets all registers in the module to 00h.

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD08. Signal pins are divided between the image system (Table 1A) and control system (Table 1B), having three and two external connectors, respectively. All signal pin assignments conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
OUT00*	CN1C	12	Output image data bus (Bit 0)	O
OUT01*	CN1C	27	Output image data bus (Bit 1)	O
OUT02*	CN1C	13	Output image data bus (Bit 2)	O
OUT03*	CN1C	28	Output image data bus (Bit 3)	O
OUT04*	CN1C	14	Output image data bus (Bit 4)	O
OUT05*	CN1C	29	Output image data bus (Bit 5)	O
OUT06*	CN1C	15	Output image data bus (Bit 6)	O
OUT07*	CN1C	30	Output image data bus (Bit 7)	O
OUT08*	CN1D	2	Output image data bus (Bit 8)	O
OUT09*	CN1D	17	Output image data bus (Bit 9)	O
OUT10*	CN1D	3	Output image data bus (Bit 10)	O
OUT11*	CN1D	18	Output image data bus (Bit 11)	O
OUT12*	CN1D	4	Output image data bus (Bit 12)	O
OUT13*	CN1D	19	Output image data bus (Bit 13)	O
OUT14*	CN1D	5	Output image data bus (Bit 14)	O
OUT15*	CN1D	20	Output image data bus (Bit 15)	O
IOEL*	CN1E	2	Image bus output enable (lower)	I
IOEH*	CN1E	17	Image bus output enable (higher)	I

Note: An asterisk (*) following a signal name indicates inverse logic.

Table 1A: Image System Signal Pin Assignments

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SD08	CN1A	7	Control data bus (Bit 8)	I/O
SD09	CN1A	22	Control data bus (Bit 9)	I/O
SD10	CN1A	8	Control data bus (Bit 10)	I/O
SD11	CN1A	23	Control data bus (Bit 11)	I/O
SD12	CN1A	9	Control data bus (Bit 12)	I/O
SD13	CN1A	24	Control data bus (Bit 13)	I/O
SD14	CN1A	10	Control data bus (Bit 14)	I/O
SD15	CN1A	25	Control data bus (Bit 15)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
SA08	CN1B	21	Control address bus (Bit 8)	I
SA09	CN1B	7	Control address bus (Bit 9)	I
SA10	CN1B	22	Control address bus (Bit 10)	I
MEN*	CN1A	14	Module enable	I
BHE*	CN1B	2	Control bus upper byte enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I

Note: An asterisk (*) following a signal name indicates inverse logic.

Table 1B: Control System Signal Pin Assignments

MECHANICAL DIMENSIONS

Figure 5 shows the board size and signal pin connector locations. The board size and connector positions conform to SIDIP standards.

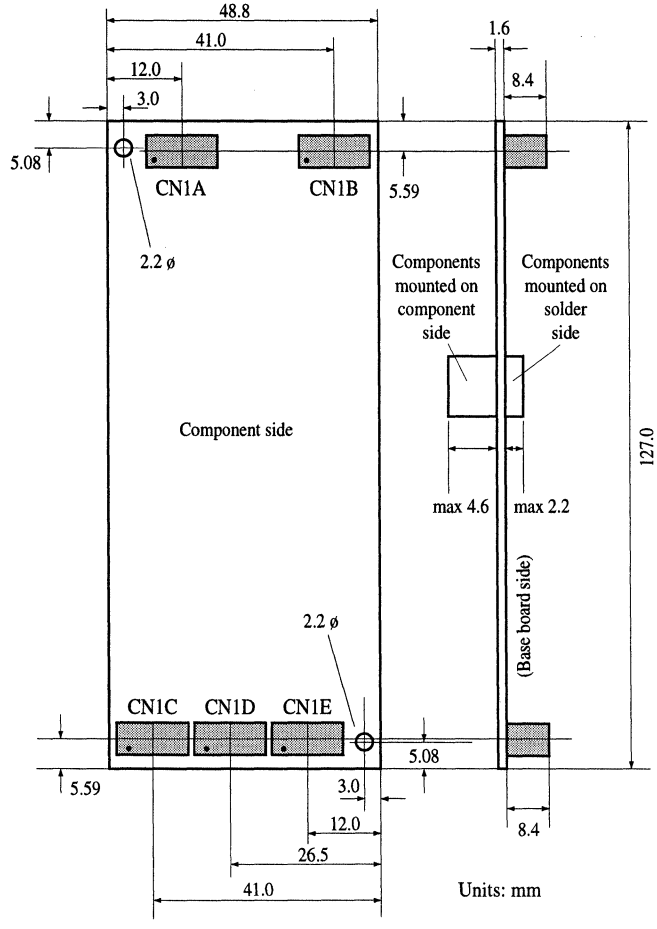


Figure 5: Board Size and Connector Positions

Notes

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PB-VF84-B (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B)

2: Connector pin number assignments:

•: #1 pin mark

3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (single-width).

5: Connector position measurements are to the center of the connector.

SIDIP Image Processing Modules

IP90MD10

Labeling Module

Product Manual

Ver. E 1.3

Sumitomo Metal Industries, Ltd.

SIDIP Labeling Module
IP90MD10

FEATURES AND FUNCTIONS

- High-speed labeling module based on the IP90C10 labeling processor LSI chip
- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- High-speed 25-MHz pixel clock
- Maximum processing area of 1024 x 1024 pixels
- Binary processing capacity with on-board input signal
- Same labeling processor functions as the IP90C10 chip:
  - up to 4094 temporary labels
  - up to 4094 primary labels
  - up to 4095 linkage information entries
- Processes one screen within the time span of 2 to 3 frames
- 4-link and 8-link processing
- Input image data width of 8-bits, with on-board binary conversion
- Output image data width of up to 12 bits (can be divided into lower 8-bit and upper 4-bit signals)
- Label tables can be loaded from the host computer
- Module ID can be read out
- Operates on a single 5V power source
- All terminal signal levels are TTL level
- Output compensation shift of 0-8 bits
- Module size of 98.6 x 127.0 mm (SIDIP double-wide size)

APPLICATIONS

- General image-processing devices
- Built-in image-processing devices for Factory Automation (FA) equipment
- Built-in image-processing devices for Office Automation (OA) equipment

PRODUCT DESCRIPTION

The IP90MD10 labeling processing module includes the IP90C10 labeling accelerator LSI (large-scale integration) chip, one of Sumitomo Industries' image-processing LSI series. The module also includes peripheral circuitry on a proprietary circuit board designed as a labeling processor module.

The module's bus architecture is compatible with the SIDIP standard interface for digital image processing proposed by Sumitomo Metal Industries. The architecture also includes bus interfaces compatible with other boards in the Sumitomo Metal Industries image processing module series.

The IP90MD10's labeling processor functions are identical to those of the IP90C10 chip. Primary labeling, label linkage processing, and secondary labeling are automatically executed in sequence and synchronized with the input image frames. (For further information about labeling, processing algorithms, and other details, refer to the technical documentation for the IP90C10 chip.)

The IP90MD10 module requires as input the pixels of a binary or 8-bit grayscale raster input image, together with the pixel clock and the horizontal and vertical synchronization signals synchronized with those pixels. The input image is converted into a labeled image of up to 12 bits, and output with the same pixel clock and horizontal and vertical synchronization signals as the input image. (For details of input and output images, see "Input/Output Images.")

The core component of the IP90MD10 module is the IP90C10 LSI chip. The basic specifications for the module's labeling processor functions are the same as those of the chip. The maximum input image that can be processed is 1024 pixels horizontally by 1024 pixels vertically, and the maximum clock frequency is 25 MHz.

The IP90MD10 module contains a comparator for binary conversion of grayscale input images, allowing the host computer to easily control the threshold value used for binary conversion. The module can handle a maximum output of 4094 labels with widths of up to 12 bits. This output can be divided into 8-bit lower and 4-bit upper values.

The IP90MD10 module also features a user-accessible memory for storing label tables, which allows label values to be assigned as needed.

The IP90MD10 module has its own module ID, which can be read by the host computer for easier configuration of systems. This ID can be used to determine whether the IP90MD10 is present, or for automatic system configuration.

The IP90MD10 module replaces previous low-speed software processing systems and bulky hardware, providing a space-saving, high-speed hardware device for easy histogram and projection processing. The module is designed for use in general-purpose image-processing devices, and particularly for built-in image-processing devices in factory automation applications that require high processing speeds. It is also ideally suited for built-in systems in office automation equipment such as copiers and fax machines.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD10 module can handle either binary-coded or 8-bit grayscale raster images (non-interleaved). The minimum system required for the IP90MD10 to perform label processing includes a pixel clock synchronized with the pixels that make up the raster image, as well as the horizontal and vertical synchronization signals necessary to construct the raster image (see Figure 1). These signals are synchronized with the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

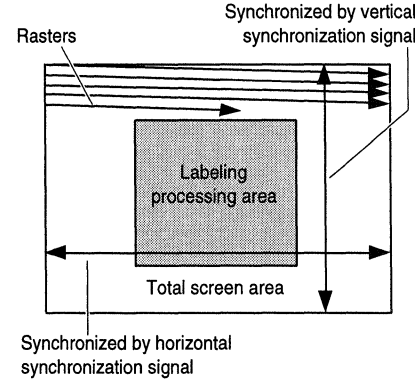


Figure 1. Definition of Image Area for Processing

The IP90MD10 module can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded area in Figure 1. The size and position of this area can be controlled from the host computer. In addition, areas outside this designated area can be considered background, and processed with a gradient value of 0 (or [0] in a binary image).

The output image is a labeled raster image, as is the input image. The output image has same pixel clock, horizontal synchronization signal, and vertical synchronization signal as the input image. However, an internal delay factor delays the output image by 8 pixels relative to the input image.

Figure 2 shows an example of an actual application.

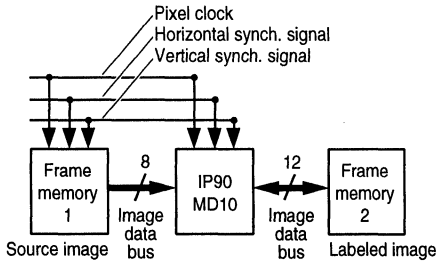


Figure 2. Sample Application

(2) Image Input Block

Figure 3 shows a block diagram of the IP90MD10 module.

Inverted image data is input at the input image bus (located at IN00-IN07). The input block inverts this data again, defines the area of the image to be processed (as shown in Figure 1), and performs binary conversion of 8-bit grayscale data. If binary data is input directly, the values used are 00h for [0] and FFh for [1].

(3) 1H Line Delay

The IP90MD10 module requires two consecutive lines of horizontal image data for labeling processing. Because incoming image data enters only in ranks of one line at a time, the module internally applies a line delay of one horizontal line.

(4) Labeling Chip Block

Labeling takes place in this block. The IP90C10 chip performs the necessary functions for labeling, and also controls peripheral circuits.

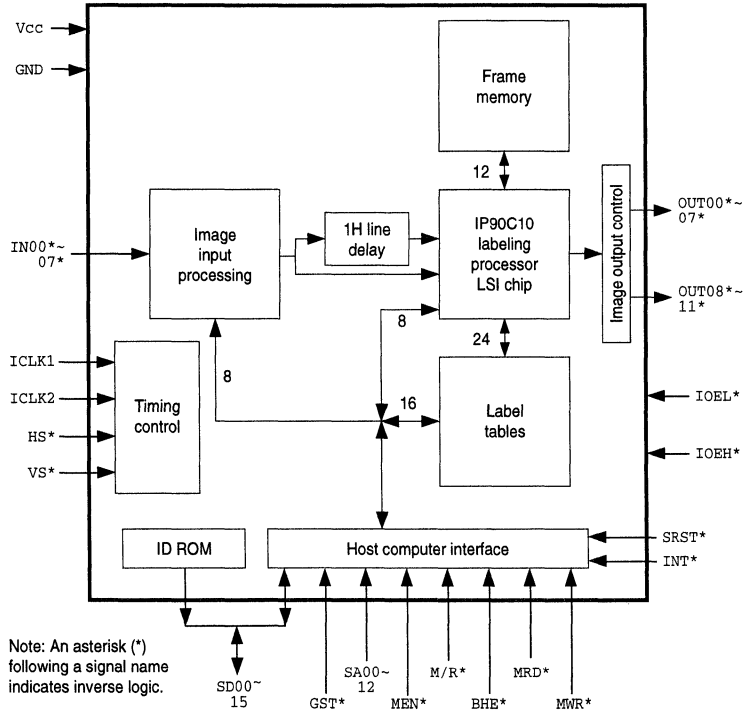


Figure 3. Block Diagram

(5) Label Table Block

This block holds data linked with the temporary labels assigned in the primary labeling process. The block also contains the tables used to convert the temporary labels created in the label linkage process into final labels. These tables are used in the secondary labeling process, in which the temporary labeled image is converted to a final labeled image. Label tables are controlled by the IP90C10 chip.

(6) Frame Memory Block

This block contains the temporary labeled image from the primary labeling process. The temporary labeled image is then read out for secondary labeling. This activity is controlled by the IP90C10 chip. The size of frame memory is 1024 horizontal pixels by 1024 vertical pixels by 12-bit data width, equivalent to one frame of data.

(7) Image Output Control Block

The labeled image is output to the image bus with a bit width of 12 bits (consisting of lower 8-bit and upper 4-bit signals, each in three-state output with independently controlled connection status), and with inverted signal logic.

(8) Timing Control Block

The timing control block is the point of entry for the pixel clocks (ICLK1 and ICLK2), horizontal synchronization signal (HS*), and vertical synchronization signal (VS*).

At frequencies higher than 20 MHz, a two-phase clock input is created using both pixel clocks ICLK1 and ICLK2. When the pixel clock frequency is 20 MHz or lower, only ICLK1 is used.

(9) Host Interface Block

The host interface block contains the interface for the computer that controls the IP90MD10 module.

The data bus (SD00-15) is 16 bits wide, and the address bus (SA00-12) is 13 bits wide. The MWR* and the MRD* signals control write and read operations, respectively. Byte access and word access are obtained by combining address A0 and the *BHE signal. The M/R* signal controls memory/register access.

(a) ID Number Reading

The GST* signal outputs the ID number of the IP90MD10 module (04h) to the data bus.

(b) Interrupt Signal

The interrupt signal becomes active when an error occurs in labeling processing, such as an overflow in the number of temporary labels or temporary label linkages. (For further details about these errors, see the technical documentation for the IP90C10 LSI chip.)

(c) System Reset

The SRST* signal resets all registers in the module to 00h.

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD10 module. Signal pins are divided between the image system (Table 1A) and control system (Table 1B), having three and two external connectors, respectively. These pin assignments conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
OUT00*	CN1C	12	Output image data bus (Bit 0)	O
OUT01*	CN1C	27	Output image data bus (Bit 1)	O
OUT02*	CN1C	13	Output image data bus (Bit 2)	O
OUT03*	CN1C	28	Output image data bus (Bit 3)	O
OUT04*	CN1C	14	Output image data bus (Bit 4)	O
OUT05*	CN1C	29	Output image data bus (Bit 5)	O
OUT06*	CN1C	15	Output image data bus (Bit 6)	O
OUT07*	CN1C	30	Output image data bus (Bit 7)	O
OUT08*	CN1D	2	Output image data bus (Bit 8)	O
OUT09*	CN1D	17	Output image data bus (Bit 9)	O
OUT10*	CN1D	3	Output image data bus (Bit 10)	O
OUT11*	CN1D	18	Output image data bus (Bit 11)	O
IOEL*	CN1E	2	Image bus output enable (lower)	I
IOEH*	CN1E	17	Image bus output enable (upper)	I

Table 1A. Image System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SD08	CNIA	7	Control data bus (Bit 8)	I/O
SD09	CN1A	22	Control data bus (Bit 9)	I/O
SD10	CNIA	8	Control data bus (Bit 10)	I/O
SD11	CN1A	23	Control data bus (Bit 11)	I/O
SD12	CNIA	9	Control data bus (Bit 12)	I/O
SD13	CN1A	24	Control data bus (Bit 13)	I/O
SD14	CN1A	10	Control data bus (Bit 14)	I/O
SD15	CNIA	25	Control data bus (Bit 15)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
SA08	CN1B	21	Control address bus (Bit 8)	I
SA09	CN1B	7	Control address bus (Bit 9)	I
SA10	CN1B	22	Control address bus (Bit 10)	I
SA11	CN1B	8	Control address bus (Bit 11)	I
SA12	CN1B	23	Control address bus (Bit 12)	I
MEN*	CN1A	14	Module enable	I
BHE*	CN1B	2	Control bus upper byte enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
INT*	CN1A	27	Interrupt request signal	O
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I

Table 1B. Control System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

MECHANICAL DIMENSIONS

Figure 4 shows the IP90MD10's board size and signal pin connector locations. The size and locations conform to SIDIP standards.

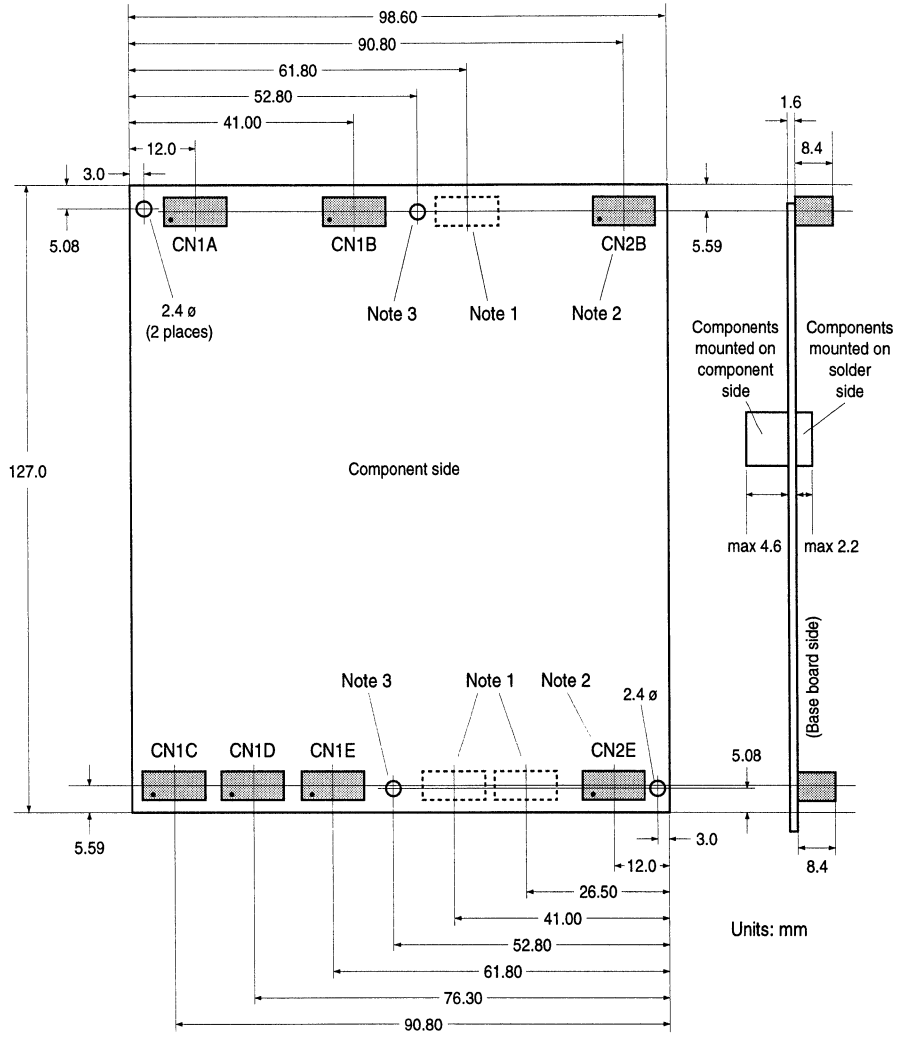


Figure 4. Board Size and Connector Positions

Notes

1: The IP90MD10 module has no connectors CN2A, CN2C, or CN2D.

2: The labeling processor board uses these two connectors (CN2B, CN2E) as dummy connectors (no electrical function) to hold the board in position.

3: The IP90MD10 module has no holes at these positions.

Additional Information

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PB-VF84-B. (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B.)

2: Connector pin number assignments:

component side: •: #1 pin mark

3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (double-wide).

5: Connector position measurements are to the center of the connector.

SIDIP Image Processing Modules

IP90MD15

Sketch Module

Image Data Reduction by Averaging/
Simple Enlargement Module

Product Manual

Ver. E 1.2

Sumitomo Metal Industries, Ltd.

SIDIP Image Data Reduction by Averaging/ Simple Enlargement Sketch Module IP90MD15

FEATURES AND FUNCTIONS

- Exceptionally easy-to-use module based on the IP90C15 LSI chip
- Performs simple horizontal and vertical enlargement
- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- Maximum pixel clock frequency of 25 MHz
- Maximum processing area of 1024 x 1024 pixels
- Provides compression ratios of 1/2, 1/4, 1/8, and 1/16
- Provides enlargement ratios of 2, 4, 8, and 16
- Input and output image data bus widths of 8 bits
- Variable processing area
- Variable processing frame count (maximum of 64 frames, with continuous loop processing available)
- Variable interval frame count setting allows frame processing at regular intervals
- Provides tiling of average-value compressed images in frame memory
- Module ID can be read out
- Operates on a single 5V power source
- All terminal signal levels are TTL level
- Module size of 98.6 x 127.0 mm (SIDIP double-wide size)

APPLICATIONS

- Built-in image-processing devices for security monitoring systems
- Dynamic image-processing devices
- General image-processing devices
- Built-in image-processing devices for Factory Automation (FA) equipment
- Built-in image-processing devices for Office Automation (OA) equipment

PRODUCT DESCRIPTION

The IP90MD15 module contains the Sumitomo Metal Industries IP90C15 LSI (large scale integration) chip, one of Sumitomo Industries' image-processing LSI series. The IP90MD15 also includes additional peripheral circuitry in a compact, exceptionally easy-to-use processing module.

The module's bus architecture is compatible with the SIDIP standard interface for digital image processing proposed by Sumitomo Metal Industries. The architecture also includes bus interfaces compatible with other boards in Sumitomo Industries' image-processing module series.

The IP90MD15 module processes image data in real time, synchronized with the frames of the input image. It can also initialize internal frame memory. (For information about the algorithms the IP90MD15 uses to reduce image data by averaging, see the technical documentation for the IP90C15 chip.)

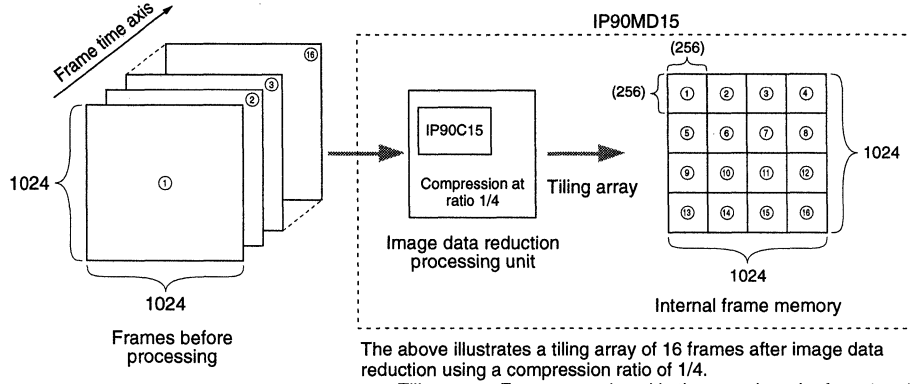


Figure 1. Schematic Diagram: Image Data Reduction by Averaging Processing and Tiling Array

The IP90MD15 module requires only input of the pixels of an 8-bit grayscale input image, together with the pixel clock and the horizontal and vertical synchronization signals synchronized with those pixels. After processing, the image is output with either the same pixel clock as the input image (in 1-phase clock mode) or with different phases (in 2-phase clock mode). The output image's horizontal and vertical synchronization signals are the same as the input image's. (For details about the input image, see the description of module operation.)

The core component of the IP90MD15 module is the IP90C15 LSI chip, which can process images with a maximum size of 1024 pixels horizontally by 1024 pixels vertically, and which has a maximum clock frequency of 25 MHz.

The IP90MD15 module places the results of image data reduction into a tiling array in internal frame memory. This is an efficient way to simultaneously display multiple compressed images on the output monitor (see Figure 1). The module offers a variable "interval frame count" setting, which establishes a fixed number of frames between each frame selected for data reduction. Continuous data reduction processing is also available. The module's internal frame memory can store up to 64 processed frames at once.

The IP90MD15 can also enlarge input images horizontally and vertically (see Figure 2).

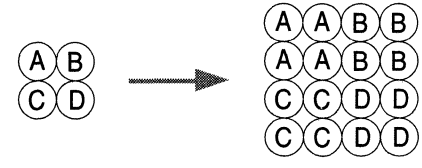


Figure 2. Schematic Diagram: Simple Enlargement

The IP90MD15 module has its own module ID, which can be read by the host computer for easier configuration of systems. This ID can be used to determine whether the IP90MD15 is present, or for automatic system configuration.

The IP90MD15 module replaces previous low-speed software processing systems and bulky hardware, providing a space-saving, high-speed hardware device for easy image data reduction or enlargement. The module is designed for use in general-purpose image-processing devices, particularly for the high processing speeds demanded by dynamic image-processing systems, as well as for built-in image-processing devices used in FA equipment, security monitoring equipment, and OA equipment.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD15 module processes 8-bit grayscale raster images (non-interleaved). The minimum system required by the IP90MD15 includes a pixel clock synchronized with the pixels that make up the raster image, as well as the horizontal and vertical synchronization signals necessary to construct the raster image (see Figure 3). These signals are synchronized with the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

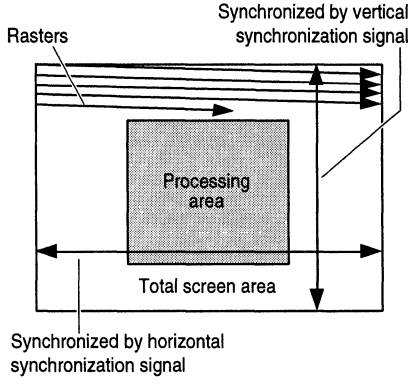


Figure 3. Definition of Image Area for Processing

The IP90MD15 module can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded area in Figure 1. The size and position of this area can be controlled from the host computer. Areas outside this designated area can be considered background, and processed with a gradient value of 0.

The output image is a raster image, as is the input image. The output image has same pixel clock and horizontal and vertical synchronization signals as the input image.

Figure 4 shows an example of an actual application.

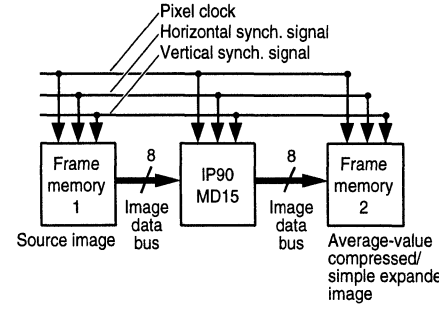


Figure 4. Sample Application

(2) Image Input Block

Figure 5 shows a block diagram of the IP90MD15 module.

Inverted image data enters at the input image bus (located at IN00-IN07). The input block inverts this data again, and defines the area of the image to be processed.

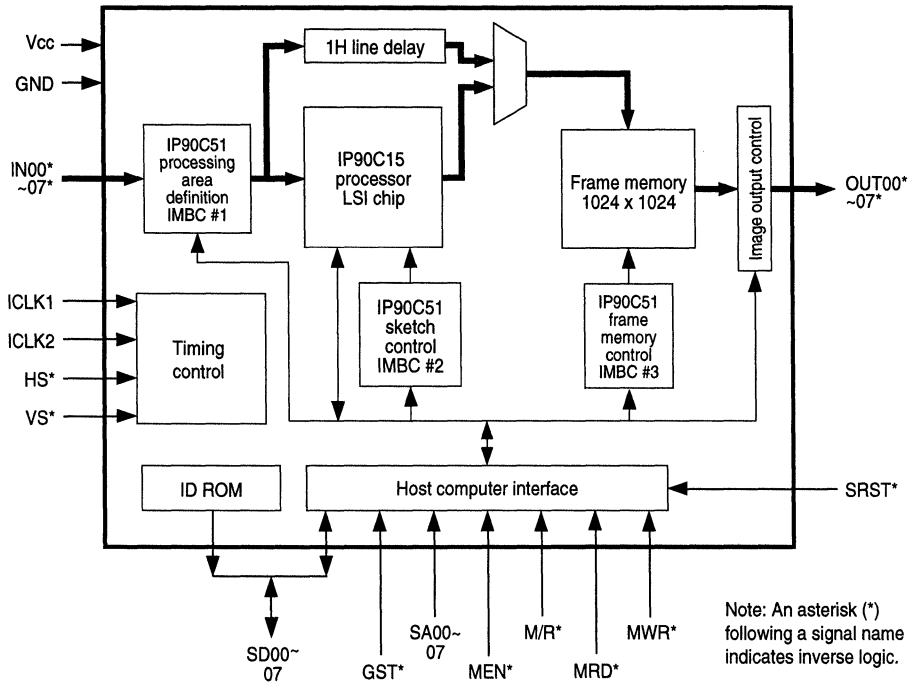


Figure 5. Block Diagram

(3) Image Data Reduction by Averaging Processor

The core of the IP90MD15 module is Sumitomo Industries' IP90C15 LSI chip. The IP90MD15 provides a variable compression ratio of 1/2, 1/4, 1/8, or 1/16, as well as variable settings for processing frame count and interval frame count.

Whenever a compressed image is smaller than the assigned tiling size, the module fills the remaining area with the gradient value 0. (For more information, see Table 1, the explanation of tiling in item (4), the explanation of processing modes in item (5), and Figure 7.)

(4) Tiling

Compressed images can be placed in a tiling array in internal frame memory (1024 x 1024 pixels). The host computer controls the point of origin and the direction (vertical or horizontal) of tiling. Processed images can also be placed in the same tiling location.

Figure 8 shows examples of tiling arrays, and Table 1 lists image sizes used in tiling. When a processing area of 1024 x 1024 pixels is compressed to 1/4 size, the size of the image in a tiling array is 256 x 256 pixels, and a total of 16 frames of average-value compressed images can be placed in the array. The point of origin can be any point from (0, 0) to (3, 3).

The minimum image size for tiling is 64 x 64 pixels. In some cases, image data reduction by averaging can produce images that are smaller than 64 x 64 pixels (such as 32 x 32 pixels), in which case the result is placed starting in the upper left corner of a 64 x 64 pixel area for tiling, and part of the tiling area is undefined (see Figure 6).

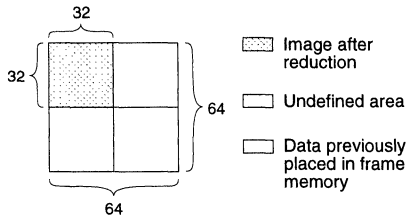


Figure 6. Internal Frame Memory Array When the Result of Image Data Reduction is Smaller Than 64 x 64 Pixels

(5) Processing Modes

Image data reduction can be applied to a frame count of 1 to 64 frames. Also, a number of frames can be skipped between processed frames, as specified by an interval frame count of 0 to 1023 frames.

The IP90MD15 can operate either in "one-time mode," in which processing stops automatically after a designated number of frames, or in "loopback mode," in which processing continues as described above until the host computer sends a stop command. The processing time required (using an interval frame count of 0 frames) is equal to the frame count plus one (see Figure 9).

(6) Simple Enlargement Processing

The IP90MD15 performs simple enlargement processing by replicating pixel data along the horizontal or vertical axis, with the number of replications corresponding to the enlargement ratio (this is the zero-order hold method). An internal delay factor of one horizontal line is applied so that a line of horizontal pixel data can be replicated next to its original position. This process begins at the upper left corner of the processing area, and operates using an enlargement factor of 2, 4, 8, or 16. If the resulting image is smaller than the module's internal frame memory, the remaining area is filled in with gradient value 0.

If the resulting image is larger than the internal frame memory, the excess portions of the image cannot be stored or displayed (see Table 1 and Figure 10).

(7) Image Output Control Block

The processed image is output to the image bus with a width of 8 bits, in three-state output with independently controlled connection status, and with inverted signal logic.

(8) Timing Control Block

The timing control block is the point of entry for the pixel clocks (ICLK1 and ICLK2), horizontal synchronization signal (HS*), and vertical synchronization signal (VS*).

At frequencies higher than 20 MHz, a two-phase clock input is created using both pixel clocks ICLK1 and ICLK2. When the pixel clock frequency is 20 MHz or lower, only ICLK1 is used.

(9) Host Interface Block

The host interface block contains the interface for the computer that controls the IP90MD15 module.

The data bus (SD00-07) and the address bus (SA00-07) are both 8 bits wide. The MWR* and the MRD* signals control write and read operations, respectively. The M/R* signal at low level controls memory/register access.

(a) ID Number Reading

The GST* signal outputs the IP90MD15's 8-bit ID number (10h) to the data bus.

(b) System Reset

The SRST* signal resets all registers in the module to 00h.

Area processed	Compression ratio	Tiled image size	Enlargement ratio
1024 x 1024	1/2	512 x 512	
through	1/4	256 x 256	
513 x 513	1/8	128 x 128	
	1/16	64 x 64	
512 x 512	1/2	256 x 256	x 2
through	1/4	128 x 128	
257 x 257	1/8	64 x 64	through
	1/16	64 x 64*	
256 x 256	1/2	128 x 128	x 16
through	1/4	64 x 64	
129 x 129	1/8	64 x 64*	
128 x 128	1/2	64 x 64	
through 65 x 65	1/4	64 x 64*	
64 x 64 or less	1/2	64 x 64*	

* See description of tiling in section (4).

Table 1. Image Data Reduction by Averaging/Simple Enlargement Ratios

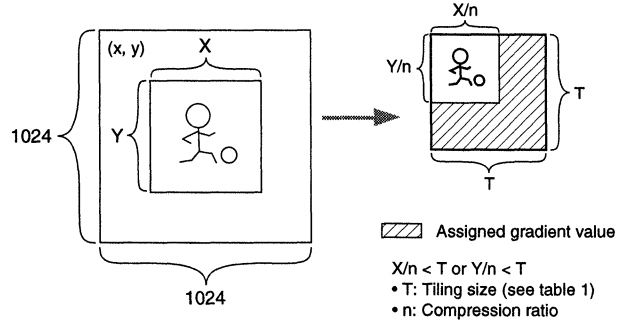


Figure 7. Example of Image Data Reduction by Averaging

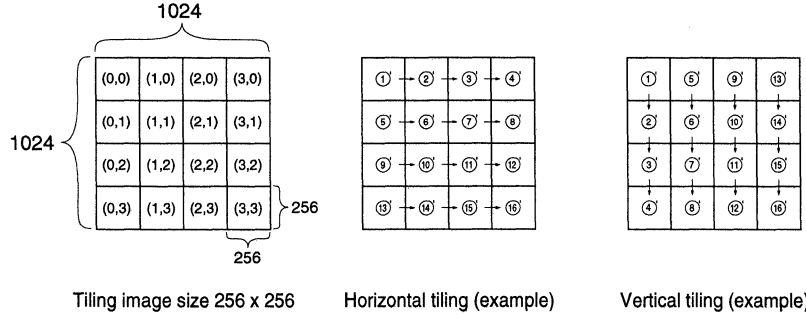
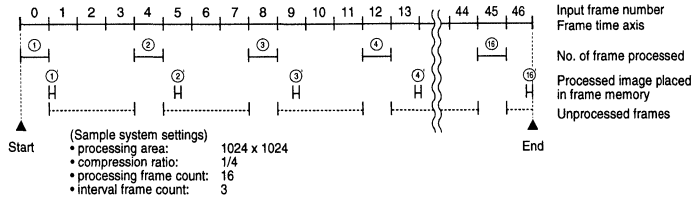
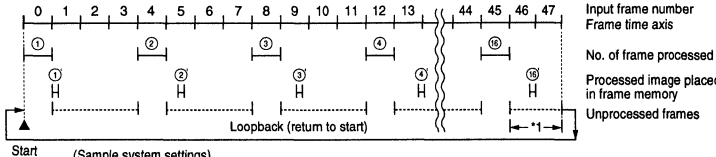


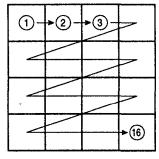
Figure 8. Examples of Tiling Arrays



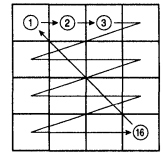
(A): Processing in One-Time Mode (Example)



(B): Processing in Loopback Mode (Example)



(C): Sample Tiling Array in One-Time Mode (see A)



(D): Sample Tiling Array in Loopback Mode (see B)

- In example (A), processing ends once the processing frame count is reached.
- In example (B), processing of the designated processing frame count repeats until a stop signal is received from the host computer.
- Frames 1 through 16 are the frames input to the IP90MD15 for processing, and frames 1' through 16' are the processed images placed in the tiling array in frame memory.
- (C) is an example showing placement of the results of example (A) in internal frame memory in a horizontal tiling array from 1' 2' 3' through 16'.
- (D) is an example showing placement of the results of example (B) in internal frame memory in a repeating horizontal tiling array from 1' 2' 3' through 16', then again 1' 2' 3' through 16', 1' ... etc.

Figure 9. Examples of Processing Modes

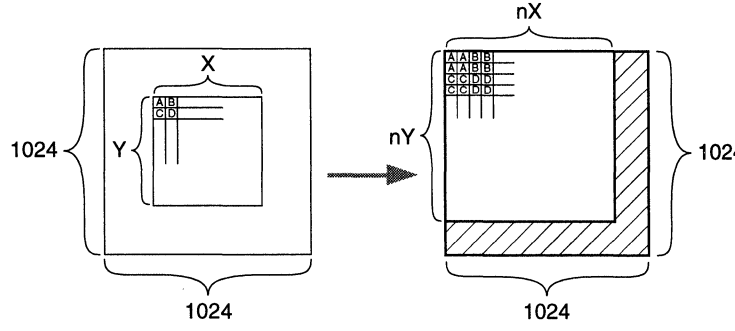


Figure 10. Example of Simple Enlargement

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD15 module. Signal pins are divided between the image system (Table 2A) and control system (Table 2B), having three and two external connectors, respectively. These pin assignments conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
OUT00*	CN1C	12	Output image data bus (Bit 0)	O
OUT01*	CN1C	27	Output image data bus (Bit 1)	O
OUT02*	CN1C	13	Output image data bus (Bit 2)	O
OUT03*	CN1C	28	Output image data bus (Bit 3)	O
OUT04*	CN1C	14	Output image data bus (Bit 4)	O
OUT05*	CN1C	29	Output image data bus (Bit 5)	O
OUT06*	CN1C	15	Output image data bus (Bit 6)	O
OUT07*	CN1C	30	Output image data bus (Bit 7)	O
IOEL*	CN1E	2	Image bus output enable (lower)	I

Table 2A. Image System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
MEN*	CN1A	14	Module enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I

Table 2B. Control System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

MECHANICAL DIMENSIONS

Figure 11 shows the IP90MD15's board size and signal pin connector locations. The size and locations conform to SIDIP standards.

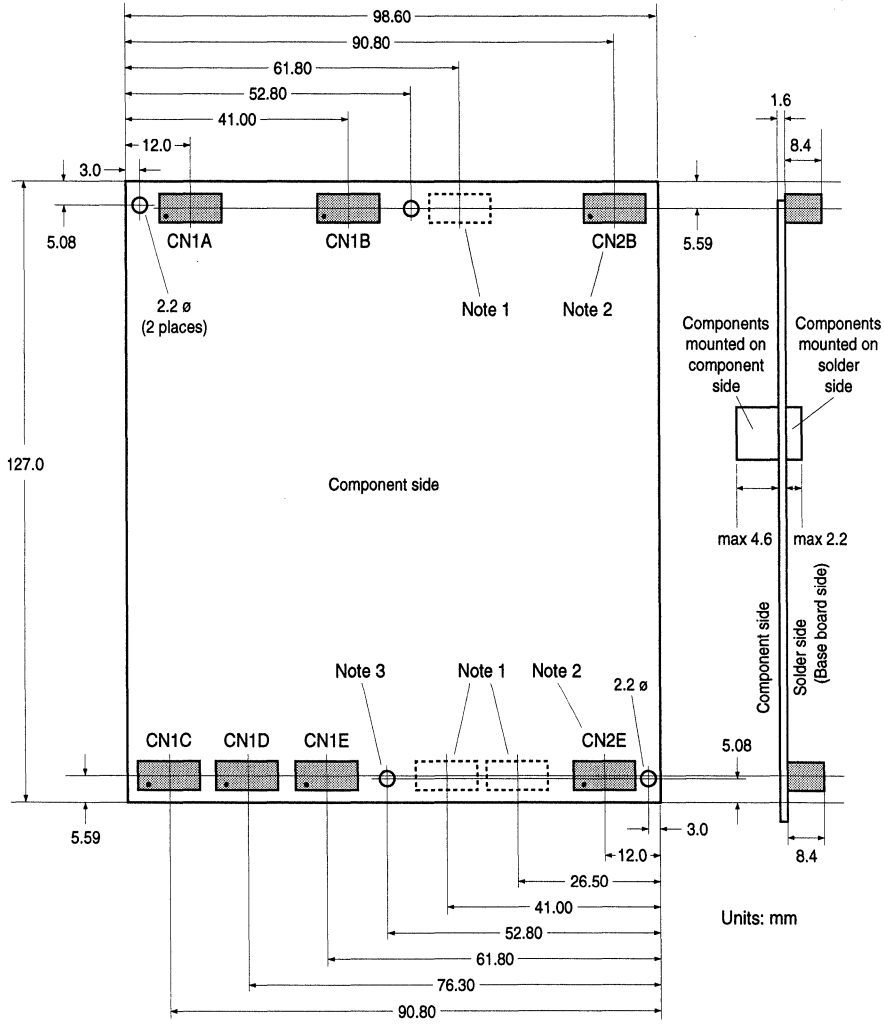


Figure 11. Board Size and Connector Positions

Notes

1: The IP90MD15 module has no connectors CN2A, CN2C, or CN2D.

2: The module board uses these two connectors (CN2B, CN2E) as dummy connectors (no electrical function) to hold the board in position.

3: The IP90MD15 module has no hole at this position.

Additional Information

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PB-VF84-B (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B)

2: Connector pin number assignments:

component side: 

•: #1 pin mark

3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (double-wide).

5: Connector position measurements are to the center of the connector.

SIDIP Image Processing Modules

IP90MD20

Rank Value Filter Module

Product Manual

Ver. E 1.2

Sumitomo Metal Industries, Ltd.

SIDIP Rank Value Filter Module
IP90MD20

FEATURES AND FUNCTIONS

- High-speed rank value filter processor module built around a rank value filter processor LSI (IP90C20) core
- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- Maximum pixel clock frequency of 25 MHz
- Maximum processing area of 2048 (horizontal) x 4095 (vertical) pixels
- Rank Value filter processing characteristics identical to those of the IP90C20 LSI

  Maximum/median/minimum filter processing for nine (3 x 3) local areas (median filter can be applied to 1 x 3 or 3 x 1 local areas)
- 8-bit input image data width

  Enables binary conversion within module
- 8-bit output image data width
- Module ID can be read out
- Operates on single 5V power source
- All terminal signals at TTL level
- Module size 48.8 x 127.0 mm (SIDIP single-wide size)

APPLICATIONS

- Noise reduction, expansion, compression
- General image-processing devices
- Built-in image-processing devices for factory automation (FA) equipment
- Built-in image-processing devices for office automation (OA) equipment

PRODUCT OVERVIEW

The core of the IP90MD20 rank value filter processor module is the LSI (IP90C20) rank value filter processor, from Sumitomo Metal Industries. Additional peripheral circuits are added to the processor to form a board designed exclusively for rank value filter processing. The module also features a bus architecture compatible with the SIDIP standards proposed by Sumitomo Metal Industries, and thereby provides a common bus interface with other image-processing modules in the same series.

The IP90MD20 performs rank value filter processing in exactly the same manner as the IP90C20 core. Maximum, median and minimum values set by the host computer are outputted at high speed concurrently with throughput at the input image frequency. (See the technical documentation for the IP90C20 for information on rank value filter processing algorithms.)

The IP90MD20 receives data from the raster image of a grayscale (8-bit) image, which enters the system accompanied by a synchronizing pixel clock signal and the horizontal and vertical synchronization signals from the image. The IP90MD20 then performs rank value filter processing and outputs the result using the same pixel clock and horizontal and vertical synchronization signals as the input image. (See the description of the IP90MD20 operation on the next page for a description of the input and output images.)

The maximum image size for processing is 2048 pixels horizontally by 4095 pixels vertically. The maximum clock frequency is 25 MHz.

Although the IP90MD20 is a rank value filter processing module, the image input block includes a Sumitomo Metal Industries image data bus controller IP90C51: which can generate binary or ternary output. When the host computer sets the threshold values and the IP90C20 to the transparent mode , the IP90MD20 outputs the binary or ternary image from an incoming grayscale image.

To facilitate system configuration, the IP90MD20 has a module ID that can be read by the host computer to determine whether the IP90MD20 is present in the system, or as an identifier for automatic system configuration.

The IP90MD20 replaces bulky hardware processing devices and slow, outdated software processes, and provides easy rank filter processing through compact, high-speed hardware. The IP90MD20 can be used in general image-processing devices, particularly built-in image processing components in FA applications requiring high-speed processing, as well as in office equipment such as copiers and facsimiles.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD20 can handle either binary-coded or 8-bit grayscale raster images (non-interleaved). The minimum system required for the IP90MD20 to perform rank value filtering includes pixel clocks synchronized with the pixels that make up the raster image, as well as the horizontal and vertical synchronization signals necessary to structure the raster image (see Figure 1). The horizontal and vertical synchronization signals are synchronized by the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

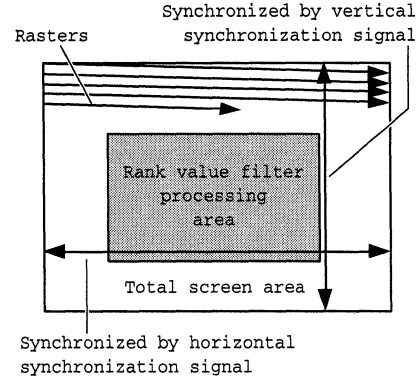


Figure 1. Definition of Image Area for Processing

The IP90MD20 can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded rectangular area in Figure 1. The size and position of this area can be controlled from the host computer. In addition, areas outside this designated area can be considered background and processed in gradients (0-255) controlled from the host computer.

The output image is a filtered raster image, as is the input image. The pixel clock and the horizontal and vertical synchronization signals for the output image are the same as for the input image. However, the internal delay factor causes the output image to be delayed by 16 pixels on the horizontal axis, and by one line on the vertical axis.

Figure 2 shows an example of an actual application.

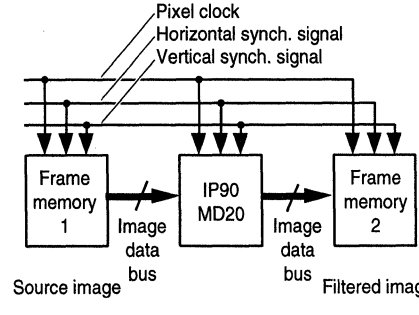


Figure 2. Sample Application

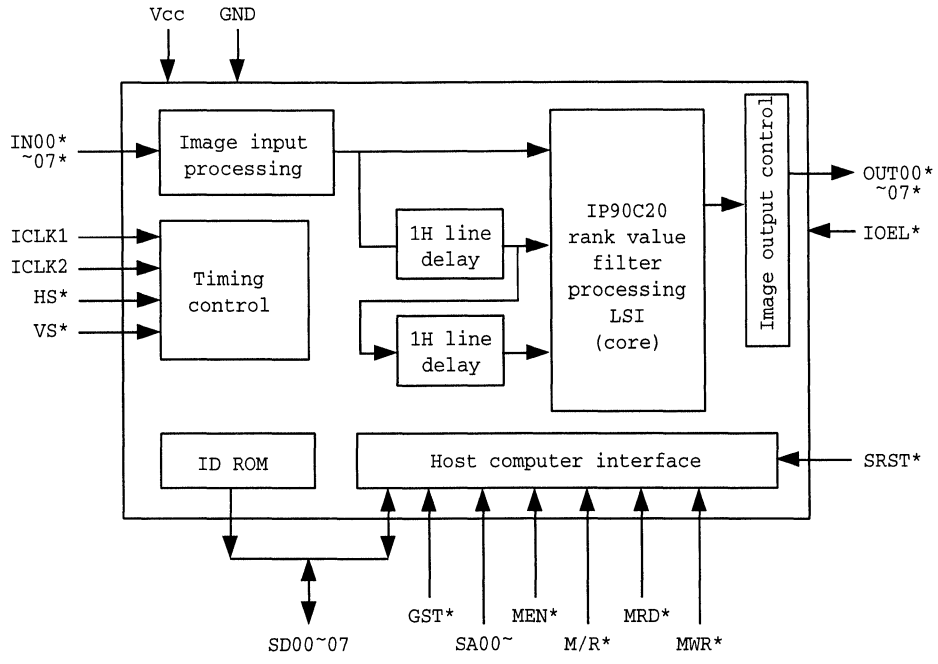


Figure 3. Block Diagram

(2) Image Input Block

Figure 3 shows a block diagram of the IP90MD20. (Note: an asterisk [*] following a signal name indicates inverse logic.)

Inverted image data is input at the input image bus (located at IN00-IN07). At the input block this data is inverted again, the previously described designation of the processing area takes place, and, if binary processing is used, 8-bit grayscale data is converted to binary data. Binary data is converted to 8-bit logic using the values 00h for [0] and FFh for [1]. Note that, because the bus carries values that are inverted for processing, data entering the module appears in the bus as FFh for [0] and 00h for [1].

(3) 1H Line Delay

Rank value filter processing in the IP90MD20 requires three successive lines of horizontal image data. Because incoming image data enters in ranks of one line at a time, the module meets the processing requirement by applying a line delay of one horizontal line, in two iterations.

(4) Core Block

The rank value filter LSI (IP90C20) is used in its normal role, performing the necessary functions for rank value filter processing.

(5) Image Output Control Block

Following rank value filter processing, the image is output to the image bus with a width of 8 bits, and with signal logic still inverted. The output signal is in three states, enabling control over the connection.

(6) Timing Control Block

This block is the point of entry for the pixel clocks (ICLK1, ICLK2), horizontal synchronization signal (HS*), and vertical synchronization signal (VS*).

When the pixel clock frequency is higher than 20 MHz, both ICLK1 and ICLK2 are used. When the frequency is 20 MHz or lower, only ICLK1 is used.

(7) Host Interface Block

The host interface provides control over the IP90MD20. Its data bus (SD00-07) is 8 bits wide, and its address bus (SA00-11) is 12 bits wide. Write operations are controlled by the MWR* signal, and read operations by the MRD* signal. The registers on the IP90MD20 are write-only, and they are accessed by the M/R* signal.

a) ID Number Reading

The GST* signal outputs the IP90MD20's ID number (08h) through the data bus.

b) System Reset

The SRST* signal causes a hardware reset, which returns all registers on the module to 00h.

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD20. Signal pins are divided between the image system and control system, having three and two external connectors, respectively. These signal pin assignments conform to SIDIP standards.

Table 1A lists the signal pin assignments for the image system, and Table 1B lists signal pin assignments for the control system.

MECHANICAL DIMENSIONS

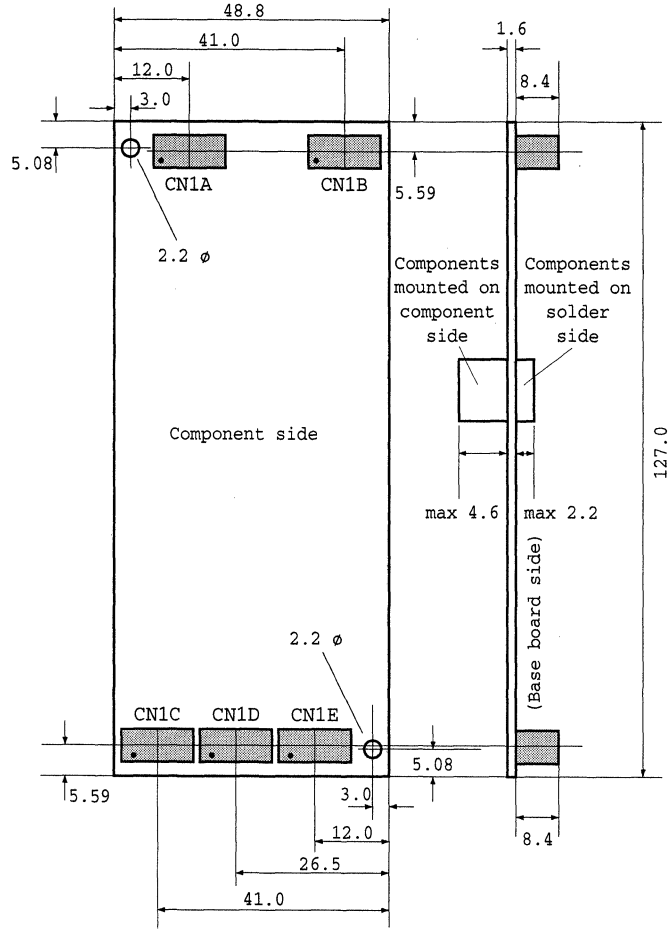
Figure 4 shows the board size and signal pin connector locations. The board size and connector positions conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
OUT00*	CN1C	12	Output image data bus (Bit 0)	O
OUT01*	CN1C	27	Output image data bus (Bit 1)	O
OUT02*	CN1C	13	Output image data bus (Bit 2)	O
OUT03*	CN1C	28	Output image data bus (Bit 3)	O
OUT04*	CN1C	14	Output image data bus (Bit 4)	O
OUT05*	CN1C	29	Output image data bus (Bit 5)	O
OUT06*	CN1C	15	Output image data bus (Bit 6)	O
OUT07*	CN1C	30	Output image data bus (Bit 7)	O
IOEL*	CN1E	2	Image bus output enable (lower)	I

Table 1A. Image System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
SA08	CN1B	21	Control address bus (Bit 8)	I
SA09	CN1B	7	Control address bus (Bit 9)	I
SA10	CN1B	22	Control address bus (Bit 10)	I
SA11	CN1B	8	Control address bus (Bit 11)	I
MEN*	CN1A	14	Module enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I

Table 1B. Control System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.



Note 1: Units: mm

Figure 4. Board Size and Connector Positions

Notes

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PE-VF84-B (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B)

2: Connector pin number assignments:

•: #1 pin mark

3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (single-wide).

5: Connector position measurements are to center of connector.

SIDIP Image Processing Modules

IP90MD25

Spatial & Logical Filter Module

Product Manual

Ver. E 1.1

Sumitomo Metal Industries, Ltd.

SIDIP Spatial & Logical Filter Module
IP90MD25

FEATURES AND FUNCTIONS

- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- High-speed filter built around an IP90C25 chip
- Broad internal computation range with no overflow
- Binary processing capacity with on-board input signal
- 3 x 3-pixel local area size
- 8-bit input/output image data width
- 8-bit, encoded index value/addition constant
- 0-8 bit output compensation shift
- Maximum processing area of 2048 (horizontal) x 4095 (vertical) pixels
- 25 MHz maximum operating frequency
- Control software reference source list
- Module ID can be read out
- Operates on a single 5V power source
- All terminal signal levels are TTL level
- Module size 48.8 x 127.0 mm (SIDIP single-wide size)

APPLICATIONS

- Built-in image-processing devices for factory automation (FA) equipment
- Built-in image-processing devices for office automation (OA) equipment
- Dynamic image-processing devices
- Option board for general image-processing devices
- IP90C25, IP90C51 evaluation

PRODUCT OVERVIEW

The core of the IP90MD25 spatial and logical processing module is the Sumitomo Metal Industries IP90C25 spatial and logical filter LSI (large scale integration) chip. This module is compatible with the SIDIP standard interface for digital image processing proposed by Sumitomo Metal Industries, and can perform spatial and logical filter processing of 3 x 3-pixel local areas at pixel clock speeds of up to 25 MHz.

The module has two basic operating modes:

- A spatial filtering mode for processing 8-bit grayscale images using a programmable filter coefficient. (The filter's value is expressed in 8-bit, 2-complement format.)
- A logical filtering mode for processing binary images using user-defined logical expressions. These expressions are designated by writing truth tables into a look-up table (LUT), which allows new filters to be created by rewriting the LUT.

(For further details about filter processing, refer to the technical documentation for the IP90C25 chip.)

The functions provided by the IP90MD25 module were previously only available by using bulky equipment. In contrast, the 48.8 x 127.0-mm (SIDIP single-size) module is designed for use as an add-on daughter board, and need only be connected to a CPU bus for control signals and to digital image input/output terminals to provide real-time spatial and logical filtering.

The SIDIP module series features a base board with connections for VME, ISA, and other interfaces. These allow individual SIDIP add-on modules to be connected as daughter boards for easy configuration of image processing systems.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD25 module can handle either binary-coded or 8-bit grayscale raster images (non-interleaved). The minimum system required for the IP90MD25 to perform spatial and logical filtering includes a pixel clock synchronized with the pixels that make up the raster image, as well as the horizontal and vertical synchronization signals necessary to construct the raster image (see Figure 1). These synchronization signals are synchronized with the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

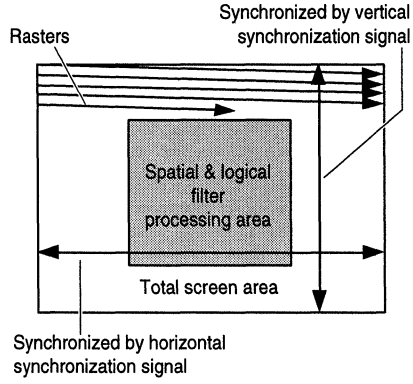


Figure 1. Definition of Image Area for Processing

The IP90MD25 module can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded area in Figure 1. The size and position of this area can be controlled from the host computer. In addition, areas outside this area can be considered background and processed in gradients (0-255) designated by the host computer.

The output image is a filtered raster image, as is the input image. At this point, the pixel clocks and the horizontal and vertical synchronization signals are the same as for the input image. However, the internal delay factor delays the output image by 15 to 39 rows relative to the input image, depending on the operating mode.

Figure 2 shows an example of an actual application.

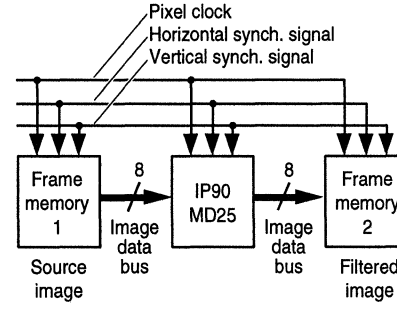


Figure 2. Sample Application

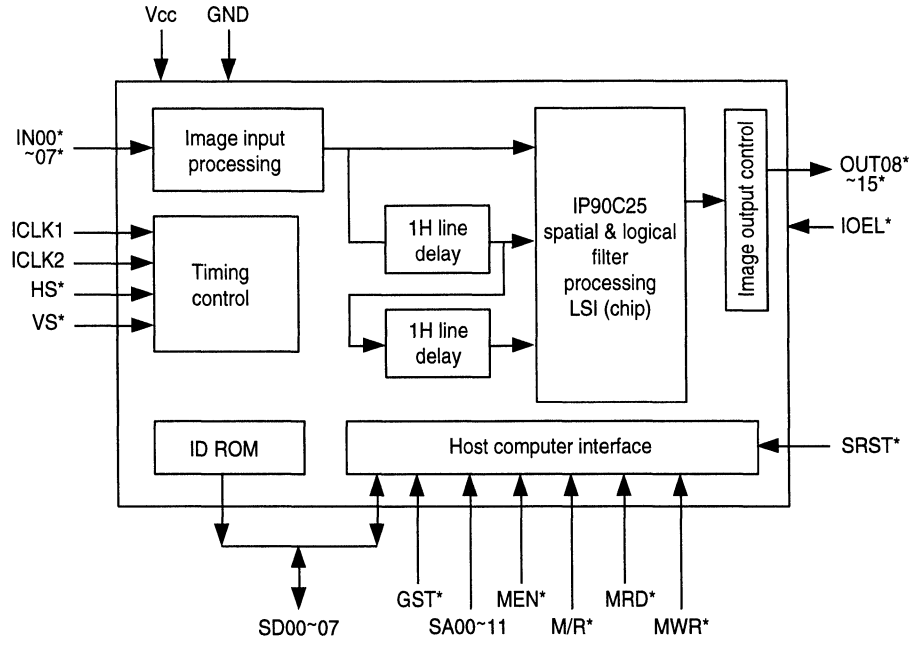


Figure 3. Block Diagram

(2) Image Input Block

Figure 3 shows a block diagram of the IP90MD25 module. (Note: an asterisk [*] following a signal name indicates inverse logic.)

Inverted image data is input at the input image bus (located at IN00-IN07). The input block inverts this data again, and also converts 8-bit grayscale data to binary data if binary processing is used. Binary data is converted to 8-bit logic using the values 00h for [0] and FFh for [1]. Note that because the bus carries values that are inverted for processing, data entering the module appears in the bus as FFh for [0] and 00h for [1].

(3) 1H Line Delay

Rank filter processing in the IP90MD25 requires three successive lines of horizontal image data. Because incoming image data enters in ranks of one line at a time, the module meets the processing requirement by applying a line delay of one horizontal line, in two iterations.

(4) Chip Block

This is the block in which filter processing takes place. The spatial and logical filter LSI (IP90C25) performs the necessary functions for spatial and logical filter processing.

(5) Image Output Control Block

The filtered image is output to the image bus with a width of 8 bits, and with signal logic inverted. The output signal is in three states, enabling connection to tri-state buses in other modules.

(6) Timing Control Block

The timing control block is the point of entry for the two pixel clock signals (ICLK1, ICLK2), the horizontal synchronization signal (HS*), and the vertical synchronization signal (VS*).

When the pixel clock frequency is 20 MHz or lower, only ICLK1 is used. When the frequency is higher than 20 MHz, a two-phase clock input is created using both ICLK1 and ICLK2.

(7) Host Interface Block

The host interface block contains the interface for the host computer that controls the operation of the IP90MD25 module. Its data bus (SD00-07) is 8 bits wide, and its address bus (SA00-11) is 12 bits wide. The MWR* and MRD* signals control write and read operations, respectively. The M/R* signal controls memory/register access in accordance with SIDIP standards, although the IP90MD25 module contains no memory units.

(8) Other

a) ID Number Reading

The GST* signal outputs the IP90MD25's ID number (0Ch) through the data bus.

b) System Reset

The SRST* signal causes a hardware reset, and must be executed every hardware startup.

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD25. Signal pins are divided between the image system (Table 1A) and control system (Table 1B), having three and two external connectors, respectively. These pin assignments conform to SIDIP standards.

MECHANICAL DIMENSIONS

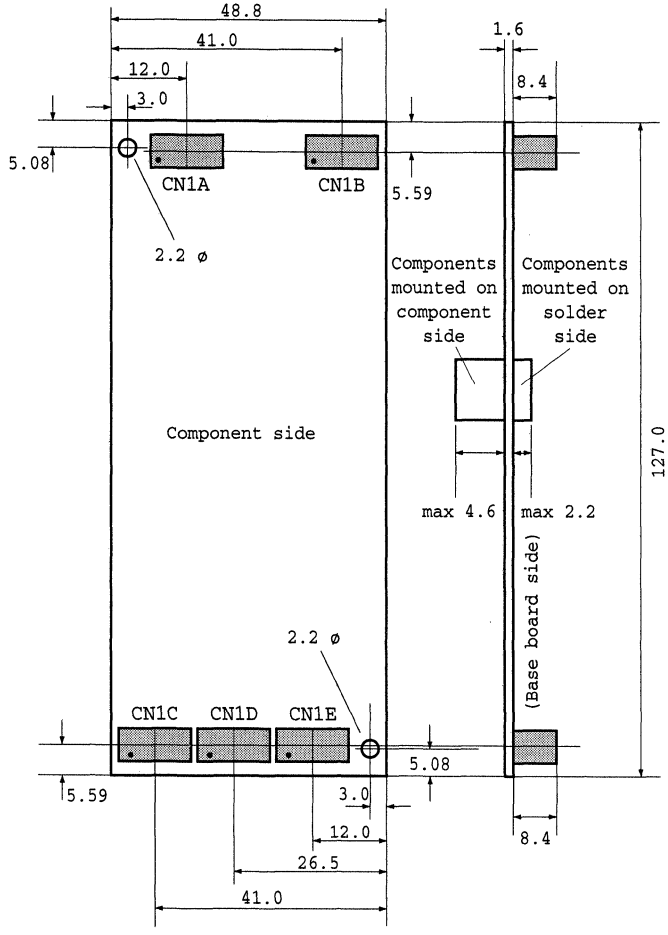
Figure 4 shows the board size and signal pin connector locations. The size and locations conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
OUT08*	CN1D	2	Output image data bus (Bit 0)	O
OUT09*	CN1D	17	Output image data bus (Bit 1)	O
OUT10*	CN1D	3	Output image data bus (Bit 2)	O
OUT11*	CN1D	18	Output image data bus (Bit 3)	O
OUT12*	CN1D	4	Output image data bus (Bit 4)	O
OUT13*	CN1D	19	Output image data bus (Bit 5)	O
OUT14*	CN1D	5	Output image data bus (Bit 6)	O
OUT15*	CN1D	20	Output image data bus (Bit 7)	O
IOEL*	CN1E	2	Image bus output enable (lower)	I

Table 1A. Image System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
SA08	CN1B	21	Control address bus (Bit 8)	I
SA09	CN1B	7	Control address bus (Bit 9)	I
SA10	CN1B	22	Control address bus (Bit 10)	I
SA11	CN1B	8	Control address bus (Bit 11)	I
MEN*	CN1A	14	Module enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I

Table 1B. Control System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.



Note 1: Units: mm

Figure 4. Board Size and Connector Positions

Notes

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PB-VF84-B (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B)

2: Connector pin number assignments:

•: #1 pin mark

3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (single-wide).

SIDIP Image Processing Modules

IP90MD81/01/05

Histogram / Projection Module

Product Manual

Ver. E 1.2

Sumitomo Metal Industries, Ltd.

SIDIP Histogram / Projection Module
IP90MD81/01/05

FEATURES AND FUNCTIONS

- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- Module size 48.8 x 127.0 mm (SIDIP single-wide size)
- Operates on single 5V power source
- All terminal signal levels are TTL level
- Compatible with 8- and 16-bit CPU busses
- Module ID can be read out

Histogram Unit

- Histogram processing of 8-bit grayscale image data
- Maximum histogram processing area of 1023 x 1023 pixels
- Vertical and horizontal processing area dimensions can be programmed independently
- One-shot clearing of histogram data
- Pixel clock frequency range of 5 to 25 MHz

Projection Unit

- Projection processing of 8-bit grayscale image data
- Simultaneous projection processing of vertical and horizontal dimensions
- Vertical and horizontal processing area dimensions can be programmed independently
- Maximum projection processing area of 1024 x 1024 pixels
- Pixel clock frequency range of 5 to 25 MHz

PRODUCT LINEUP

The IP90MD81 module can perform both histogram and projection processing. It is also available in versions for histogram processing only (IP90MD01) and for projection processing only (IP90MD05).

The IP90MD81/01/05 series is available in different models based on the size of the maximum processing area, as shown in the table below. Unless otherwise specified, this manual describes model IP90MD81-1K. Other models are available only in mass-produced versions.

	Maximum processing area	
	Histogram processing	Projection processing
IP90MD81-1K	1023 x 1023	1024 x 1024
IP90MD81-512	1023 x 1023	512 x 512
IP90MD01	1023 x 1023	—
IP90MD05-1K	—	1024 x 1024
IP90MD05-512	—	512 x 512

APPLICATIONS

- Character-recognition or pattern-recognition devices
- General purpose image-processing devices
- Built-in image-processing devices for factory automation (FA) equipment
- Built-in image-processing devices for office automation (OA) equipment

PRODUCT OVERVIEW

The IP90MD81 histogram/projection processing module uses a core containing two chips from the Sumitomo Metal Industries image-processing LSI (large scale integration) series: the IP90C01 histogram-processing LSI, and the IP90C05A projection-processing LSI. The core also includes peripheral circuitry in a proprietary circuit board designed for histogram and projection processing.

The module's bus architecture is compatible with the SIDIP standard interface for digital image processing proposed by Sumitomo Metal Industries, and includes bus interfaces for other image-processing modules.

Histogram Processing

The IP90MD81 performs histogram processing in the same way as the IP90C01 chip.

Histogram processing requires input of the pixels of the 8-bit grayscale raster image to be processed, together with the pixel clock and the horizontal and vertical synchronization signals synchronized with those pixels. All grayscale values are expressed in a 20-bit histogram. Grayscale histogram processing is done on the image bus in real time by the IMBC (image data bus controller IP90C51), which is located in the image input processing unit along with the IP90C01 chip.

The maximum area that can be processed is 1023 horizontal pixels by 1023 vertical lines. The pixel clock operates at a maximum frequency of 25MHz.

Projection Processing

The IP90MD81 board includes four IP90C05A chips, and performs projection processing in the same way as the IP90C05A chip.

Projection processing can be either Y-axis processing (of the sums of the grayscale values of each row), or X-axis processing (of the sums of the grayscale values of each column).

Projection processing requires input of the pixels of the 8-bit grayscale raster image to be processed, along with the pixel clock and the horizontal and vertical synchronization signals synchronized with those pixels.

The IP90MD81 module executes X-axis and Y-axis projection processing on the image bus simultaneously and in real time by the IMBC (image data bus controller IP90C51), located in the image input processing unit with the IP90C05A chips.

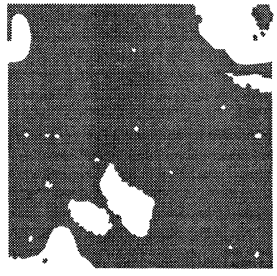


Figure 1(a). Original Image

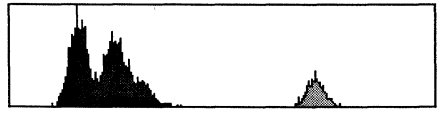


Figure 1(b). Example of a Histogram of Distribution of Grayscale Values

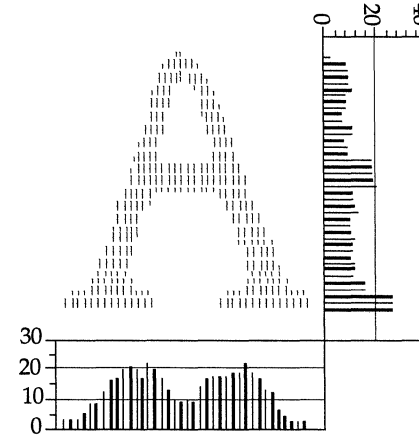


Figure 2. Example of Character Recognition by Projection Processing

The maximum area that can be processed is 1024 horizontal pixels by 1024 vertical lines. The pixel clock operates at a maximum frequency of 25 MHz.

The IP90MD81 module has its own module ID, which can be read by the host computer for easier configuration of systems. This ID can be used to determine whether the IP90MD81 is present, or for automatic system configuration.

The IP90MD81 module replaces previous low-speed software processing systems and bulky hardware, providing a space-saving, high-speed hardware device for easy histogram and projection processing. The module is designed for use in general-purpose image-processing devices, and particularly for built-in image processing devices in FA applications that require high processing speeds. It is also ideally suited for built-in systems in OA equipment such as copiers and fax machines.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD81 module can handle either binary-coded or 8-bit grayscale raster images (non-interleaved). The minimum system required for the IP90MD81 to perform histogram and projection processing includes a pixel clock synchronized with the pixels that make up the raster image, as well as the horizontal and vertical synchronization signals necessary to construct the raster image (see Figure 3). These signals are synchronized with the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

The IP90MD81 module can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded area in Figure 3. The size and position of this area can be controlled from the host computer. In addition, areas outside this designated area can be considered background and processed in gradients (0-255) controlled from the host computer.

Figure 4 shows an example of an actual application.

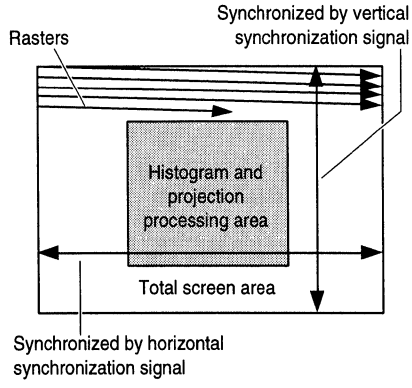


Figure 3. Definition of Image Area for Processing

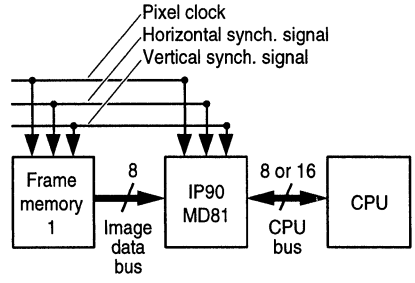
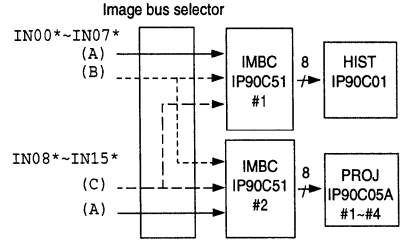


Figure 4. Sample Application



(A): IN00*-IN07* selection occurs in IMBC#1, and IN08*-IN15* selection occurs in IMBC#2.
(B): IN00*-IN07* selection occurs in both IMBC#1 and IMBC#2.
(C): IN08*-IN15* selection occurs in both IMBC#1 and IMBC#2.

Figure 5. Image Bus Selection

(2) Image Input Block

Figure 6 shows a block diagram of the IP90MD81 module. Figure 5 shows the flow of image input from the bus selector in Figure 6.

Inverted image data is input at the input image bus (located at IN00-IN15). The input block inverts this data again, and also converts 8-bit grayscale data to binary data if binary processing is used. Binary data is converted to 8-bit logic using the values 00h for [0] and FFh for [1]. Note that because the bus carries values that are inverted for processing, data entering the module appears in the bus as FFh for [0] and 00h for [1].

(3) Histogram Processing Block

This block provides the necessary functions for histogram processing, using the IP90C01 histogram-processing LSI chip.

(4) Projection Processing Block

This block provides the necessary functions for projection processing, using four IP90C05A histogram-processing LSI chips. The maximum processing area is 1024 x 1024 pixels.

(5) Timing Control Block

The timing control block is the point of entry for the pixel clocks (ICLK1, ICLK2), the horizontal synchronization signal (HS*), and the vertical synchronization signal (VS*).

The maximum pixel clock frequency handled by the IP90MD81 module is 25 MHz. At frequencies higher than 20 MHz, a two-phase clock input is created using both ICLK1 and ICLK2. At frequencies of 20 MHz or lower, only ICLK1 is used.

(6) Host Interface Block

The host interface block contains the interface for the host computer that controls the operation of the IP90MD81 module. The data bus (SD00-15) and the address bus (SA00-15) are 16 bits wide. An 8-bit data bus interface can also be used by using only the first 8 bits of the data bus (SD00-07).

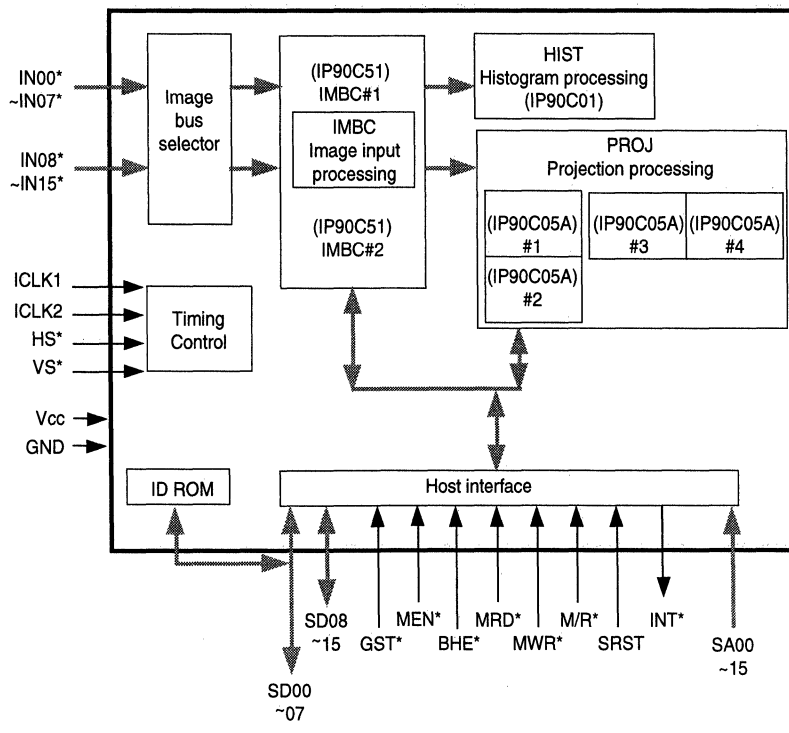


Figure 6. Block Diagram

Write operations are controlled by the MEN* and MWR* signals, and read operations by the MEN* and MRD* signals. Register identification is controlled by the M/R* signal.

a) ID Number Reading

The GST* signal outputs the IP90MD81's ID number (1Ch) through the data bus.

b) System Reset

The SRST* signal causes a hardware reset, which resets all module registers to 00h.

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD81. Signal pins are divided between the image system (Table 1A) and control system (Table 1B), having three and two external connectors, respectively. These pin assignments conform to SIDIP standards.

MECHANICAL DIMENSIONS

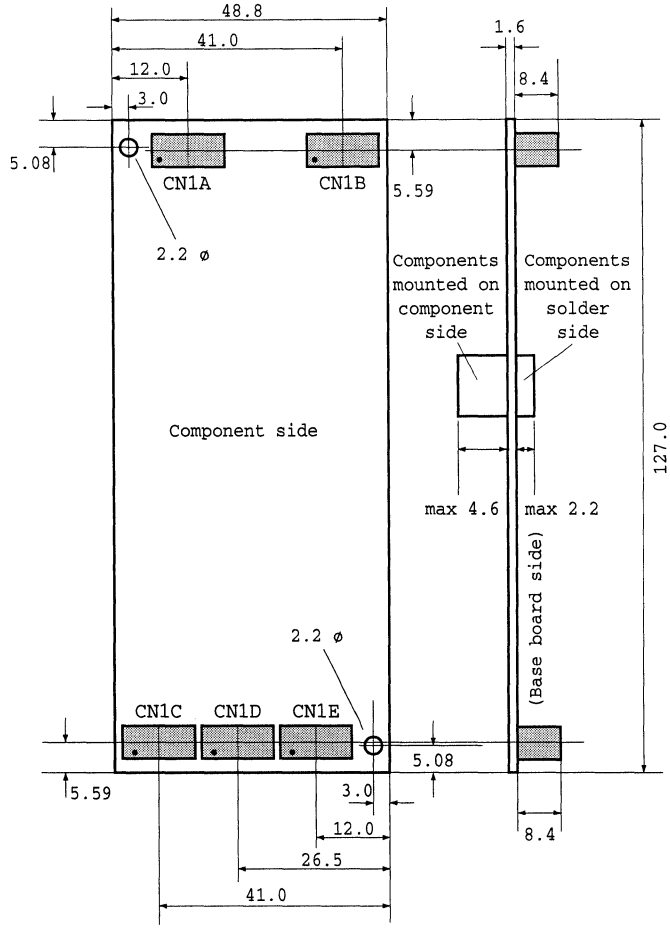
Figure 7 shows the board size and signal pin connector locations. The size and locations conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
IN08*	CN1C	7	Input image data bus (Bit 8)	I
IN09*	CN1C	22	Input image data bus (Bit 9)	I
IN10*	CN1C	8	Input image data bus (Bit 10)	I
IN11*	CN1C	23	Input image data bus (Bit 11)	I
IN12*	CN1C	9	Input image data bus (Bit 12)	I
IN13*	CN1C	24	Input image data bus (Bit 13)	I
IN14*	CN1C	10	Input image data bus (Bit 14)	I
IN15*	CN1C	25	Input image data bus (Bit 15)	I

Table 1A. Image System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SD08	CNIA	7	Control data bus (Bit 8)	I/O
SD09	CN1A	22	Control data bus (Bit 9)	I/O
SD10	CNIA	8	Control data bus (Bit 10)	I/O
SD11	CN1A	23	Control data bus (Bit 11)	I/O
SD12	CNIA	9	Control data bus (Bit 12)	I/O
SD13	CN1A	24	Control data bus (Bit 13)	I/O
SD14	CN1A	10	Control data bus (Bit 14)	I/O
SD15	CNIA	25	Control data bus (Bit 15)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
SA08	CN1B	21	Control address bus (Bit 8)	I
SA09	CN1B	7	Control address bus (Bit 9)	I
SA10	CN1B	22	Control address bus (Bit 10)	I
SA11	CN1B	8	Control address bus (Bit 11)	I
SA12	CN1B	23	Control address bus (Bit 12)	I
SA13	CN1B	9	Control address bus (Bit 13)	I
SA14	CN1B	24	Control address bus (Bit 14)	I
SA15	CN1B	10	Control address bus (Bit 15)	I
BHE*	CN1B	2	Control bus upper byte enable	I
INT*	CN1A	27	Interrupt request signal	O
MEN*	CN1A	14	Module enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I

Table 1B. Control System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.



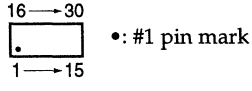
Note 1: Units: mm

Figure 7. Board Size and Connector Positions

Notes

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PB-VF84-B (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B)

2: Connector pin number assignments:



3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (single-wide).

5: Connector position measurements are to the center of the connector.

SIDIP Image Processing Modules

IP90MD100

Frame Memory Module

Product Manual

Ver. E 1.1

Sumitomo Metal Industries, Ltd.

SIDIP Frame Memory Module
IP90MD100

FEATURES AND FUNCTIONS

- Compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- Maximum pixel clock frequency of 25 MHz
- Maximum input image area of 4095 x 4095 pixels
- Frame memory size of 1024 x 1024 x 8-bit x 2 frames, or 2048 (horizontal) x 512 (vertical) x 8-bit x 2 frames
- Dual image bus systems for both input and output (four systems in all)
- Frame memory space coordinates can be set at any location within the input/output image space
- Independent read/write window scanning in frame memory space
- Supports read and write freeze functions with programmable freeze timing
- Frame memory space can be accessed from host computer
- 16-bit memory access to two memory frames
- Compatible with 8-bit and 16-bit ISA busses
- Module ID can be read out
- Operates on a single 5-V power source
- All terminal signal levels are TTL level
- Module size of 48.8 x 127.0 mm (SIDIP single-width size)

APPLICATIONS

- Expansion frame memory for image-processing systems configured with Sumitomo Metal Industries base board/module components
- Large-scale frame memory applications
- Built-in frame memory for general image-processing devices

PRODUCT DESCRIPTION

The IP90MD100 frame memory module is based on the SIDIP standard interface for digital image processing proposed by Sumitomo Metal Industries. The IP90MD100 provides memory frame capacity for two frames of 1024 (horizontal) x 1024 (vertical) x 8-bit images, or two frames of 2048 (horizontal) x 512 (vertical) x 8-bit images. The module supports window scanning for high-speed transfer and processing, using separate read and write windows whose coordinates can be defined independently within each frame memory space. Each frame memory space also supports definition of input/output image space using programmable coordinates.

The IP90MD100 module has two independent SIDIP image bus systems for input and two for output. This allows simultaneous transfer of two streams of image data to and from both frame memory spaces through the image bus systems.

The IP90MD100 supports independent freeze functions for reading and writing. It provides access to frame memory from a host computer for sequential access read and write functions.

STRUCTURE AND OPERATION

(1) Input/Output Images

The IP90MD100 module can handle binary-coded or 8-bit grayscale raster images (non-interleaved). The IP90MD100 requires as a minimum a system that includes a pixel clock synchronized with the pixels that make up the raster image, as well as the horizontal and vertical synchronization signals necessary to construct the raster image. These signals are synchronized with the pixel clock, and must have the width of one cycle of the inverse logic pulse as changed by the rise of the pixel clock.

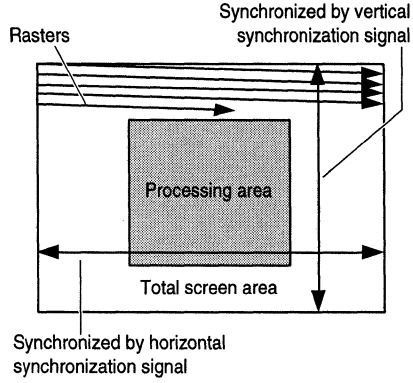


Figure 1. Definition of Image Area for Processing

The IP90MD100 module can process only the area defined by the vertical and horizontal synchronization signals, represented by the shaded area in Figure 1. The host computer controls the size and position of this area. In addition, areas outside the designated processing area can be considered background and processed as any gradient value (0-255) designated by the host computer.

The output image is a labeled raster image, as is the input image, and has the same pixel clock and synchronization signals as the input image. However, the module uses an internal delay factor to delay the output image by a specified number of pixels horizontally and vertically relative to the input image.

Figure 2 shows an example of an actual application.

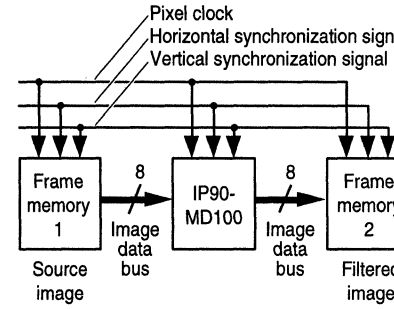


Figure 2. Sample Application

(2) Image Input Block

Figure 3 shows a block diagram of the IP90MD100 module.

Inverted image data enters through two input image bus systems: lower bytes enter at the eight-bit IN0*-IN07* bus system, and upper bytes enter at the eight-bit IN08*-IN15* bus system. The lower bytes are allocated to frame memory FM#0, and the upper bytes to FM#1.

This input data is logically inverted again in the image input block.

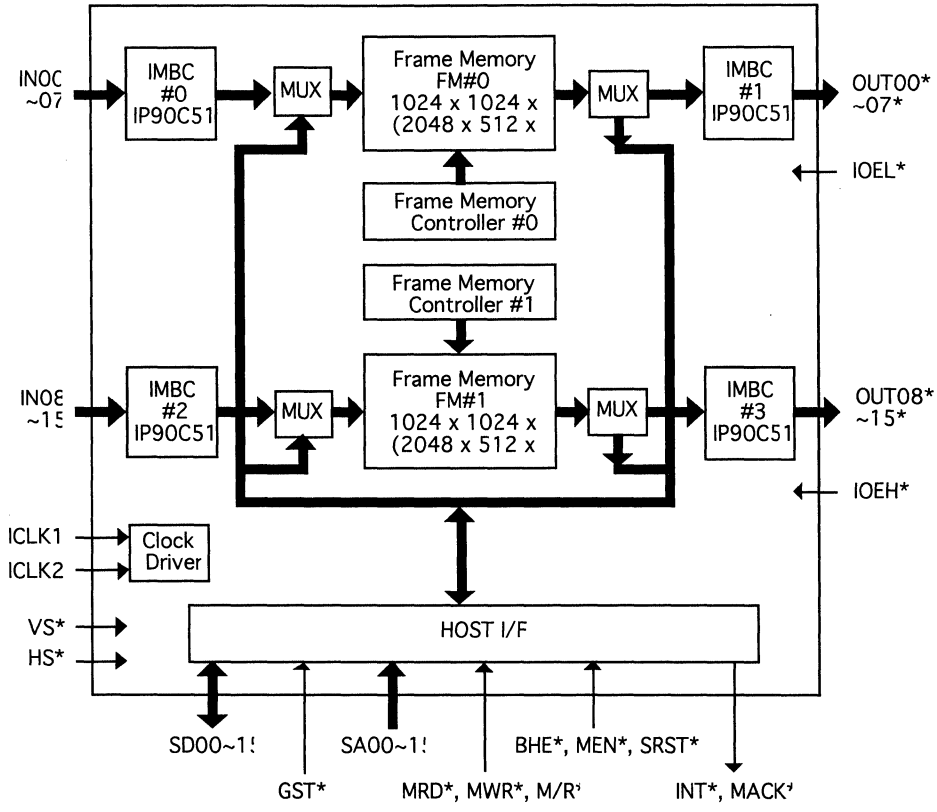


Figure 3. Block Diagram

(3) Frame Memory Block

Frame memory is constructed from field memory units, and can be used as two 1024 (horizontal) x 1024 (vertical) x 8-bit frames, or as two 2048 (horizontal) x 512 (vertical) x 8-bit frames. Frame memory function is described in "Frame Memory Function," below.

(4) Image Output Block

Data can be outputted over two output image bus systems: lower bytes at the eight-bit OUT0*-OUT07* bus system, and upper bytes at the eight-bit OUT08*-OUT15* bus system. The lower bytes are allocated to frame memory FM#0, and the upper bytes to FM#1.

The module produces three-state output, with inverted signal logic and independently controlled connection status.

Frame Memory Function

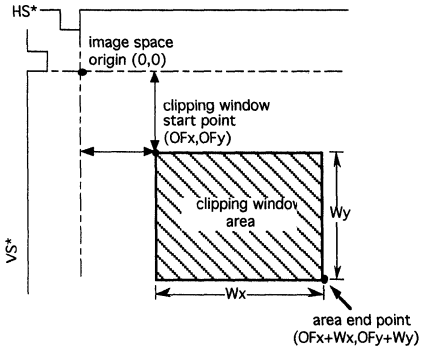
(1) Frame Memory Input/Output Space

The IP90MD100 module handles image data in terms of the following three types of space: input/output image space, frame memory space, and clipping window areas. These are described below.

(1-1) Input/Output Image Space

The IP90MD100 handles the vertical and horizontal synchronization signals VS* and HS* in terms of a rectangular image area called the clipping window area. This area has a maximum size of 4095 x 4095 pixels, and is mapped at a designated address within frame memory along the time axis, using any given coordinates. This area is used to input or output images.

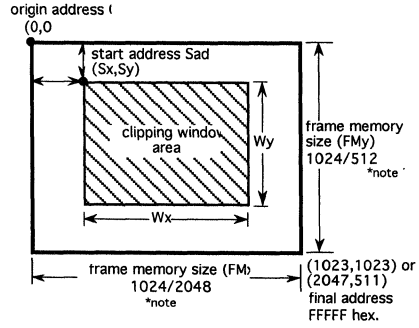
Figure 4 shows a two-dimensional image space representing a raster scan based on the synchronization signal system, with two sample clipping window areas mapped in frame memory.



[Two-dimensional image space coordinates along time

OFx: Horizontal start point of clipping window (horizontal off
OFy: Vertical start point of clipping window (vertical offset)
Wx: Clipping window width
Wy: Clipping window height

Figure 4. Schematic Diagram of Two-Dimensional Input/Output Space (Along Time Axis)



[Two-dimensional coordinates and size of frame memory address s

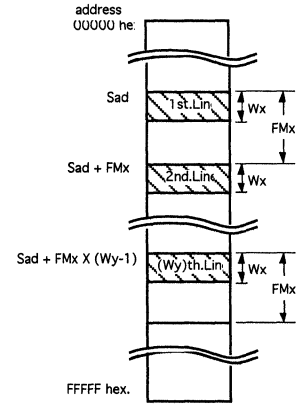
Sx: Horizontal start point of clipping window
Sy: Vertical start point of clipping window
Wx: Clipping window width
Wy: Clipping window height
FMx: Frame memory width
FMy: Frame memory height

Note 1 In 1024 x 1024 pixel mode, frame memory configuration is horizontal FMx = 1024, vertic FMy = 1024. In 2048 x 512 pixel mode, fram memory configuration is horizontal FMx = 20· vertical FMy = 512.

Figure 5. Schematic Diagram of Frame Memory Address in Two Dimensions

(1-2) Frame Memory Space

The IP90MD100's frame memory has capacity for two 1024 x 1024 x 8-bit frames, or two 2048 (horizontal) x 512 (vertical) x 8-bit frames. As Figure 5 shows, each clipping window area is defined within frame memory space by mapping selected coordinates in the input/output image space. Figure 6 shows a one-dimensional representation of frame memory address space along the memory address axis.



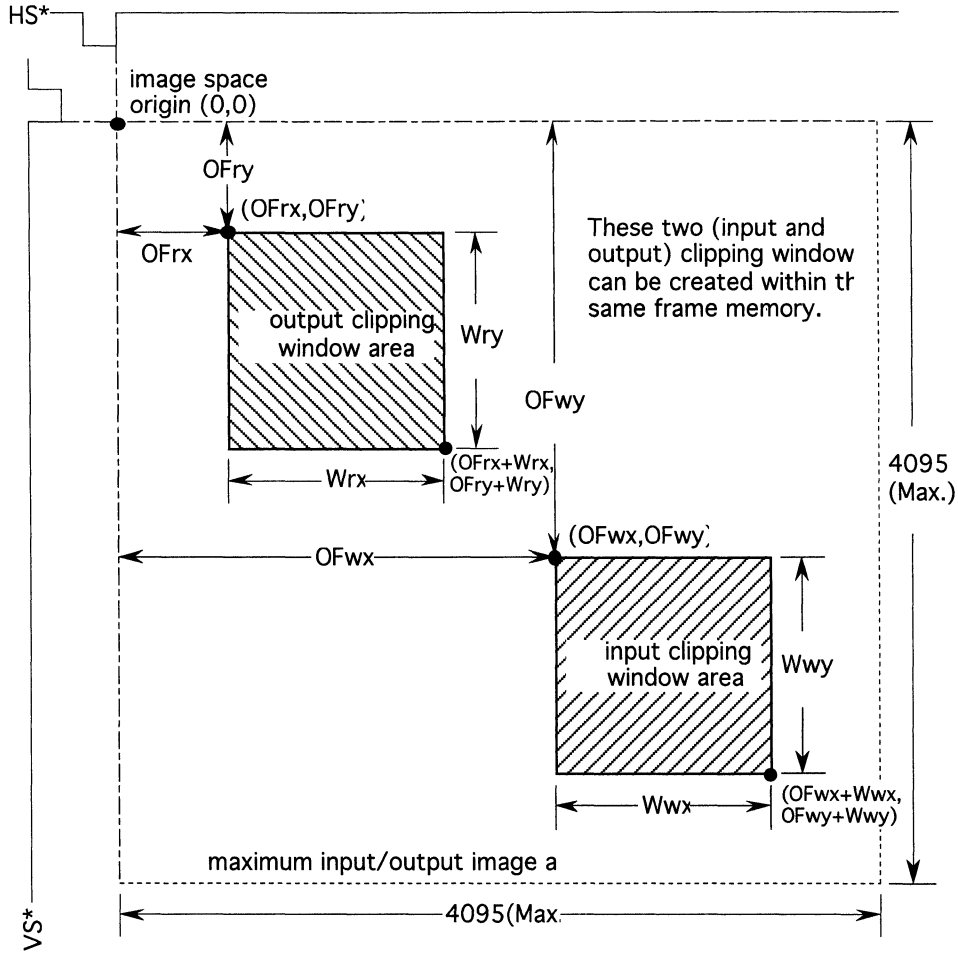
[One-dimensional addressing in frame memory address space]

Sad: Starting address for clipping window
Wx: Clipping window width
Wy: Clipping window height
FMx: Frame memory width
FMy: Frame memory height

Figure 6. Schematic Diagram of Frame Memory Address in One Dimension

(1-3) Clipping Window Areas

A clipping window is a local area of frame memory space that is defined as an input area or output area. Two areas can be designated within a single frame memory area, one as a window for output (reads from frame memory) and the other as a window for input (writes to frame memory). Figure 7 shows output and input clipping areas mapped in two-dimensional image space.



[Two-dimensional image space coordinates along time

OFrx: Horizontal start point of output clipping window (horizontal offset)
OFry: Vertical start point of output clipping window (vertical offset)
OFwx: Horizontal start point of input clipping window (horizontal offset)
OFwy: Vertical start point of input clipping window (vertical offset)
Wrx: Output clipping window width
Wry: Output clipping window height
Wwx Input clipping window width
Wwy Input clipping window height

Figure 7. Example of Mapping Input and Output Clipping Windows in Two-Dimensional Space Along the Time Axis

(2) Input/Output Data Phase Relationship

The image data written into frame memory is sent to the output image bus after a delay of 0 or 1 frames.

If the input and output clipping windows overlap in frame memory address space, the delay is then the period before the input image data is sent back as output. As long as the clipping windows do not overlap, however, the input image data is not immediately sent back as output, so no delay factor is used.

The delay factor between image data input and output differs according to the image space coordinates of the clipping window areas. Figure 8 shows three examples of phase relationships between windows, and describes the corresponding differences in delay in relation to their relative positions.

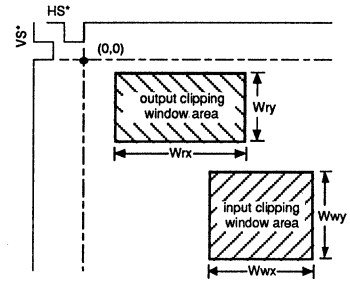
(2-1) Output Clipping Window Located at an Earlier Position in Image Space (Figure 8, Case A)

The contents of the output clipping window are sent before image data is written into the input clipping window. Thus, output is always delayed by one frame, even though the user has accessed the same address in frame memory. This means incoming data is not output until the next frame. For example, image data input in the n^{th} frame is output as the $(n+1)^{th}$ frame.

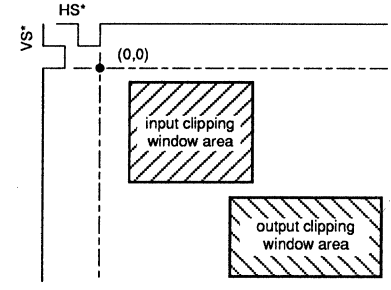
(2-2) Input Clipping Window Located at an Earlier Position in Image Space (Figure 8, Case B)

The image data is written into the input clipping window before data is read out of the output clipping window. This means the freshest incoming image data is always output within the same frame (delay of 0 frames) when the user has accessed the same address in frame memory.

(A) Output clipping window placed at an earlier position in image space than the input clipping window



(B) Input clipping window placed at an earlier position in image address space than the output clipping window



(C) Input and output clipping windows overlapping in image field space.

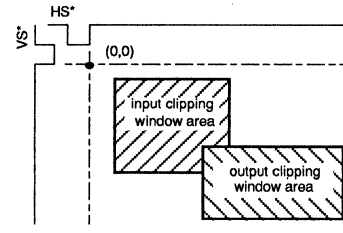


Figure 8. Three Relationships of Input and Output Clipping Windows Mapped in Two-Dimensional Input/Output Image Space and Frame Memory Address Space Along the Time Axis

(2-3) Input and Output Clipping Windows Overlap (Figure 8, Case C)

As long as the pixel data is written into frame memory and read out in 160 clock counts or less, the data read out will be one frame earlier than the latest input data. If more than 160 clock counts elapse, the data written out will be the latest input data. Therefore, if any pixels in the output clipping window occupy the same addresses as pixels in the input clipping window, and are 160 pixels or more behind the input image on the raster axis, the frame delay is 0 frames. If the difference is less than 160 pixels, the delay is 1 frame.

(3) Freeze Functions

Each module has independent freeze functions for both memory frames. These functions can freeze images during reading from frame memory to the image bus, or during writing from the image bus to frame memory. Freeze functions are controlled from the host computer.

(3-1) Write Freeze

The timing of a write freeze is determined by selecting and entering a VS* signal count setting (this setting must be in the range from 0 to 255 counts). When the host computer enables the write freeze function, the module counts the designated number of VS* signals, and freezes the image frame that is synchronized with the last VS* signal count. Applying a write freeze stops the writing of image data to frame memory from the input image bus, so frame memory contents no longer change. To notify the host computer that the write freeze is complete, the module sets the write freeze flag in the status register. Also, if a write freeze interrupt from the host computer is permitted, the module generates an interrupt request for the host computer.

When the write freeze function is then disabled from the host computer, the write freeze is released beginning with the frame synchronized with the next VS* signal that follows the disable signal.

(3-2) Read Freeze

A read freeze transfers from the module to the host computer the authority to read the contents of frame memory to the host computer. To apply a read freeze, it is first necessary to apply a write freeze from the host computer.

When a read freeze is enabled from the host computer, reading from frame memory to the image bus is stopped with the frame synchronized with the next VS* signal that follows the enable signal. After this, frame memory output is connected to the host computer bus. To notify the host computer that the read freeze is complete (meaning that the host computer can now read from frame memory), the module sets the read freeze flag in the status register. Also, if a read freeze interrupt from the host computer is permitted, the module generates an interrupt request for the host computer. When the read freeze function is enabled, the image bus output is 'Hi-z.'

When the host computer disables the read freeze function, the freeze is released beginning with the frame synchronized with the next VS* signal that follows the disable signal. The write freeze function can only be released after the read freeze function is released.

(4) Window Scan Functions

By designating certain areas of frame memory space as clipping windows for image bus input and output, the IP90MD100 module makes it possible for the image bus to read to or write from only the desired portions of frame memory space. Settings for clipping window areas are subject to the following restrictions:

- Minimum window area: 160 pixels (horizontal) x 4 lines (vertical)
- Horizontal window size increment/decrement units: 32 pixels
- Vertical window size increment/decrement units: 4 lines
- Read/write scans of two separate frames can be made by separate window scans

(5) Interface Functions Using the Control Bus

(5-1) Frame Memory Access

The host computer reads to and writes from frame memory using sequential access. Even-numbered host addresses (lower bytes) are allocated for FM#0, and odd-numbered host addresses (upper bytes) are allocated for FM#1. Thus, by using 16-bit access from the host computer through a 16-bit data bus interface, frame memory can be treated as having a data width of 16 bits, and two frames in image memory (FM#0 and FM#1) can be accessed at the same time.

When image data is written from the host computer into frame memory, a freeze is placed on writing from the image bus (write freeze), and data is sequentially written to the designated addresses in frame memory. This prevents the image bus and the host computer from simultaneously writing data to frame memory. The window scan function can be used to write data into limited local areas of frame memory.

When the host computer reads the image data in frame memory, a freeze is placed on reading from the image bus (read freeze), and the image data is read sequentially from the designated addresses in frame memory. This prevents the image bus and the host computer from simultaneously reading data from frame memory. The window scan function can be used to read data from limited local areas of frame memory.

(5-2) Interrupts

Interrupts are generated by the situations described below, and are output as the interrupt request signal INT*.

An enable/disable setting controls interrupts to the host computer. When interrupts are disabled, no interrupt signals are sent to the host computer.

- Write freeze completed interrupt: When the host computer enables a write freeze, an interrupt is generated simultaneously when the last VS* signal is detected, as determined by the prior VS* signal setting.
- Read freeze completed interrupt: When the host computer enables a read freeze, an interrupt is generated simultaneously when the last VS* signal is detected.

Refer to the interrupt status register to determine which of these conditions generated an interrupt signal.

When the interrupt status register is read, the interrupt condition is cleared.

(5-3) Module ID

When the GST* signal is active, the module ID (40h) is output to the control system data bus at SD0-SD7.

PIN ASSIGNMENTS

Table 1 lists the pin assignments for the IP90MD100 module. Signal pins are divided between the image system (Table 1A) and control system (Table 1B), having three and two external connectors, respectively. These pin assignments conform to SIDIP standards.

Signal	Connector	Pin No.	Function	I/O
ICLK1	CN1D	12	Image clock 1	I
ICLK2	CN1D	14	Image clock 2	I
HS*	CN1D	8	Horizontal synchronization signal	I
VS*	CN1D	10	Vertical synchronization signal	I
IN00*	CN1C	2	Input image data bus (Bit 0)	I
IN01*	CN1C	17	Input image data bus (Bit 1)	I
IN02*	CN1C	3	Input image data bus (Bit 2)	I
IN03*	CN1C	18	Input image data bus (Bit 3)	I
IN04*	CN1C	4	Input image data bus (Bit 4)	I
IN05*	CN1C	19	Input image data bus (Bit 5)	I
IN06*	CN1C	5	Input image data bus (Bit 6)	I
IN07*	CN1C	20	Input image data bus (Bit 7)	I
OUT00*	CN1C	12	Output image data bus (Bit 0)	O
OUT01*	CN1C	27	Output image data bus (Bit 1)	O
OUT02*	CN1C	13	Output image data bus (Bit 2)	O
OUT03*	CN1C	28	Output image data bus (Bit 3)	O
OUT04*	CN1C	14	Output image data bus (Bit 4)	O
OUT05*	CN1C	29	Output image data bus (Bit 5)	O
OUT06*	CN1C	15	Output image data bus (Bit 6)	O
OUT07*	CN1C	30	Output image data bus (Bit 7)	O
OUT08*	CN1D	2	Output image data bus (Bit 8)	O
OUT09*	CN1D	17	Output image data bus (Bit 9)	O
OUT10*	CN1D	3	Output image data bus (Bit 10)	O
OUT11*	CN1D	18	Output image data bus (Bit 11)	O
OUT12*	CN1D	4	Output image data bus (Bit 12)	O
OUT13*	CN1D	19	Output image data bus (Bit 13)	O
OUT14*	CN1D	5	Output image data bus (Bit 14)	O
OUT15*	CN1D	20	Output image data bus (Bit 15)	O
IOEL*	CN1E	2	Image bus output enable (lower)	I
IOEH*	CN1E	17	Image bus output enable (upper)	I

Table 1A. Image System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

Signal	Connector	Pin No.	Function	I/O
SD00	CN1A	3	Control data bus (Bit 0)	I/O
SD01	CN1A	18	Control data bus (Bit 1)	I/O
SD02	CN1A	4	Control data bus (Bit 2)	I/O
SD03	CN1A	19	Control data bus (Bit 3)	I/O
SD04	CN1A	5	Control data bus (Bit 4)	I/O
SD05	CN1A	20	Control data bus (Bit 5)	I/O
SD06	CN1A	6	Control data bus (Bit 6)	I/O
SD07	CN1A	21	Control data bus (Bit 7)	I/O
SD08	CN1A	7	Control data bus (Bit 8)	I/O
SD09	CN1A	22	Control data bus (Bit 9)	I/O
SD10	CN1A	8	Control data bus (Bit 10)	I/O
SD11	CN1A	23	Control data bus (Bit 11)	I/O
SD12	CN1A	9	Control data bus (Bit 12)	I/O
SD13	CN1A	24	Control data bus (Bit 13)	I/O
SD14	CN1A	10	Control data bus (Bit 14)	I/O
SD15	CN1A	25	Control data bus (Bit 15)	I/O
SA00	CN1B	17	Control address bus (Bit 0)	I
SA01	CN1B	3	Control address bus (Bit 1)	I
SA02	CN1B	18	Control address bus (Bit 2)	I
SA03	CN1B	4	Control address bus (Bit 3)	I
SA04	CN1B	19	Control address bus (Bit 4)	I
SA05	CN1B	5	Control address bus (Bit 5)	I
SA06	CN1B	20	Control address bus (Bit 6)	I
SA07	CN1B	6	Control address bus (Bit 7)	I
SA08	CN1B	21	Control address bus (Bit 8)	I
SA09	CN1B	7	Control address bus (Bit 9)	I
SA10	CN1B	22	Control address bus (Bit 10)	I
SA11	CN1B	8	Control address bus (Bit 11)	I
SA12	CN1B	23	Control address bus (Bit 12)	I
SA13	CN1B	9	Control address bus (Bit 13)	I
SA14	CN1B	24	Control address bus (Bit 14)	I
SA15	CN1B	10	Control address bus (Bit 15)	I
MEN*	CN1A	14	Module enable	I
BHE*	CN1B	2	Control bus upper byte enable	I
M/R*	CN1B	25	Memory/register	I
MRD*	CN1B	27	Read signal	I
MWR*	CN1B	12	Write signal	I
INT*	CN1A	27	Interrupt request signal	O
GST*	CN1A	29	ID read signal	I
SRST*	CN1B	28	System reset	I
MACK*	CN1B	13	Module acknowledge	O

Table 1B. Control System Signal Pin Assignments
Note: An asterisk (*) following a signal name indicates inverse logic.

MECHANICAL DIMENSIONS

Figure 9 shows the IP90MD100's board size and signal pin connector locations. The size and locations conform to SIDIP standards.

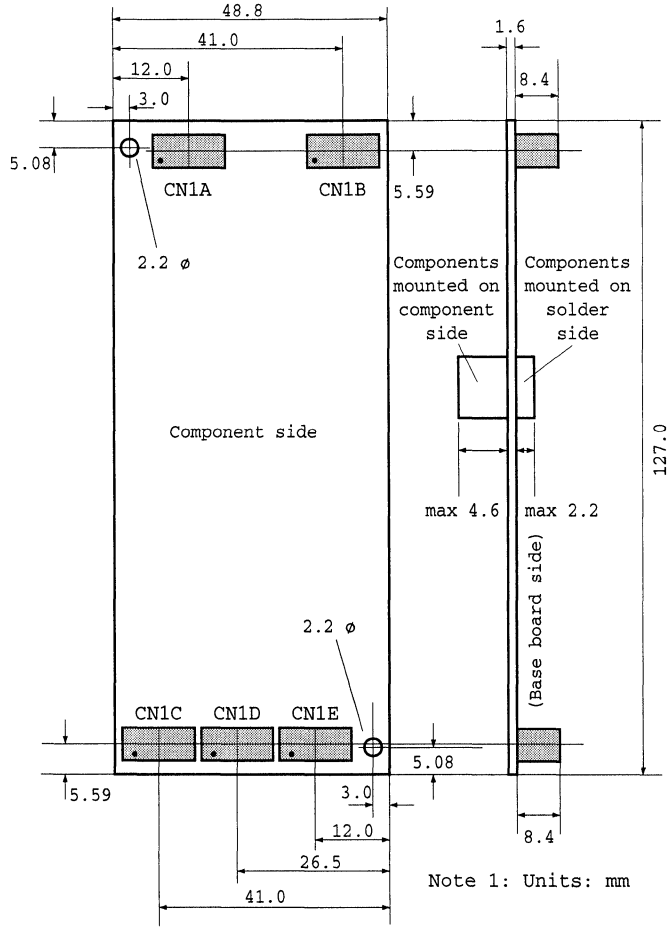


Figure 9. Board Size and Connector Positions

Notes

1: Connectors: Japan Aviation Electronics Industry, Limited IL-WX-30PE-VF84-B (Mating connector: Japan Aviation Electronics Industry, Limited IL-WX-30SB-VF-B)

2: Connector pin number assignments:

•: #1 pin mark

3: Connectors to be mounted on the soldered surface.

4: To conform to specifications for module external shape (single-wide).

5: Connector position measurements are to center of connector.

IP90BD301/351

Baseboard and Expansion Board for ISA Bus (PC/AT)

Product Manual

Ver. E 2.1

Sumitomo Metal Industries, Ltd.

Image-Processing Module Baseboard Series IP90BD301/351 (ISA Bus)

FEATURES AND FUNCTIONS

- Monochrome frame-grabber board designed for mounting image-processing modules compatible with the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries
- Compatible with the ISA bus interface used by PC/AT-compatible computers
- Accommodates 8-bit and 16-bit ISA bus widths
- Accommodates up to 3 single-size image-processing modules (or a double-size module and a single-size module)
- Four 1024 x 512 x 8-bit frame memory units (can be used with two 1024 x 512 x 16-bit screens)
- Two 1024 x 512 x 8-bit frame memories, one for image grabbing and one for display (640 x 480 effective pixels)
- Two Sumitomo IP90C55 image datastream controllers (IMSC), each of which includes a 16-bit ALU, barrel shifter, min/max comparator
- Eight 8-bit x 256 look-up table (LUT) banks
- Four EIA-compatible camera input channel connections
- VGA (640 x 480) or NTSC video signal output (selected by software)
- Can display 256 colors from a palette of 16.7 million colors
- Accepts an external trigger for freezing frame memory
- Can freeze by frame, or by a designated field
- The IP90BD351 allows image-processing module expansion
- Independent synchronization systems in the image input block, internal processing block, and image display block for high-speed processing (with an internal processing rate of 50 ns per pixel)
- Internal processing available for designated fields only

APPLICATIONS

- Built-in image-processing devices for Factory Automation (FA) equipment
- Algorithm evaluation
- Evaluation of image-processing modules and LSIs (large scale integrated circuits)
- General image-processing devices

PRODUCT DESCRIPTION

The IP90BD301/351 baseboard is designed to accommodate SIDIP-compatible image-processing modules. Each baseboard has the built-in video input/output interface and frame memories necessary to function as a monochrome frame grabber. The baseboard is also equipped with an ISA bus host computer interface, with a choice of 8-bit or 16-bit data bus widths. Board dimensions are SIDIP full-size equivalent.

The IP90BD301 baseboard has its video interface, frame memory, and ALU all on a single board, enabling the board to receive and display images for software image processing. For applications requiring higher processing speeds, the board can easily accommodate image-processing modules for filtering or other functions at speeds up to 50 ns per pixel. The IP90BD301 can hold up to three single-size image-processing modules, or a combination of a double-size and a single-size module.

The IP90BD351 expansion baseboard provides additional room for mounting extra image-processing modules.

The IP90BD301 baseboard contains two Sumitomo IP90C55 image datastream controller (IMSC) chips, allowing easy software control over the flow of image data. (For further information about IMSCs, see the technical documentation for the IP90C55 chip.) These chips contain ALUs capable of up to 16-bit calculations, as well as barrel shifters and min/max comparator circuits that can compare the brightness of two pixels and output the results, all at pixel-to-pixel processing speeds of 50 ns per pixel.

The IP90BD301 has four frames of 1024 x 512 x 8-bit frame memory. Of these, two frames can be used as 1024 x 512 x 16-bit frame memory. This is sufficient to store image data output within the time span of one frame, in cases when processing the maximum of 4094 labels (by the IP90MD10 labeling processor module) or ALU calculation results exceed 8 bits.

The IP90BD301's image input/output interface provides input connections for up to four EIA-compatible monochrome cameras. Output can be compatible with VGA (640 x 480) or NTSC, and can be displayed using 256 colors selected from a palette of 16.7 million colors.

The IP90BD301 baseboard contains eight banks of 256 x 8-bit LUTs. This allows the board to perform up to eight types of image remapping, all of which can be changed instantaneously through software settings.

The IP90BD301 can freeze a camera input image at a signal from a host computer or external freeze trigger. The user controls the conditions under which the freeze occurs, using (for example) the first vertical synchronization signal after the freeze request, or the vertical synchronization signal following the completion of a designated field or frame. An interrupt request can also be sent from the baseboard CPU when a freeze is completed.

After a freeze, a designated field can be extracted from the image for transfer over the internal image bus. This enables "field processing," which is substantially faster than full-frame processing.

The IP90BD351 is based on the IP90BD301, which allows more image-processing modules to be added to the system. The IP90BD351, however, has no video input/output interface, frame memory, or LUT, though it has the same number of module connections as the IP90BD301. Up to nine expansion baseboards can be added to a system.

STRUCTURE AND OPERATION

Figure 1 shows a block diagram of the IP90BD301 baseboard, and Figure 2 shows a block diagram of the IP90BD351 board.

(1) Input Processing Block

The input processing block performs camera switching, gain and offset adjustment, and A/D conversion.

Up to four cameras can be connected to the board. A software setting selects one of the cameras as the input signal. If multiple cameras are used, the system must be synchronized with the synchronization signal of each camera. All cameras connected to the system must meet EIA standards and use D-Sub 9-pin input connectors.

Software settings can adjust the gain and offset of the incoming image signal using 64 gradations. The gain adjustment range is ±6 dB, and the offset range is ±0.5 V.

A/D conversion uses 8-bit quantization, and a sampling frequency of 12.5 MHz.

The digitized image signal is sent to field memory, where it is converted from interlace to non-interlace, and read into the (SIDIP standard) internal image bus using a 20-MHz pixel clock. The image size for frame-grabbing is 640 x 480 pixels.

These conversions take place in the field memory blocks, which are also used to freeze the input image.

The freezing process begins with setting freeze conditions (described later). The system then freezes the input image when a freeze request is sent by the CPU or an external trigger and the freeze conditions are met.

Any of the following freeze conditions can be selected, depending on how the input image vertical synchronization signal is treated:

(1) at the first detection of a vertical synchronization signal after a freeze request

(2) at the first detection of a vertical synchronization signal between field 1 and field 2 after a freeze request

(3) at the first detection of a vertical synchronization signal between field 2 and field 1 after a freeze request

The baseboard can also take freeze-frame image data in field memory (frame data) and output only a designated field to the internal image bus. This enables the system to perform field processing.

Note that when the system is used with a VTR unit, the manufacturers cannot assure image stability and quality.

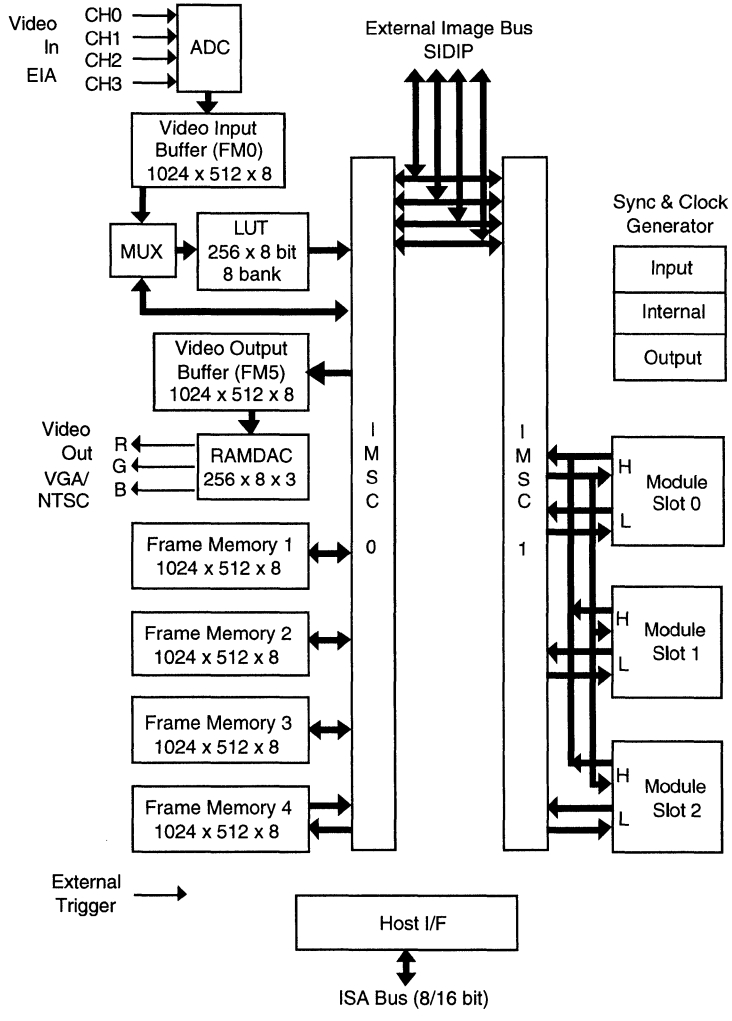


Figure 1. IP90BD301 Block Diagram

(2) Output Processing Block

The output processing block produces a VGA or NTSC image for display.

Digital image data sent over the internal image bus is written into a frame memory for display. The image size at this point is 640 x 480 pixels. If an NTSC image is required, the signal is converted from non-interlace to interlace, and the pixel clock is changed. Switching between VGA and NTSC output is by software setting.

The D/A converter uses three 8-bit processors, one for each component of the RGB signal. An internal LUT is used to produce a display in 256 colors (from a palette of 16.7 million colors). The output connector is a high-density D-Sub 15-pin connector.

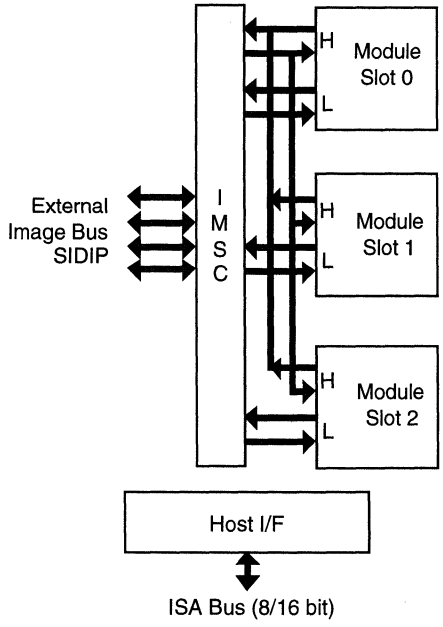


Figure 2. IP90BD351 Block Diagram

(3) Frame Memory Blocks

The IP90BD301 baseboard has a total of four 1024 x 512 x 8-bit frame memory units, each with an effective pixel count of 640 x 480. Of these, three (FM1 through FM3) use dual-port DRAM to allow random access from the CPU simultaneously with writing or reading exchanges on the image bus. FM4 uses a field memory and cannot be accessed from the CPU, but can simultaneously write data to or read data from the image bus. This allows the FM4 block to absorb the pixel-unit delays that are generated in the course of pipeline-style processing in various image-processing modules.

Each of the frame memory blocks, controlled by the CPU, can freeze an image. The timing is controlled by the detection of the first vertical synchronization signal (in the internal processing system) after the CPU freeze request.

The CPU has only byte access to FM1 and FM2, but has byte access and word access to FM3. FM1 and FM2 can be used together as a 16-bit-wide frame memory, with word access capability in which even addresses are allocated to FM1 and odd addresses to FM2. This is an efficient way to allow the CPU to handle images that exceed 8-bit width after labeling processing.

(4) LUT Block

This block provides image remapping using built-in 256 x 8-bit LUTs. Up to 8 banks of LUTs can be designated by the CPU, and their settings can be changed instantaneously. The input to a LUT can be provided by output from the internal image bus or video input processing block. The output is connected to the internal image bus.

(5) Image-Processing Module Interface

The IP90BD301 baseboard can be connected to as many as three single-size image-processing modules, or to a combination of a double-size module and a single-size module. (For details about image bus connections, see the description of the image bus control block.)

	Port	Connects to	I/O
I M S C # 0	A	External bus A	I/O
	B	External bus B	I/O
	C	External bus C	I/O
	D	External bus D	I/O
	E	Frame memory 0	I/O
	F	Frame memory 1	I/O
	G	Frame memory 2	I/O
	H	LUT	O
	I	LUT	I
	J	Frame memory 3	O
	K	Frame memory 3	I
	L	Output frame buffer	I/O
I M S C # 1	A	External bus A	I/O
	B	External bus B	I/O
	C	External bus C	I/O
	D	External bus D	I/O
	E	Module #0 lower byte	O
	F	Module #0 lower byte	I
	G	Module #1 lower byte	O
	H	Module #1 lower byte	I
	I	Module #2 lower byte	O
	J	Module #2 lower byte	I
	K	All modules upper byte	O
	L	All modules upper byte	I

Table 1. IMSC Connections

(6) Image Bus Control Block

This block controls the IP90BD301's image bus.

The IP90BD301 contains two Sumitomo Industries' IP90C55 image stream controller (IMSC) LSI chips, while the IP90BD351 contains a single IP90C55 IMSC chip. The image bus control block controls the internal image bus connected to each of the processing blocks described earlier. The block also controls the external bus connections when the IP90BD351 baseboard is present.

Logic calculation	AND OR NOT EXOR
Arithmetic calculation	+ (add) – (subtract) Carry calculation
Shift calculation	0-8 bit or 0-16 bit (right shift)
MIN/MAX	MIN (A, B) MAX (A, B) Compare/select

Table 2. IMSC Internal Calculation Functions

These LSI chips make it possible to use software settings to create any configuration of processing functions connected to any internal or external bus. In addition, each IMSC chip has a built-in 8-bit ALU and barrel shifter, as well as two sets of min/max calculation functions (with a combined 16-bit calculation capacity) to output either the greater or lesser gradient values from incoming image data streams. These features can be used in the same way as any of the processing functions connected to the image bus.

Table 1 shows the connections of each port on the two IMSC chips mounted on the IP90BD301. Table 2 shows the calculation functions built into the IMSC chip.

(7) Synchronization Signal Processing Block

The IP90BD301 provides optimum synchronization signals and pixel clock signals by using independent synchronization signal generators and image clock generators for image input, internal processing, and image display.

The pixel clock used for internal processing has a speed of 20 MHz, which reduces the time for processing an image frame from 33 ms to approximately 21 ms.

(8) Host Interface

The IP90BD301/351 baseboards use an ISA bus as a host computer interface.

Mapping is handled by bank switching when less than 1 Mb of CPU address space is used. However, when 1 Mb or more is used, the entire capacity of the IP90BD301/351 is allocated linearly.

The IP90BD301 has 2 Mb of address space, and the IP90BD351 has 512 Kb.

Bank switching requires a 16-Kb window in CPU address space. This window is designated as the 16 Kb beginning with address 000000h. (Bank switching also occupies 1 byte of I/O space. I/O addresses are read in 16-bit full decoding.)

When 1 Mb or more address space is allocated, the address space is defined as the 2 Mb beginning with address 100000h.

The ISA data bus can be used in either 8-bit or 16-bit width. Note, however, that ISA bus specifications require that 16-bit width have 128 Kb of continuous address space.

The interrupt request signal can be IRQ 10, 11, 12, or 15.

Note that PC/AT-compatible computers have characteristics that differ according to the manufacturer. Sumitomo Metal Industries continually monitors the operating performance of various computers using IP90BD301/351 baseboards, and users are advised to consult with Sumitomo Industries if they are uncertain how the baseboard will perform with their particular computer.

POWER CONSUMPTION

Due to the characteristics of image-processing devices, the IP90BD301/351 baseboards use a substantial amount of power as compared to other PC expansion boards. Refer to the specifications and make sure the personal computers and racks used provide sufficient power capacity.

SOFTWARE

The following software products are available for the IP90BD301/351 baseboard:

Image-Processing PC Control Libraries

1. Baseboard Control Library
2. Module Control Library
   a. Object Code Version for 1 & 2: Model IPCLE-300FD

      (Provided on 1.44 MB, 3.5" FD in MS-DOS Format)

   b. Source Code Version for 1 & 2: Model IPCLS-300FD

      Provided on 1.44 MB, 3.5" FD in MS-DOS Format)

   c. Sample software examples using contol libraries are provided in source code.

The image-processing module control library for PC contains the following function groups:

- Initializing function group
- Interrupt control function group
- IMSC control function group
- Image output control function group
- Image input control function group
- Frame memory control function group
- Image bus control function group
- Control function groups for individual modules

(Ask for the names of the most recent image-processing models.)

Note that the PC image-processing module control library does not include general image-processing routines such as filtering. Inquire separately about configuring image-processing systems and creating other applications.

(2) Image-Processing Module Control Library Source Code (Model IPCLS-300FD)

This is a set of source code for the PC image-processing module control library described above. All necessary source code is included, though the user must be aware of the following condition:

- ❑ Source code may be altered by the user. However, the supplier does not guarantee the operation of the source code after alteration. The supplier also does not provide support for altering source code.

Advice to IP90BD301/351 users:

The IP90BD301/351 baseboard has the following options:

(1) IP90BD301-series 4-channel input camera connecting cable (Model IPC-VC1).

(2) One cable for SIDIP bus connection between IP90BD301 and IP90BD351 (Model IPC-SC2).

(3) One cable for SIDIP bus connection between one IP90BD301 and two IP90BD351 (Model IPC-SC3).

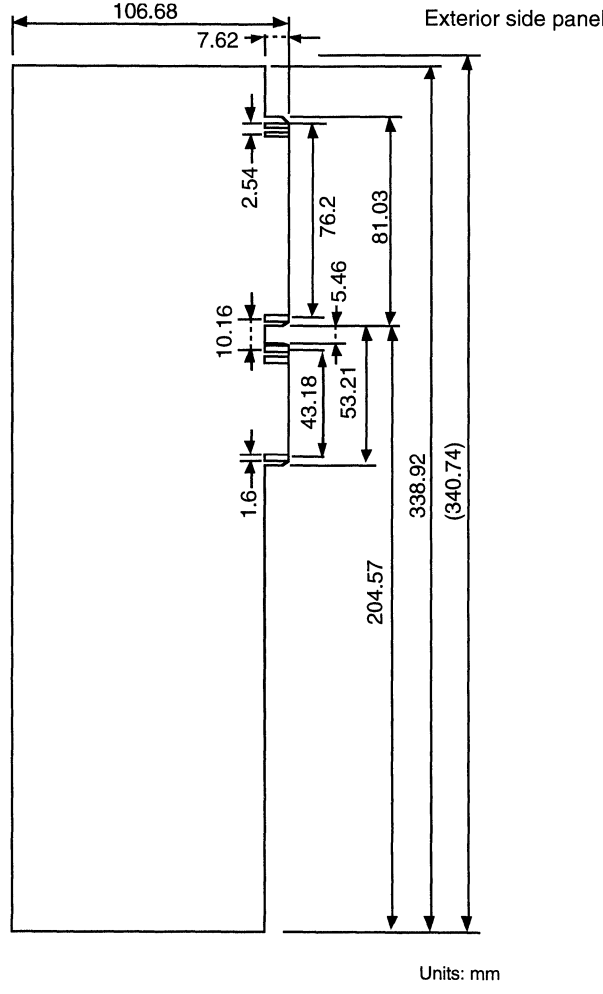


Figure 3. Mechanical Dimensions

Item		IP90BD301	IP90BD351	Remarks
Image input	Signal type	EIA	---	
	Channels	4 channels	---	
	Impedance	75W	---	
	Gain	Variable (±6 dB)	---	Software setting
	Offset	Variable (±0.5V)	---	Software setting
	Pin type	9-pin D-Sub	---	
Image output	Signal type	VGA/NTSC	---	
	Channels	1 channel	---	
	Impedance	75W	---	
	Pin type	15-pin HD D-Sub	---	
Pixel clocks	Input	12.5 MHz	---	
	Internal	20.00 MHz	---	
	Output	25.17 MHz	---	
Input synchron-ization signals	Horizontal	15.734 kHz	---	2:1 interlace
	Vertical	60 Hz	---	
Internal synchronization signals	Horiz. (min)	64 clock	---	Non-interlace
	Vert. (min)	64 clock	---	
	Horiz. (max)	64K clock	---	
	Vert. (max)	64K clock	---	
Output synchronization signals	Horizontal	31.5 kHz	---	VGA
	Vertical	60 Hz	---	
	Horizontal	15.734 kHz	---	NTSC
	Vertical	60 Hz	---	2:1 interlace
Frame memory	Size	1024 x 512	---	
	Effective area	640 x 480	---	
	Frames	4 frames	---	
	Host access	16-bit	---	
LUTs	Size	256 x 8-bit	---	
	No.	1 set (8 banks)	---	
ALUs	Bit width	16-bit	16-bit	Built into IMSC
	No.	2 sets	1 set	
	Logical calc. functions	AND, OR, NOT, EXOR	AND, OR, NOT, EXOR	
	Arithmetic calc. functions	+, –	+, –	
Barrel shifter	Bit width	16-bit	16-bit	Built into IMSC
Max/min comparator		Min(A, B), Max(A, B)	Min(A, B), Max(A, B)	Built into IMSC
External trigger	Freeze request	1 set	---	CMOS level
Host interface	Data width	ISA bus 8/16 bit	ISA bus 8/16 bit	
	Interrupt	1 level	1 level	Freeze/internal synch. signal
Board size (mm)		333.5x106.6	333.5x106.6	
Power consumption	+5V	approx. 2A	approx. 0.7A	without modules
	+12V	approx. 0.1A	---	without modules

Table 3. IP90BD301/351 Specifications

IP90BD550

Baseboard for VME Bus (6U)

Product Manual

Ver. E 1.3

Sumitomo Metal Industries, Ltd.

VME Extended Baseboard for Image-Processing Modules
The IP90BD550

FEATURES AND FUNCTIONS

- Four SIDIP standard image-processing module slots
- Maximum number of modules mountable:
  - Single-width: 4
  - Double-width: 2
  - (single- and double-width modules can be mixed)
- Two IP90C55 (IMSC) 8-bit image datastream bi-directional controller LSIs, allowing flexible image data transmission controlled by software
- Maximum image-processing space of 64K x 64K
- Operations between frames
- Expandable baseboard using a SIDIP standard global bus
- Motorola MVME162 controller connects to the MD-series modules
- VME A32 extended addressing
- VME D32 slave port
- One interrupt level
- Readable baseboard ID
- Single 5V (±5%) power source
- Board size: VME 6U (1 slot)
- Image-processing clock frequency:
  - 25 MHz (maximum)
  - 40 nsec/pixel transmission rate

APPLICATIONS

- Internal high-speed image-processing devices
- Evaluation of Sumitomo Metal Industries image-processing modules

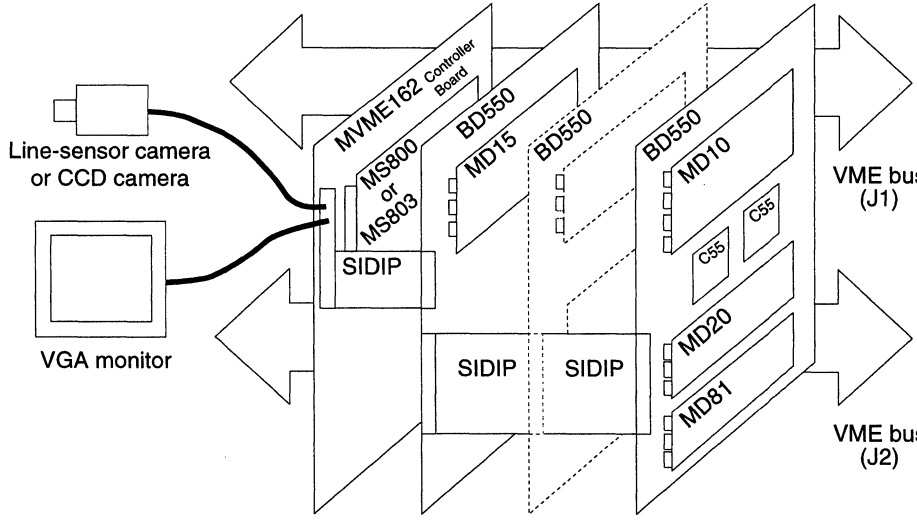


Figure 1: System Configuration

PRODUCT SUMMARY

The IP90BD550 VME extended baseboard includes two 8-bit image datastream bi-directional controller IP90C55 (IMSC) LSIs. The IP90C55s provide flexible data transmission between image-processing modules by using the SIDIP (Standard Interface for Digital Image Processing) interface. The IP90BD550 also includes a global bus to allow data transmission between baseboards, and providing easy system expandability.

Any of the four screens or frame memories can be used as an overlay frame memory, using the lower four bits in the frame memory as a 4-bit scale. This overlay data can display 15 colors using a D/A converter.

Each IP90C55 has twelve 8-bit I/O ports, a 16-bit ALU (or two 8-bit ALUs), and a 16-bit barrel shifter function (or two 8-bit barrel shifter functions), enabling real-time frame operations. (For details, please see the IP90C55 technical documentation.)

The IP90BD550 can hold up to single-width four image-processing modules or two double-width modules. Single- and double-width modules can also be mounted in combination.

Although the IP90BD550 has a 64K x 64K image-processing space, this space may be constrained by the modules mounted. The image-processing clock has a maximum frequency of 25 MHz.

The IP90BD550 is a VME 6U-sized board that occupies one slot in its host. The host access uses 32-bit addressing and 32-bit data. Interrupts are single-level, selected from seven available levels.

The IP90BD550 includes a board ID that can be read by the host, allowing the host to easily identify the types of modules mounted on each baseboard. (The IDs in each module can be set.) This can help simplify system construction.

Through the connections on the Sumitomo Metal Industries IndustryPack™ Boards mounted on the Motorola MVME162 controller board, the IP90BD550 provides the module series interface for the controller board.

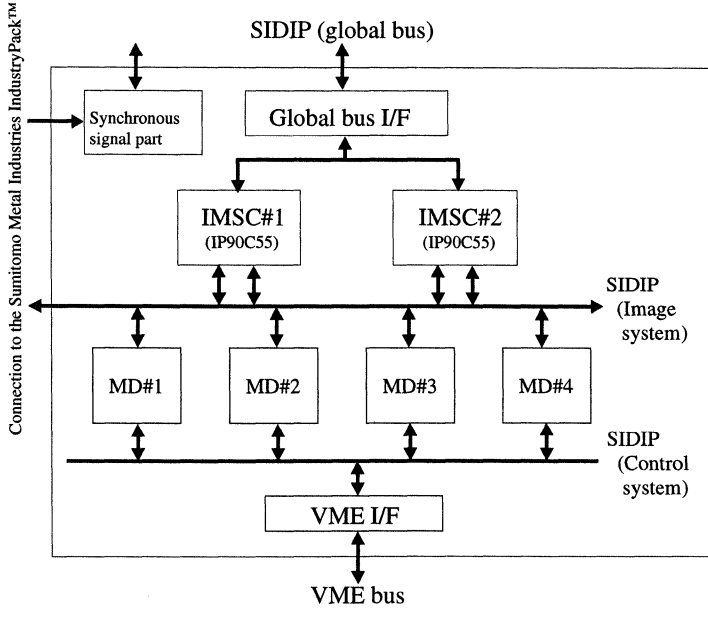


Figure 2: IP90BD550 Block Diagram

STRUCTURE AND OPERATION

Figure 2 shows a block diagram of the IP90BD550.

(1) Synchronous System

The IP90BD550 synchronous system uses the SIDIP standard. Because the IP90BD550 does not have a signal generation function in its synchronous system, it must receive one through the SIDIP bus. For example, when the IndustryPack is connected, the SIDIP standard synchronization signal from this IndustryPack is output to the global bus, thus synchronizing the entire expansion baseboard.

(2) Image System

The IP90BD550 digital image bus is based on the SIDIP standard. Its IMSC #1 and #2 ports (8 bits/port) are connected to each of the module's MD #1–#4 I/O ports. This allows software control of the IMSC #1 and #2 ports to provide flexible connections. The connections for the IndustryPack are also one-to-one with the IMSC ports.

The global bus has an 8-bit image bus, and the system can be expanded by connecting it to other baseboards.

Although the IP90BD550 uses interlaced and non-interlaced bi-directionality, only the non-interlaced works when connected to the IndustryPack. The modules mounted may also impose additional constraints.

(3) Operations Between Frames

The IP90BD550's two IP90C55s make real-time between-frame operations simple. They also make it easy to set an image background. (For details, see the IP90C55 technical documentation.)

Each IP90C55 has a 16-bit ALU and a 16-bit barrel shifter function, which can also be used as an 8-bit ALU x 2 and an 8-bit barrel shifter function x 2.

Logic calculation	AND OR NOT EXOR
Arithmetic operations	+ (addition) – (subtraction) carry operation
Shift operations	0–8 bit or 0–16 bit right shift
MIN/MAX	MAX (A, B) MIN (A, B) comparative selection

(ALU: 8-bit ALU x2, equivalent to the 74181)

Table 1: IMSC Internal Calculation Functions

(4) The VME Interface

The IP90BD550 is a VME 32-bit extended address, D32 (D32, D16, D08(EO), D08(O)) slave board. It occupies 1 Mb of a 4-Gb space, (the base address of this 1-Mb space is set with jumpers). 512 Kb of the 1 Mb is allocated: 128 Kb each to MD #1–#4.

Each module's interrupt signal uses a single line. These signals are combined into a single line in the IP90BD550, which is then output on one of the seven levels in the VME bus.

(5) Module Mounting

The IP90BD550 has four slots for mounting SIDIP modules. (See Figure 3.) The board can mount four single-width or two double-width modules. Single- and double-width boards can also be mounted in combination. Third-party boards can also be mounted as long as they conform to the SIDIP standard.

(6) The Global Bus

The IP90BD550's global bus is a SIDIP standard system expansion bus for connecting to a baseboard series. It includes four 8-bit image busses and two synchronization busses.

(7) Connecting with the IndustryPack

The IP90BD550 can be connected to IndustryPack boards, which are mounted on the MVME162 board. This provides for high-speed image processing using MD-series modules in the MVME162 system.

The bus includes one 8-bit image input bus, one 8-bit image output bus, and one line for the synchronization system.

IP90BD550 PRODUCT SPECIFICATIONS

Item	Specifications
• Image processing space	• Maximum of 64K x 64K
• Between-frame operations (for each IP90C55)	• 16-bit ALU (8 -bit x 2) • 16-bit barrel shifter (8-bit x 2) • Image paint out
• I/F with the VME	• D32, D16, D08(EO), D08(O) • 32-bit expanded address compatible • 1 interrupt level • 1 Mb selected from a 4-Gb space
• IndustryPack I/F	• Two 8-bit image busses that can be connected directly to the IP90MS800/803 IndustryPack image bus using a Motorola MVME162 board.
• Expansion global bus	• Four 8-bit image busses
• Board size	• VME board (6U size)
• Image processing clock frequency	• Maximum of 25 MHz (40 nsec/pixel transmission rate)
• Maximum module mounting	• 4 (single-width) • 8 (double-width) (single-width and double-width can be mounted together)
• Operating temperatures	• 10–35°C RH 20 -80%)
• Power supply voltage	• 5V ± 5%
• Other	• Uses area of interest synchronization signals

SIDIP (Standard Interface for Digital Image Processing) Specifications

- Image System
  - Image address bus: 8-bit X 4

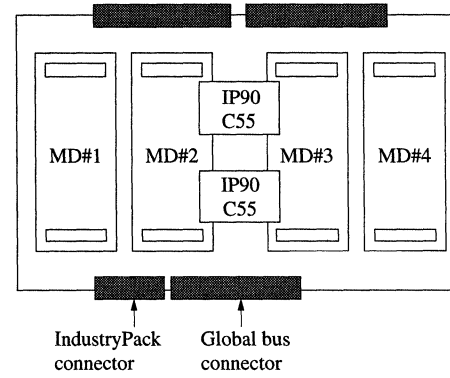


Figure 3: IP90BD550 Exterior

  - Image clock: Maximum of 25 MHz/pixel
- Control System
  - Address bus: 16-bit memory space
16-bit register space
  - Data bus: 16-bit width
(byte access possible)
  - Interrupts: One level
- Other
  - Board Size
Single-width: 48.8 x 127.0 mm
Double-width: 98.6 x127.0 mm

MODULE LINEUP

- IP90MD10 Labeling Processor
- IP90MD15 Average Value Compression/ Simple Expansion
- IP90MD20 Rank Value Filter
- IP90MD25 Spatial/Logical Filter
- IP90MD81/ 01/05 Histogram and Projection
- IP90MD100 Frame Memory

IndustryPack Products

IP90MS800

Area Sensor Image Frame Grabber

Product Manual

Ver. E 1.2

Sumitomo Metal Industries, Ltd.

The Area Sensor Image Frame Grabber Board The IP90MS800

The IP90MS800 is a monochrome frame grabber board that captures 8-bit grayscale images from monochrome cameras conforming to the EIA or CCIR standard, producing output for standard VGA displays.

The board has four frames of 512 x 480 pixels x 8-bit frame memory, and can be accessed by the CPU.

The board also uses the SIDIP (Standard Interface for Digital Image Processing) interface proposed by Sumitomo Metal Industries to provide a wide range of expandability.

FEATURES AND FUNCTIONS

(1) FEATURES

- An 8-bit monochrome frame grabber based on the industry standard IndustryPack™ (IP) developed by GreenSpring Computers.
- Can be mounted on the Motorola MVME 162 Board series and in the double-width GreenSpring Computers I Carrier series.
- A SIDIP-based image bus.
- High-speed processing made possible by the ability to interface directly with SIDIP-based image-processing modules mounted on the board.
- An external trigger (TTL level) input to frame-freeze the frame memory.
- Interrupt request output capability.

(2) FUNCTIONS

- Image Input
  4 sets of EIA or CCIR standard inputs
  512 (horizontal) x 480 (vertical) pixels x 8-bit
- Image Output
  VGA color display with 256 colors selected from a 16.7-million color palette
  512 (horizontal) x 480 pixels (vertical) x 8-bit display.
  4-bit, 15-color overlays
- Frame Memory
  4 frames of 512 (horizontal) x 480 pixels (vertical) x 8-bit frame memory (two of which can be accessed by the CPU through serial access)
- LUT (Look-Up Table)
  256 pixels x 8 bit
- Maximum IP Bus Transmission Rate
  8 Mb/second (16 bits)
- Maximum SIDIP Bus Transmission Rate
  50 Mb/second (16 bits)
- Board Size
  IP double-width (91.4 x 99.1 mm)

THE PRODUCT

The IP90MS800 specifications are detailed in Table 1, and a block diagram of its functions is shown in Figure 1.

The board is equipped with both an 8-bit A/D converter and an 8-bit D/A converter. The A/D converter contains a 256 x 8-bit look-up table (LUT), and can transform the grayscale image data input from the camera. The D/A converter is also equipped with a color palette that can display 256 colors selected from 16.7 million available colors, permitting a pseudo-color display.

The image input can use up to four monochrome cameras conforming to the EIA or CCIR standards (the standard is selected by the software).

The image output signal conforms to the VGA standard, and can display 512 (horizontal) x 480 pixels (vertical).

The IP90MS800 is equipped with four frame memories of 512 (horizontal) x 480 pixels (vertical) x 8-bit. These frame memories store image data written from the camera, and also data to be displayed on the VGA monitor. They can be accessed by the CPU serially through the MVME162.

One of the 4 screens or frame memories can be used as overlay frame memory, using the lower four bits in the frame memory as a 4-bit scale. This overlay data can display 15 colors using the D/A converter described above.

The IP90MS800 is connected directly to the image bus modules in the Sumitomo Metal Industries' image-processing module series.

This simplifies high-speed image processing and allows image data transfer rates of up to 25 MHz.

The board size is the IndustryPack™ double-width standard (91.4 x 99.1 mm).

	Item	Specifications	Notes
Image input	Signal format	EIA or CCIR standard	Software selected
	Input terminals	Four sets	Requires special cable
	Impedance	75 ohms, fixed	
	Horizontal sync. signal	15.734/15.625 KHz	EIA/CCIR
	Vertical sync. signal	59.94/50 Hz	EIA/CCIR
	Quantum number	8 bits	
	Maximum number of pixels	512 (horizontal) x 480 (vertical) pixels	For both EIA and CCIR
Image output	Signal format	VGA standard	
	Output terminals	1 set each: RGB, HD, VD	Requires special cable
	Impedance	75 ohms	
	Horizontal sync. signal	31.5 KHz	
	Vertical sync. signal	60 Hz	
	Display clock	25.175 MHz	
	Number of colors	256 colors from a 16.7-million color pallet	
	Maximum number of pixels	512 (horizontal) x 480 (vertical) pixels	
	Overlay	15-color display	
Frame memory	Number of pixels	512 (horizontal) x 480 (vertical) pixels	
	Gradations	8-bit	
	Number of frames	4	
	CPU access	Serial access	Only 2 of the 4 frames
	CPU access rate	8 Mb/sec (16-bit)	IP bus
	Control function	Freeze	
LUT	Capacity	256 x 8 bits	
SIDIP	Display clock	25.175 MHz	
	Buses	8 bits x 2 (1 input, 1 output)	
Other	Board size	91.4 x 99.1 mm	IP double-width
	Power consumption	DC+5V:670 mA, DC+12V:20 mA	Standard value

Table 1: IP90MS800 Specifications

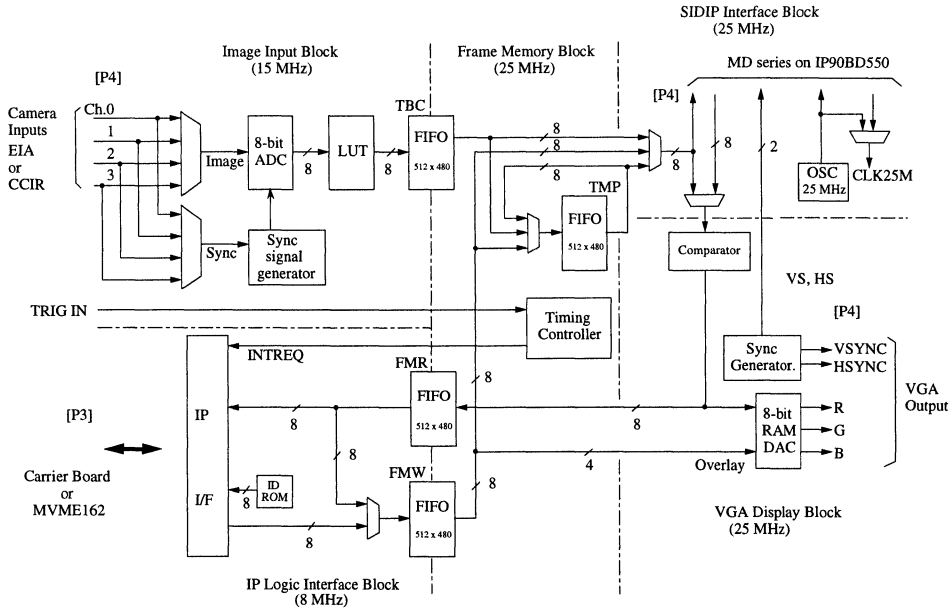


Figure 1: Block Diagram

STRUCTURE

The internal structure of the IP90MS800 is divided into three sections based on the clock frequencies used. Figure 1 shows the functional block diagram. The functional blocks are listed below:

- An image input block (@ 15 MHz) that performs A/D conversion on the camera signal (interlace)
- A frame memory block (@ 25 MHz) that receives the interlaced data from the image input block and converts it to non-interlaced data and then into a VGA signal
- A VGA display block (@ 25 MHz)
- An IP Logic interface block (@ 8 MHz) that outputs to the host system
- A SIDIP Interface Block (@ 25 MHz)

Data transfers between these four blocks are done on a FIFO basis by TBC (Time Base Corrector), FMR (Frame Memory Read), and FMW (Frame Memory Write).

(1) IMAGE INPUT PROCESSOR BLOCK (15 MHz)

The image input processor block performs A/D conversion on the video signal from the camera, and then writes that data to the TBC frame memory through the LUT. Moreover, the data is stored in an interlaced form.

1.1) The Timing Signal Generator

This circuit generates vertical and horizontal synchronization signals, field signals, image clock signals, etc. from the input video signal.

1.2) The A/D Converter Circuit (ADC)

This circuit performs A/D conversion on the input video, converting it into 8 bits, 256 levels. A multiplexer is used to allow a single channel to be selected from the four available.

1.3) The Look-Up Table (LUT)

This is RAM that allows discretionary changes to be made to the 8-bit, 256-level output of the A/D converter circuit.

1.4) The TBC Frame Memory

This is a 512 (horizontal) x 480 (vertical) pixels x 8-bit FIFO memory. Image data is converted from interlaced to non-interlaced using this TBC and the TMP (temporary) frame memory.

Transferring the contents of the TBC into the FMR frame memory allows the host to indirectly read its contents.

(2) THE FRAME MEMORY BLOCK (25 MHz)

The frame memory block converts the image data from interlaced to non-interlaced, then to binary form, and finally to a VGA signal.

2.1) The TMP Frame Memory

TMP is a 512 (horizontal) x 480 (vertical) pixels x 8-bit FIFO memory. The TMP converts the interlaced data received from the TBC memory into non-interlaced form, then stores it in the TMP frame memory. FMW and TBC memory images can be copied directly to the TMP frame memory.

Transferring the contents of TMP into the FMR frame memory allows the host to indirectly read the data.

(3) THE VGA DISPLAY BLOCK

This circuit generates VGA picture signals, and also the synchronization signal. The picture signal consists of image data from the output of the comparator, and 4-bit overlay data from the frame memory FMW.

3.1) Comparator

This circuit converts image data into binary (00h, FFh) or ternary form (01h, 02h, 04h).

(4) THE SIDIP EXTENDED INTERFACE CIRCUIT

This circuit connects the image bus of the IP90BD550 module baseboard to the IP90MS800 and VME buses. This image bus conforms to the SIDIP standard.

The extended interface circuit consists of an output image data bus (8-bit), an input image data bus (8-bit), an image clock, and frame synchronization signal outputs (VS, HS).

The SIDIP image bus in the baseboard is synchronized by the frame synchronization signals (VS, HS) from the IP90MS800 25-MHz system.

(5) THE IP INTERFACE BLOCK

The host CPU controls the IP90MS800 as a whole through the registers of the 8-MHz IP interface block. Image data is input and output through the FMR and FMW frame memories. Although each frame memory inputs and outputs in byte units, byte-to-word and word-to-byte conversion is provided, allowing the host to access in word (two-byte) units.

5.1) Frame Memory FMR

FMR is a 512 (horizontal) x 480 (vertical) pixels x 8-bit FIFO memory used when the host reads image data.

FMR image data can be transferred to the FMW frame memory without being accessed by the host.

5.2) Frame Memory FMW

FMW is a 512 (horizontal) x 480 (vertical) pixels x 8-bit FIFO memory. Data written by the host passes through the FMW and is transferred to the 25-MHz system (the frame memory block and the VGA display blocks).

The FMW also overlays VGA output on normal image data. The four least significant bits are assigned to this overlay data.

EXAMPLE APPLICATION

The IP90MS800 has two principle functions—frame grabbing and display —and its hardware processes include image grayscale transformation using a LUT. Because of this, a simple system could use the IP90MS800 combined with the Motorola MVME162 series to make a single-board image I/O device. Figure 2 shows this minimal system configuration. In this case, image processing is executed by software through the CPU (an MC68040) mounted on the MVME162. Because the IP90MS800 is a double-width card, the MVME162 can mount one double-width or tow single-width IndustryPacks™. These IndustryPacks™ can include A/D converters for calculations, parallel interfaces, etc. An image processing system with a single VME board would allow the device structure to be very compact. In this example, neither a VME bus system rack nor a mother board is needed, leading to a substantial reduction in cost.

Furthermore, when the algorithms in the software execute too slowly to be practical, hardware can be added to the minimal system described above. Upgrading to a high-speed image-processing system is easily accomplished. The IP90MS800 SIDIP interface is also used in other image-processing modules that Sumitomo Metal Industries has already brought to market. By combining the IP90MS800 with the IP90BD550 VME baseboard (which also uses the SIDIP interface and is the carrier board for the VME image-processing modules), the IP90MS800 can have a direct hardware connection to the image-processing module. This makes it possible to replace slow software processes with hardware, thus making the system a high-speed image-processing device.

Figure 3 shows an example of a system that uses an image-processing module. The SIDIP interface is an open system that will be enhanced by Sumitomo, and allows users to develop specialized modules or boards. Third-party enhancements are also expected, so that the flexibility of the system will increase even further.

As previously noted, a minimal system using the IP90MS800 and the MVME162 would be used in system configurations not requiring great speed or algorithm checking. Conversely, firms requiring high-speed systems (such as firms using manufacturing lines or high-speed processing) can extend the lives of their software and reduce the work of system configuration by adding Sumitomo Metal Industries' image-processing modules to their software system, ultimately reducing costs.

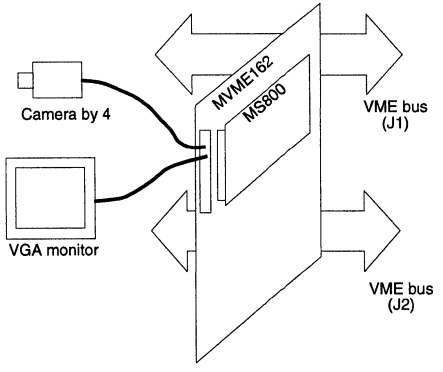


Figure 2: Example of a Minimal System

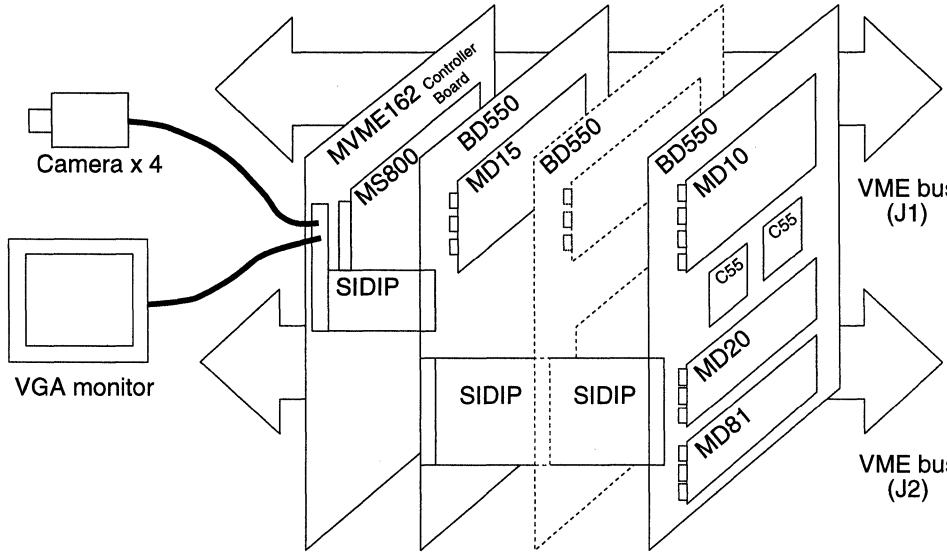


Figure 3: Example of an Expanded System Structure

SAMPLE SOFTWARE

A printout of a sample source list (to be used as a reference) for controlling the IP90MS800 is included in the package. The following are included in the list as rough functions to be used as reference materials:

- Initialize
- Set the A/D converter block (the image input processor block) LUT
- Set the analog video input channel
- Set the palette for the display
- Set to binary
- Toggle the input signal between EIA and CCIR
- Read/clear the status registers
- Interrupt management

NOTE: Be aware that these sample programs, provided only as references, carry no warranties and may not be supported with follow-up versions, etc. Also, note that the hardware is supplied with printed material and a floppy disk (MS-DOS format 3.5-inch).

PIN CONNECTORS

The IP90MS800 follows the IP standard. Its external dimensions are double-width, though it functions as do single-width modules. In other words, the logic I/O connectors consist of P3 and P4; the board has no P1 or P2. P3 and P4 connect to slot B or slot D of the MVME162 (or the IP carrier) series. (However, the IP90MS800 cannot connect to slot D of the GreenSpring Computers VIPC610.)

(1) I/O CONNECTOR P4

P4 consists of a camera input, a VGA signal output, an extended interface, and an external trigger input. The configuration of the pins is shown in Table 2. Also, P4 has been converted to the 50-pin flat cable plug on the MVME162 (or the IP carrier). The pin number for each signal is the same for each connector.

(2) EXPLANATION OF SIGNALS

All signals are TTL levels except for AS0–AS3, and R, G, and B. An asterisk (*) indicates negative logic.

Camera input:

- AS0–AS3: These are the four channel inputs for EIA/CCIR cameras. Each channel has synchronization and image signals, and the channel is selected through software.
- AG0–AG3: These are the ground connections for AS0–AS3.

VGA output:

The VGA output signal has a resolution of 640 x 480 pixels (although only 512 x 480 pixels can be displayed), with a horizontal synchronization of 31.5 KHz and a vertical synchronization of 60 Hz.

SIDIP Extended Interface:

This manual assumes that the VME IP90BD550 baseboard is connected to the SIDIP Extended Interface.

The SIDIP extended interface signals connect to the CN-IP connector of the IP90BD550 through a flat cable (connecting pin 1 to pin 1, pin 2 to pin 2, etc.).

(3) THE ADAPTER CABLE

The signals input through I/O connector P4 pass through the 50-pin flat cable plug on the MVME162 (or the IP carrier), which connects to the external devices. Below are the specifications for the cable plug (see Table 2) and the adapter cable (see Table 4) for connecting to external devices. Note that the requirements for the adapter cable vary depending on the environment (e.g., external noise changes the requirements of the cable). Thus, the cable specifications below are merely tentative guidelines. Adjust these specifications as necessary.

Sumitomo Metal Industries has standard adapter cables are available for purchase. Please request a catalog of hardware products and optional products (cables and connectors) from your sales representative.

- Receptacle Connector (A)
  KEL Co. 6230-050-601

NOTE: This connector extends slightly higher than the parts on the standard VME board. However, if the tapped strain relief (for use in disconnecting the cable) is removed, the assembly falls within the specified dimensions.

- Receptacle Connector (B)
  KEL Co. 6230-030-601

- Flat Cable (C)
  AWG#28 -- 30 pins
  The length between sides (A) and (B) is less than 70 mm.

- Flat Cable or Twisted Flat Cable (D)
  AWG#28 -- 2 pins
  The flat cable can be up to 200 mm long, while the twisted flat cable can be up to 450 mm long. For greater lengths, use a shielded cable.

- Flat Cable or Twisted Flat Cable (E)
  AWG#28 -- 10 Pins
  The flat cable can be up to 70 mm long, while the twisted flat cable can be up to 200 mm long. For greater lengths, use a VGA cable.

- Flat Cable or Twisted Flat Cable (F)
  AWG#28 -- 8 Pins
  The flat cable can be up to 50 mm long, while the twisted flat cable can be up to 100 mm long. For greater lengths, use 75-ohm coaxial cable.

Classification	Pin number	Signal name	Function
	1	DGND	Ground
	2	S_ICLK	Source oscillator image clock output
	3	DGND	Ground
	4	E_ICLK2	Image clock 2 input
	5	DGND	Ground
	6	E_OUT0	Image data output (bit 0)
	7	E_OUT1	Image data output (bit 1)
	8	E_OUT2	Image data output (bit 2)
	9	E_OUT3	Image data output (bit 3)
	10	DGND	Ground
Extended	11	E_OUT4	Image data output (bit 4)
	12	E_OUT5	Image data output (bit 5)
	13	E_OUT6	Image data output (bit 6)
	14	E_OUT7	Image data output (bit 7)
SIDIP	15	DGND	Ground
	16	E_IN0	Image data input (bit 0)
	17	E_IN1	Image data input (bit 1)
	18	E_IN2	Image data input (bit 2)
	19	E_IN3	Image data input (bit 3)
Interface	20	DGND	Ground
	21	E_IN4	Image data input (bit 4)
	22	E_IN5	Image data input (bit 5)
	23	E_IN6	Image data input (bit 6)
	24	E_IN7	Image data input (bit 7)
	25	DGND	Ground
	26	S_HS*	Horizontal synchronization signal output
	27	DGND	Ground
	28	S_VS*	Vertical synchronization signal output
	29	DGND	Ground
	30	N/C	Not used
External	31	E_TRIG*	External trigger input
Trigger Input	32	TGND	Ground for external trigger input
	33	HD*	Horizontal sync. output (TTL, negative)
	34	VGA_GND	Ground for synchronization signal
	35	VD*	Vertical sync. signal output (TTL, negative)
VGA	36	VGA_GND	Ground for sync. signal
Signal	37	R	R signal output
Output	38	R_GND	R signal ground
	39	G	G signal output
	40	G_GND	G signal ground
	41	B	B signal output
	42	B_GND	B signal ground
	43	AS3	Camera input (channel 3)
	44	AG3	Camera input ground (channel 3)
	45	AS2	Camera input (channel 2)
Camera	46	AG2	Camera input ground (channel 2)
Input	47	AS1	Camera input (channel 1)
	48	AG1	Camera input ground (channel 1)
	49	AS0	Camera input (channel 0)
	50	AG0	Camera input ground (channel 0)

Table 2: I/O Connector pins and P4 pins

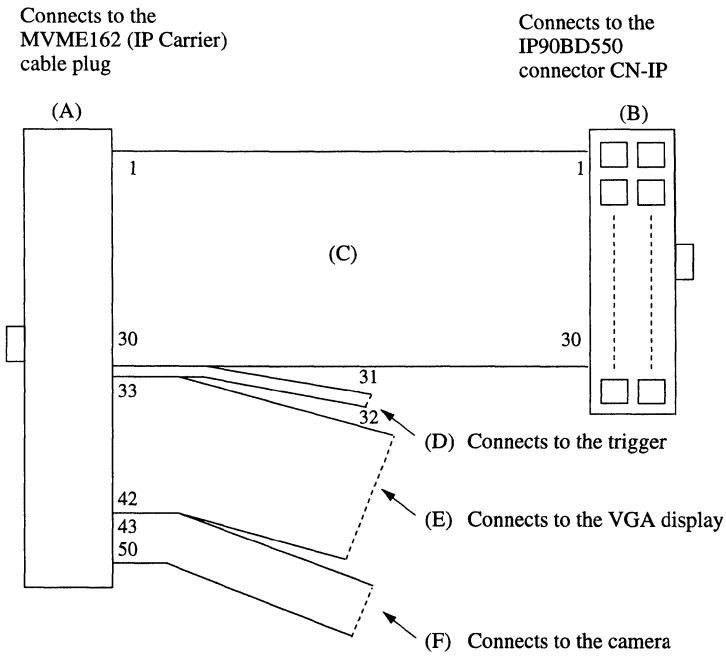
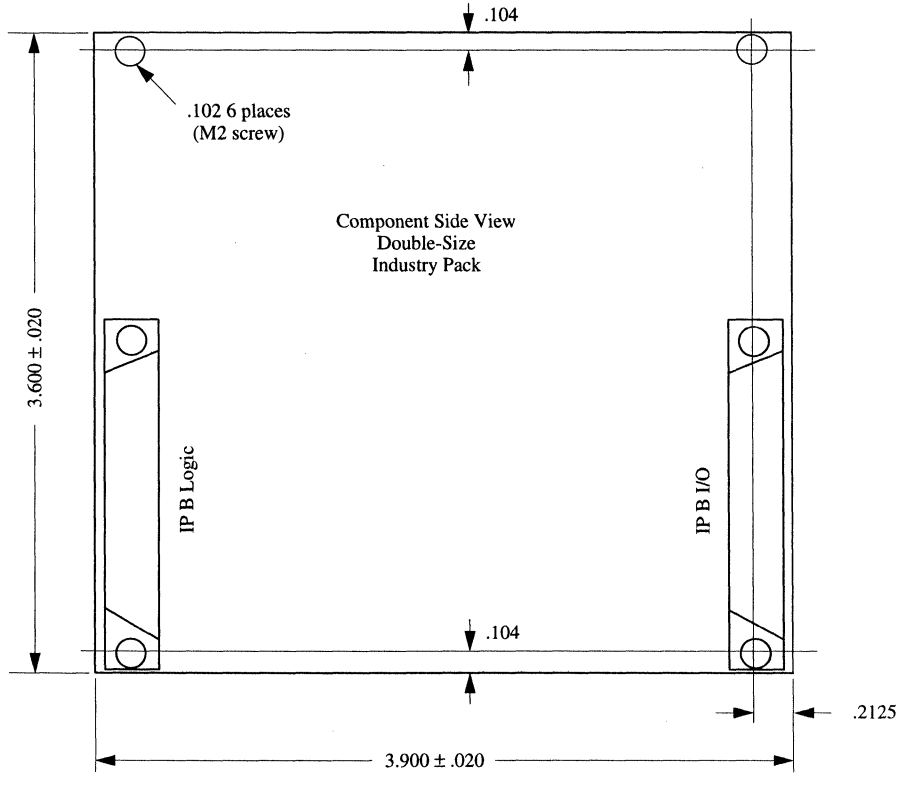


Figure 4: Adapter Cable

MECHANICAL DIMENSIONS

Figure 5 shows the size of the board and the layout of the connectors. The size of the board and the layout of the connectors are as per the IndustryPack™ (IP) standards of GreenSpring Computers in the United States.



Note 1: All units are in inches

Notes:

① The external dimensions of the board follow the IP double-width standard.

② The arrangement of the connector pins and their mechanical dimensions are as per the IP standard.

Figure 5: Board Dimensions and Locations of Connectors

IndustryPack Products

IP90MS803

Line Sensor Image Frame Grabber

Product Manual

Ver. E 1.2

Sumitomo Metal Industries, Ltd.

Line Sensor Image Frame Grabber Board
IP90MS803

The IP90MS803 is a monochrome frame grabber board that includes a line sensor input interface, and a 1,296-Kb frame memory that can be accessed by the CPU. It can capture images measuring 256 to 7936 pixels horizontally, and 164 to 5184 lines vertically. Its output images conform to the VGA standard.

The IP90MS803 can build two-dimensional images from one-dimensional input images (using its frame memory). It can display any VGA-size (640 x 480) portion of the input image, and can also produce a pseudo-color display using a color palette.

The IP90MS803 uses the SIDIP interface proposed by Sumitomo Metals Industries to provide a wide range of expandability.

FEATURES AND FUNCTIONS

(1) FEATURES

- An 8-bit, 256-gradation frame grabber for line sensors with the industry standard IndustryPack™ (IP) developed by GreenSpring Computers.
- Can be mounted on the Motorola MVME162 Board series and the double-width GreenSpring Computers IP Carrier series.
- Incorporates a SIDIP-based image bus.
- High-speed processing by directly interfacing with the SIDIP-based image processing module mounted on it.

(2) FUNCTIONS

- Image Input
  - Horizontal: 256–7936 pixels
  - Vertical: 164–5184 lines
  - Pixel clock frequency: 156.25 KHz–20 MHz
  - Gradation: 8 bits
- Image Output
  - VGA pseudo-color display uses 256 colors selected from palette of 16.7 million colors
  - Can display 640 (horizontal) x 480 (vertical) x 8 bits
- LUT
  - 256 x 8 bits
- Maximum Transmission Rate of IP Bus
  - 8 Mb/sec (16 bits)
- Maximum Transmission Rate of SIDIP Bus
  - 50 Mb/sec (8 bits x 2)
- Board Size
  - IP double-width (91.4 x 99.1 mm)

PRODUCT OVERVIEW

Table 1 gives the IP90MS803's specifications, and Figure 1 shows a block diagram of its functions.

The IP90MS803 connects to a line sensor with up to 7,936 pixels that supplies RS-422 standard clock and start pulses. The clock pulse can be set as high as 20 MHz, and the start pulse can be set at intervals from 11 to 32,830 clock cycles. (Clock frequency and start pulse intervals can be set by software.)

Input image data is converted into an 8-bit digital image data by an A/D converter. The grayscale image data can also be changed using a LUT in the A/D converter.

The IP90MS803 includes two frame memories: one for input images, and one for displayed images. The size (horizontal x vertical) of the image input frame memory can be set by software: for example, 256 x 5184, 1024 x 1296, 2048 x 684, 4096 x 324, or 7936 x 164. The gradation is 8 bits.

	Item	Specifications	Notes
Image input	No. of devices connected	1	Requires external power supply
	Video signal input voltage	0–2.5 V, 0–5 V, 75 ohms	Jumper switching
	Camera drive pulses	Clock and start pulses	RS-422
	Camera clock	Maximum 20 MHz, 1/1–1/128 divisions, 156.25 KHz-20 MHz	Software selected
	Camera drive signal	Camera clock and start pulses	RS-422
	Clock to video signal ratio	1:1, 1:2, 2:1	Can be set by software
	Start pulse cycle	11–32830 clock cycles	Can be set by software (line cycle)
	Input start position	7–16404 clock cycles	Can be set by software (video rise position)
	Start pulse H-level width	2–5 clock cycles	
	Quantum number	8 bits	
	Maximum number of pixels	256 (256 x 5184) 1024 (1024 x 1296) 2048 (2048 x 684) 4096 (4096 x 324) 7936 (7936 x 164)	Can be set by software (typical examples shown at right)
Image Output	Signal format	VGA	
	Output terminals	RGB, HD, VD (1 set each)	Requires a special cable
	Impedance	75 ohms	
	Horizontal sync signal	31.5 KHz	
	Vertical sync signal	60 Hz	
	Display clock	25.175 MHz	
	Number of colors	256 out of 16.7 million	
	Maximum size	640 (horizontal) x 480 (vertical)	
Image Input Frame Memory	Capacity	1296 Kb (324K x 8 bits x 4)	
	Grayscale	8 bits	
	Number of frames	1	
	CPU access	Serial access	
	CPU access rate	8 Mb/sec (16 bits)	IP bus
	Control function	Freeze	
Video Display Frame Memory	Display size	640 (horizontal) x 480 (vertical)	VGA
	Gradation	8 bits	
	Number of frames	1	
	CPU access	Serial access	
	CPU access rate	8 Mb/sec (16 bits)	IP bus
LUT	Capacity	256 x 8 bits	
SIDIP	Video clock	25.175 MHz	
	Number of buses	8 bits x 2	
Misc.	Board size	91.4 x 99.1 mm	IP double-width
	Power consumption	5 VDC 980 mA, 12 VDC 23 mA, -12 V 23 mA	Standard value

Table 1: IP90MS803 Specifications

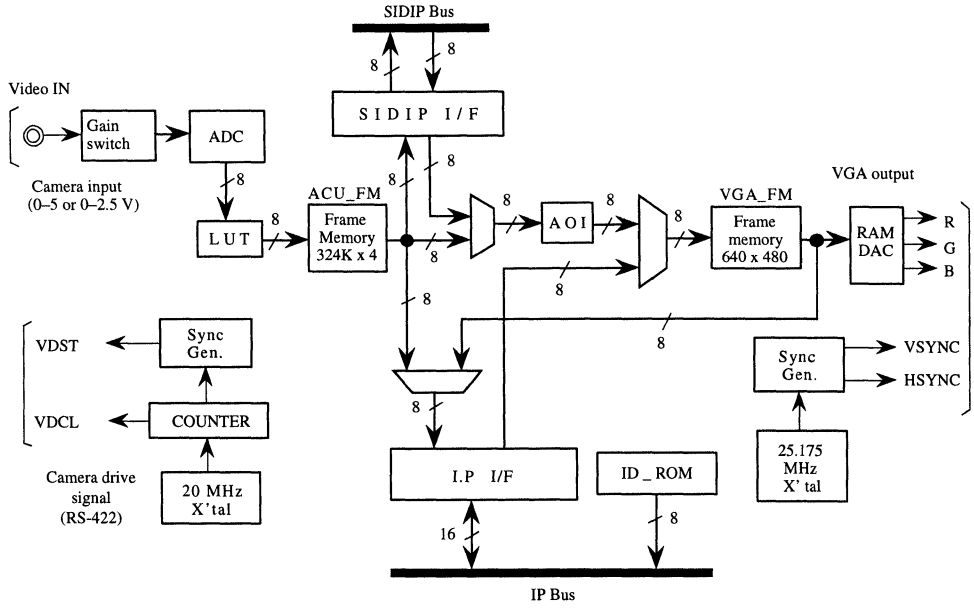


Figure 1: Block Diagram

The display frame memory can hold images up to 640 x 480 pixels, and conforms to the VGA standard. Input images are displayed by transferring the desired VGA-size display portion of the input frame memory to the display frame memory through an internal image bus.

Image data stored in the display frame memory is converted by the D/A converter into VGA video signals. The D/A converter is equipped with a color palette that can display 256 colors selected from 16.7 million colors, permitting a quasi-natural color display.

The IP90MS803 uses the SIDIP interface and can be connected directly to the image bus of modules from Sumitomo Metals Industries' image-processing module series. This simplifies high-speed image processing, and provides data transfer rates of up to 25 MHz.

The board size is the IndustryPack™ double-width size (91.4 x 99.1 mm).

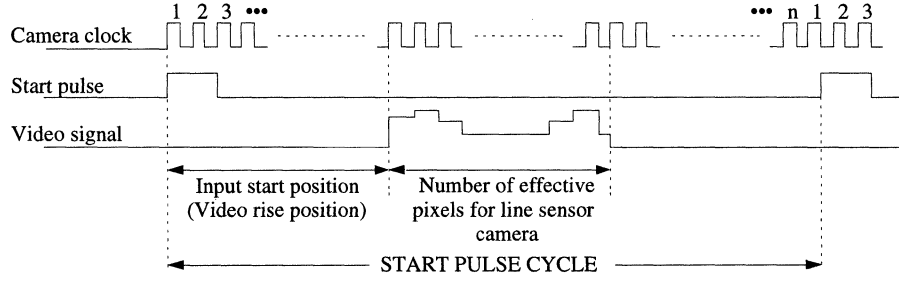


Figure 2: Line Sensor Camera Signals

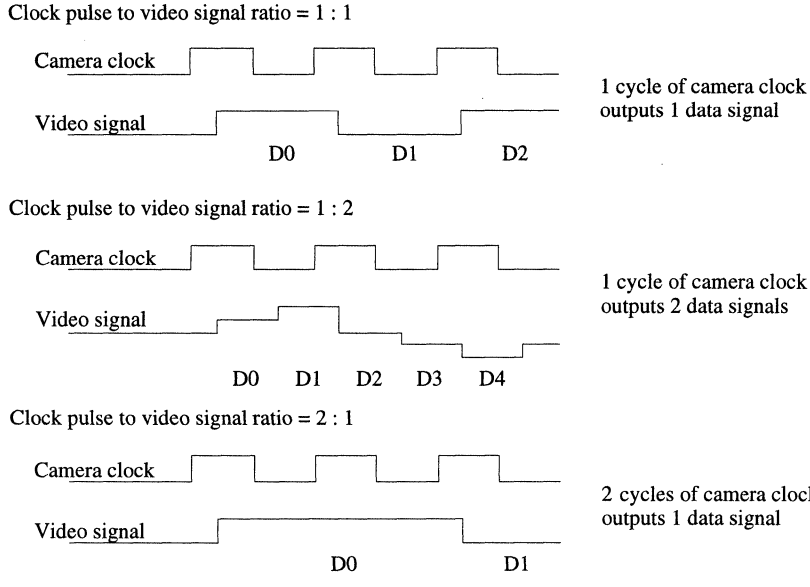


Figure 3: Clock Pulse to Video Signal Ratio

LINE SENSOR CAMERA

The IP90MS803 supports line sensor cameras that conform to the image input specifications in Table 1.

Line Sensor Camera Terminology

The following terms are used to describe image input from line sensor cameras (see Figure 2).

- Video Rise Position (Read Start Position)

  The number of clock cycles generated after the start pulse rises until the first effective pixel is output.

  The range of the number of effective pixels is written in the data input frame memory, starting at this position.

- Start Pulse Cycle

  The number of clock cycles generated between the rises of the start pulse and the rise of the next pulse.

  Note: Each clock number is determined by counting the camera clock at the rise of a start pulse as 1.

- Number of Effective Pixels for Line Sensor Camera

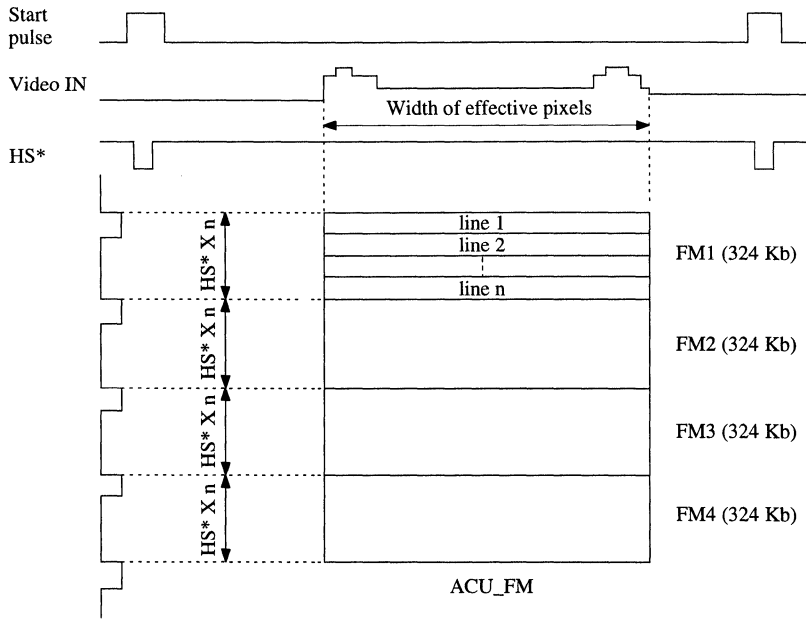
  This can be any of the three types shown in Figure 3, depending on the clock pulse to video ratio.

  Note: Camera Drive Power Source

  The IP90MS803 does not include a power supply to drive the camera; this must be purchased separately.

STRUCTURE

The synchronization signal generator stores one-dimensional image data input from the line sensor camera in the data input frame memory as a two-dimensional image, then generates synchronization signals for the camera interface. Depending on the destination, the camera clock is switched between the data input frame memory and the VGA output frame memory from among the SIDIP interface 25 MHz, IP Logic Interface 8 MHz, and internal real-time transfer (which performs VGA output while reading data). With internal transfer and SIDIP



HS* and VS* are generated by the line sensor camera synchronzation signal generator.

Figure 4: ACU_FM Configuration

transfer, the camera clock can be programmed to select a 640 x 480 area from the data input frame memory by using the AOI (Area Of Interest) extraction circuit.

(1) CAMERA INTERFACE BLOCK

1.1) Input Section

This section inputs video signals from the line sensor. Its input voltage level can be switched between 0 and 5 or 0 and 2.5 V by opening and closing the jumper post.

1.2) Camera Drive

This outputs line sensor camera drive signals (camera clock and start pulse) that comply with RS-422 standards.

(2) DATA INPUT BLOCK

2.1) ADC (A/D Converter)

This circuit performs A/D conversion on the input video signal, converting it into 8 bits and 256 gradations.

2.2) LUT (Look-Up Table)

This RAM is used to convert the A/D converter's output to 8 bits and 256 gradations.

2.3) Data Input Frame Memory (ACU_FM)

Figure 4 shows the ACU_FM structure. It selects only the range of effective pixels from the video signals output by the line sensor camera, and stores them in the first line of the ACU_FM as one-line data. It also stores the effective pixel range that begins with the next start pulse in the second line, up to line n set by the software, thus forming a frame (FM1).

This process is repeated four times to complete the frames FM1–4 that form the ACU_FM.

Figure 4 shows that the horizontal area size of the ACU_FM is determined by the number of effective pixels, with the ACU_FM set as an area ranging in size from 256 x 5,184 pixels to 7,936 x 164 pixels.

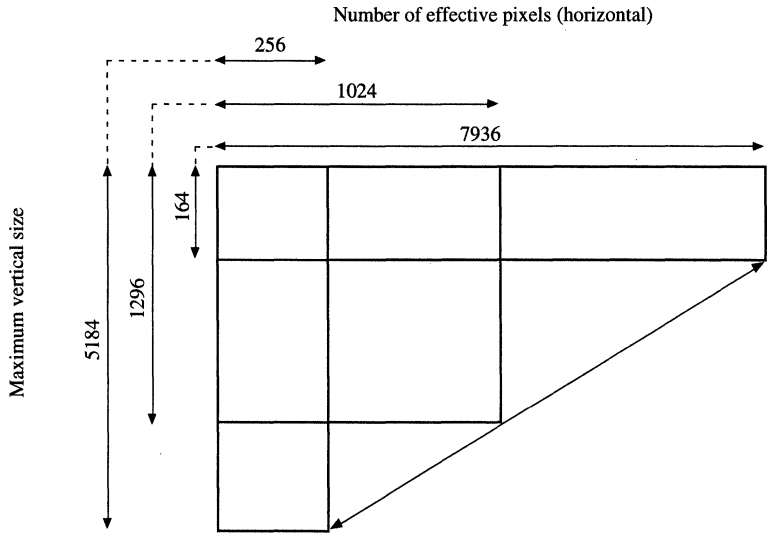


Figure 5: ACU_FM Area Setting Examples and Maximum Values

2.4) Sync. Generator for Line Sensor Camera

The software sets the start pulse width for the line sensor camera and the vertical area width for the number of effective pixels of the camera, and generates the synchronization signal needed for the camera interface.

(3) VGA OUTPUT BLOCK

3.1) VGA Output Frame Memory (VGA_FM)

Uses a 324-Kb frame memory.

VGA output produces a 640 (horizontal) x 480 (vertical)-pixel area, but the entire 324-Kb area of the memory can be used to store interim data during processing.

3.2) Sync. Generator for VGA Output

The synchronization generator for VGA output generates VGA-standard synchronization signals.

3.3) RAM DAC

Three 256-byte palette RAMs (red, green, and blue) are used to produce quasi-natural colors and remapping changes.

Number	Transfer Route	Transfer Method
①	ACU_FM › VGA_FM	Concurrent transfer
②	ACU_FM › VGA_FM	Transfer after freezing ACU_FM
③	ACU_FM › MVME162 › VGA_FM	Transfer after freezing ACU_FM
④	VGA_FM › MVME162	VGA Frame Memory Access
⑤	ACU_FM › SIDIP bus › VGA_FM	Concurrent transfer
⑥	ACU_FM › SIDIP bus › VGA_FM	Transfer after freezing ACU_FM

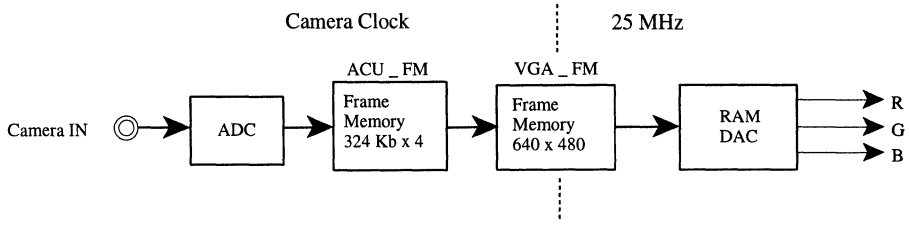


Figure 6: ACU_FM to VGA_FM
(Concurrent transfer)

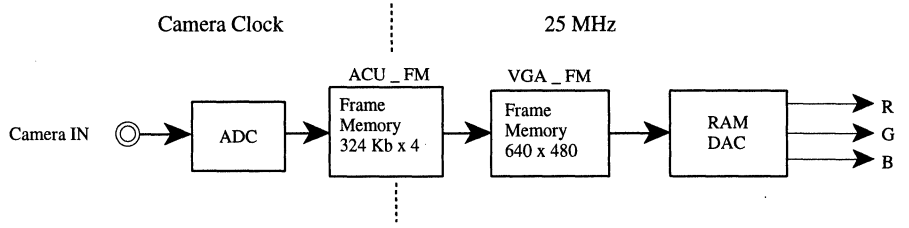


Figure 7: ACU_FM to VGA_FM
(Transfer after freezing ACU_FM)

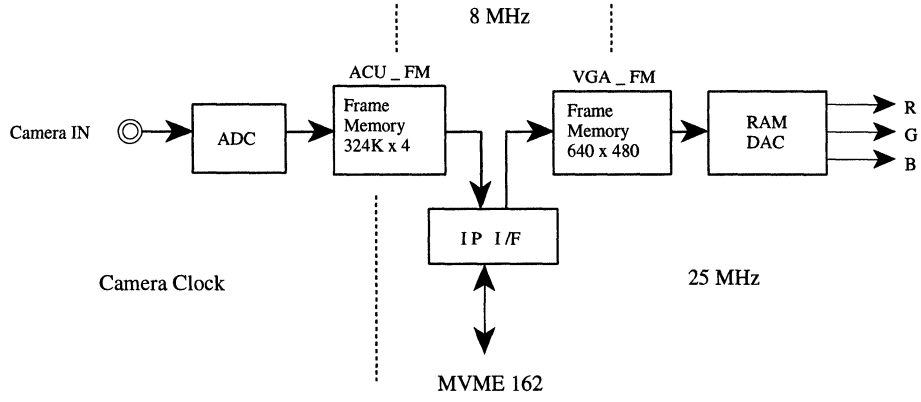


Figure 8: ACU_FM to MVME162 to VGA_FM
(Transfer after freezing ACU_FM)

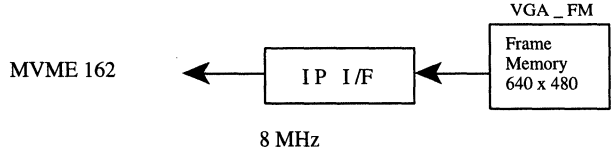


Figure 9: VGA Frame Memory Access, VGA_FM to MVME162

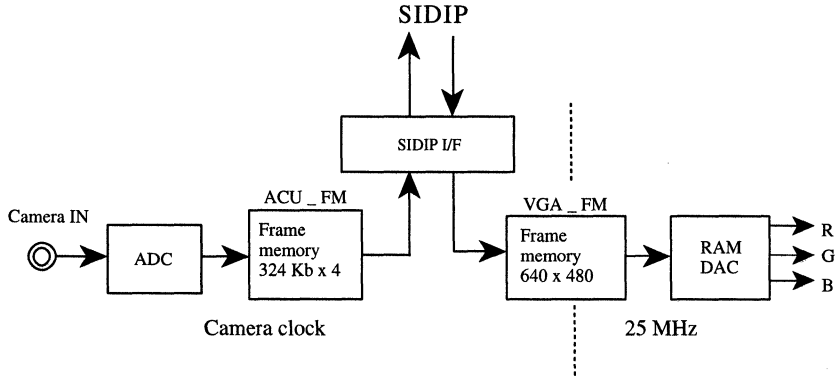


Figure 10: ACU_FM to SIDIP bus to VGA_FM (Concurrent transfer)

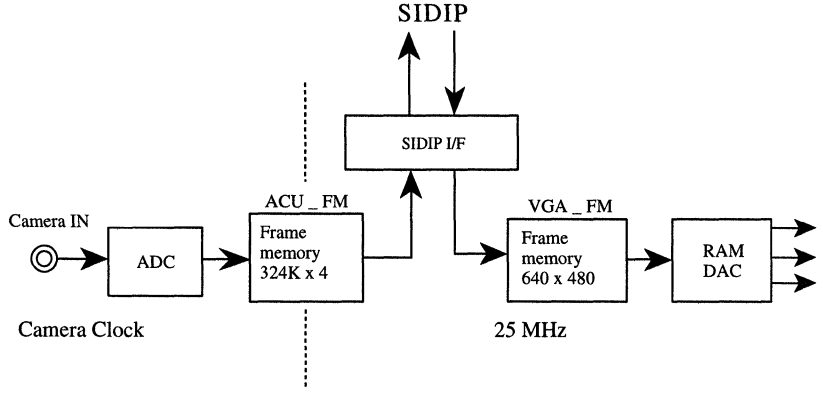


Figure 11: ACU_FM to SIDIP bus to VGA_FM (Transfer after freezing)

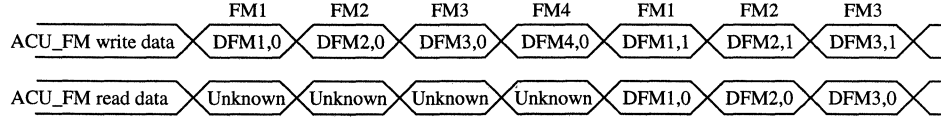


Figure 12: ACU_FM input/output delay (concurrent transfer)

(4) SIDIP EXTENDED I/F CIRCUIT

This circuit is used to connect the image bus of the module baseboard (for example, the IP90BD550) to the IP90MS803. The image bus conforms to the SIDIP standard.

The SIDIP extended interface circuit includes an output image data bus, input image data bus, image clock, and frame synchronization signal outputs (VS, HS). ACU_FM image data is output to the output image data bus, and image data sent through the input image data bus is picked up by the VGA_FM.

OPERATIONS

The ACU_FM's structure is determined by the number of pixels of the line sensor camera. It performs six different operations depending on the method of transfer used from ACU_FM to VGA_FM and the transfer route. The transfer rates vary depending on which operation is run (see the table on page 6).

(1) Input Transfer Modes

1) ACU_FM to VGA_FM (Figure 6)

The transfer rate depends on the camera clock used.

2) ACU_FM to VGA_FM (Figure 7)

Transfer rate: 25 Mb/sec (8 bits).

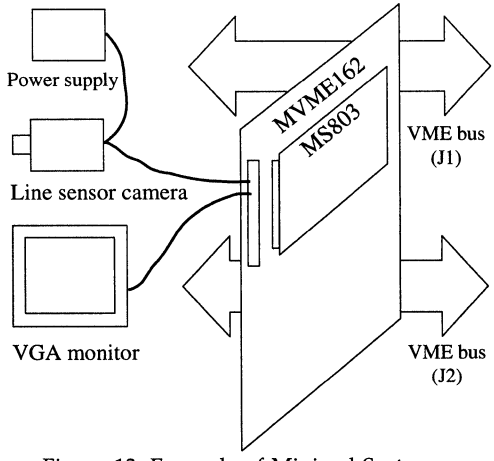


Figure 13: Example of Minimal System Configuration

3) ACU_FM to MVME162 to VGA_FM (Figure 8)

Transfer rate: 8 Mb/sec (16 bits).

Note: Stop the line sensor camera synchronization generator in this mode.

4) VGA_FM to MVME162 (Figure 9)

Transfer rate: 8 Mb/sec (16 bits).

Note: Stop the VGA synchronization generator in this mode.

5) ACU_FM to SIDIP bus to VGA_FM (Figure 10)

The transfer rate depends on the camera clock used.

6) ACU_FM to SIDIP bus to VGA_FM (Figure 11)

Transfer rate: 50 Mb/sec (8 bits x 2).

(2) ACU_FM Input/Output Delay

The IP90MS803 has two transfer methods: one method simultaneously transfers image data as it is input according to the camera clock, and the other transfers data at a high speed after freezing the ACU_FM. When the image to be measured is too large for the frame memory, image processing can still be performed by transferring image data as it is input. However, with this method, the ACU_FM output data is four frames "behind." Figure 12 shows the ACU_FM input/output delay caused by this method.

Note: If the four-frame delay is too long, set the register to perform a simultaneous transfer using a delay of effective pixels x 4 lines.

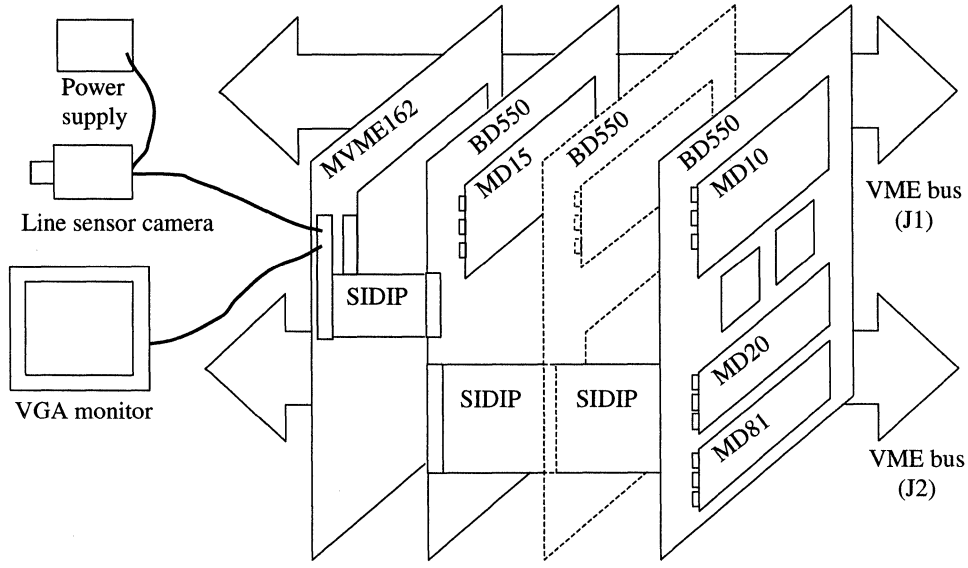


Figure 14: Example of Expanded Configuration

EXAMPLES OF APPLICATIONS

Sumitomo Metals Industries' IP90MS803 Image Frame Grabber can be used in a wide variety of applications.

The IP90MS803 has two principal functions: frame grabber and display. Its hardware processes include image remapping using a LUT. This makes it possible to build (for example) a simple single-board image input/output device using the IP90MS803 and the Motorola MVME162. Figure 13 shows an example of such a system. In this case, the image-processing algorithm is executed by software through the CPU (MC68040) on the MVME162. Because the IP90MS803 is a double-width card, MVME162 can mount a double-width or two single-width IndustryPacks™ (for example, an A/D converter for calculations and parallel interfaces) in addition to the IP90MS803. Thus, a compact image-processing system can be built on a single VME board, eliminating the need for the VME bus system rack and mother board, and reducing costs considerably.

If software alone cannot execute the algorithms fast enough, hardware can be added to increase the system's speed.

The IP90MS803 is installed with Sumitomo Metals Industries' SIDIP image-processing standard interface, also used with other image-processing modules marketed by Sumitomo. The IP90MS803 can be connected through hardware to image-processing modules by combining the VME baseboard IP90BD550 (which uses the SIDIP interface and is the carrier board for VME image processing modules). Thus, slow software processes can easily be replaced with hardware to enhance the system and create a high-speed image-processing device.

Figure 14 shows an example of a system application using an image-processing module. The SIDIP interface (the interface to these system function modules) is an open system, and will accommodate future enhancements from Sumitomo Metals Industries as well as specialized modules or boards developed by users. Third-party enhancements are also expected, increasing the system's flexibility even further.

A minimal configuration consisting of the IP90MS803 and MVME162 would be used in systems not requiring great speed or algorithm checking. Conversely, businesses requiring high-speed systems (such as companies with manufacturing lines or high-speed processing) can extend the useful life of their software and simplify system configuration by adding Sumitomo Metals Industries' image-processing modules to the minimal software system, ultimately reducing costs.

SAMPLE SOFTWARE

A reference printout of a source list of sample software for controlling the IP90MS803 is included in the package. The source list includes these functions:

- Initialization
- Setting the A/D converter block (image input processor block) LUT
- Setting the horizontal and vertical sizes of the input frame memories
- Setting the line sensor camera's clocks
- Setting the VGA display palette
- Setting and starting internal image data transfer
- Freezing the screen
- Reading/clearing the status registers

Note: Be aware that these sample programs are provided as references only. They carry no warranties and may not be supported with follow-up versions, etc. Also, note that the printout and floppy disks are provided. The diskette is in 3–1/2" MS DOS format.

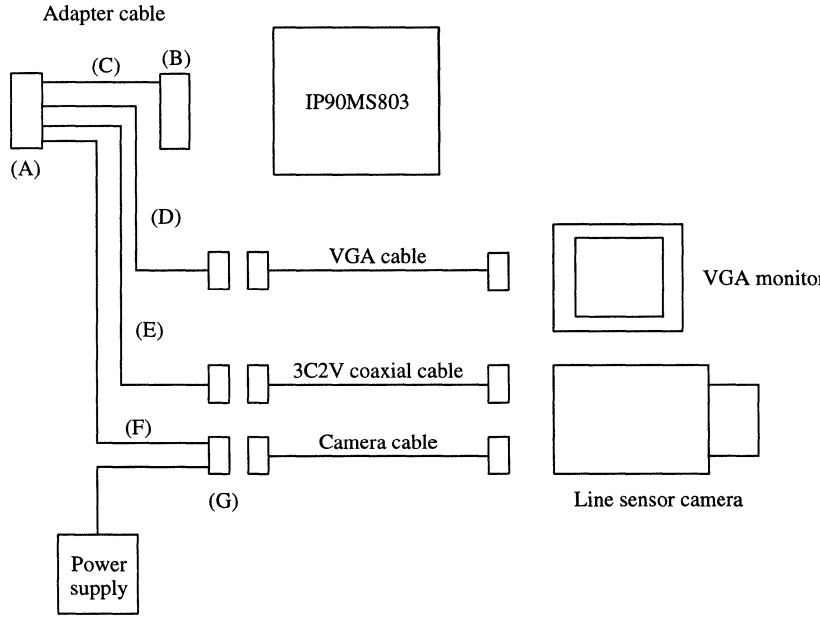


Figure 15: Connecting with External Devices

Block	Pin No.	Signal Name	Function	I/O
SIDIP Extended I/F Block	1	DGND	Ground	—
	2	S_ICLK	Originally oscillated video clock output	O
	3	DGND	Ground	—
	4	E_ICLK2	Video clock 2 input	I
	5	DGND	Ground	—
	6	E_OUT0	Image data output (bit 0)	O
	7	E_OUT1	Image data output (bit 1)	O
	8	E_OUT2	Image data output (bit 2)	O
	9	E_OUT3	Image data output (bit 3)	O
	10	DGND	Ground	—
	11	E_OUT4	Image data output (bit 4)	O
	12	E_OUT5	Image data output (bit 5)	O
	13	E_OUT6	Image data output (bit 6)	O
	14	E_OUT7	Image data output (bit 7)	O
	15	DGND	Ground	—
	16	E_IN0	Image data input (bit 0)	I
	17	E_IN1	Image data input (bit 1)	I
	18	E_IN2	Image data input (bit 2)	I
	19	E_IN3	Image data input (bit 3)	I
	20	DGND	Ground	—
	21	E_IN4	Image data input (bit 4)	I
	22	E_IN5	Image data input (bit 5)	I
	23	E_IN6	Image data input (bit 6)	I
	24	E_IN7	Image data input (bit 7)	I
	25	DGND	Ground	—
	26	S_HS*	Horizontal synchronization signal output	O
	27	DGND	Ground	—
	28	S_VS*	Vertical sync. signal output	O
	29	DGND	Ground	—
	30	N/C	Not used	—
Not Used	31	N/C	Not used	—
	32	DGND	Ground	—
	33	HD*	Horizontal sync. signal output (TTL, negative polarity)	O
	34	VGA_GND	Synchronization signal ground	—
	35	VD*	Vertical sync. signal output (TTL, negative polarity)	O
VGA Signal Output Block	36	VGA_GND	Synchronization signal ground	—
	37	R	R signal output	O
	38	R_GND	R signal ground	—
	39	G	G signal output	O
	40	G_GND	G signal ground	—
	41	B	B signal output	O
	42	B_GND	B signal ground	—
Camera Input Block	43	VDIN	Camera input	I
	44	AG	Camera input ground	—
	45	V_CLK+	Video clock+	O
Camera Drive Block	46	V_CLK-	Video clock-	O
	47	RSG	RS-422 ground	—
	48	V_START+	Video start+	O
	49	V_START-	Video start-	O
	50	RSG	RS-422 ground	—

Table 3: I/O Connector P4 Pins Assignment

PIN CONNECTORS

The IP90MS803 follows the IP standard and has double-width external dimensions, though its functions follow the single-width standard. That is, the two logic I/O connectors consist of P3 and P4; the board has no P1 and P2. P3 and P4 connect to slot B or to slot D of the MVME162 (or the IP Carrier) series. (However, the IP90MS803 cannot connect to slot D of the GreenSpring Computers' VIPC610.)

(1) I/O CONNECTOR P4

P4 is composed of a camera input, a VGA signal output, an extended I/F, and an external trigger input. Its pins are configured as shown in Table 3. P4 has been converted to the 50-pin flat cable plug on the MVME162 (or the IP Carrier). The pin number for each signal is the same for each connector.

(2) EXPLANATION OF SIGNALS

Except for the camera input and camera drive blocks, all signal levels are TTL levels. An asterisk (*) indicates negative logic.

Camera input:

- VDIN: Input signal from the line sensor camera. The display value of the input clock varies according to the length of the BNC cable that connects the IP90MS803 and the line sensor camera. This sets the display value of the input clock according to the cable length.

  Voltage levels of 0 and 5 or 0 and 2.5 V are available.
- AG: VDIN Ground.

Camera Drive:

- V_CLK+, V_CLK-: Camera clocks for the line sensor camera. These signals conform to RS-422 standards.
- V_START+, V_START-: Start pulses for the line sensor camera. These signals conform to RS-422 standards.
- RSG: RS-422 Ground.

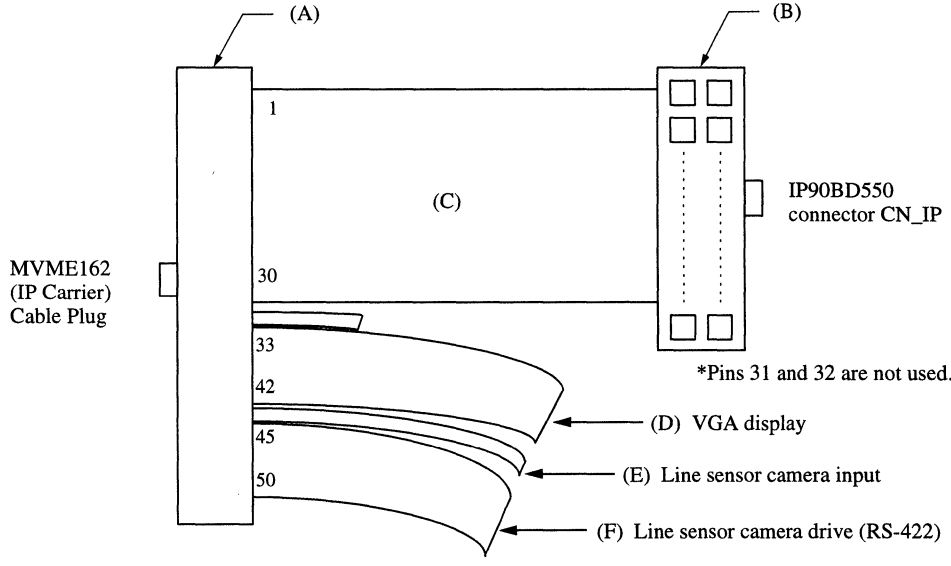


Figure 16: Adapter Cable

VGA Output:

- VGA output has a resolution of 640 x 480 pixels with a horizontal sync of 31.5 KHz and a vertical sync of 60 Hz.

SIDIP Extended Interface:

- It is assumed that the Sumitomo Metals Industries' VME Baseboard IP90BD550 is connected to the SIDIP extended interface.
- The SIDIP extended interface signals connect to the CN-IP connector of the IP90BD550 through a flat cable (connecting pins of the same number together).

(3) ADAPTER CABLE

The input/output signals through the I/O connector P4 pass through the 50-pin flat cable plug on the MVME162 (or IP Carrier) to the external devices, as shown in Figure 15. Specifications for the cable plug and the adapter cable are given on the following page.

Note: The requirements for the adapter cable vary depending on the environment (e.g., external noise). These cable specifications are meant as a guideline only, and should be adjusted as necessary.

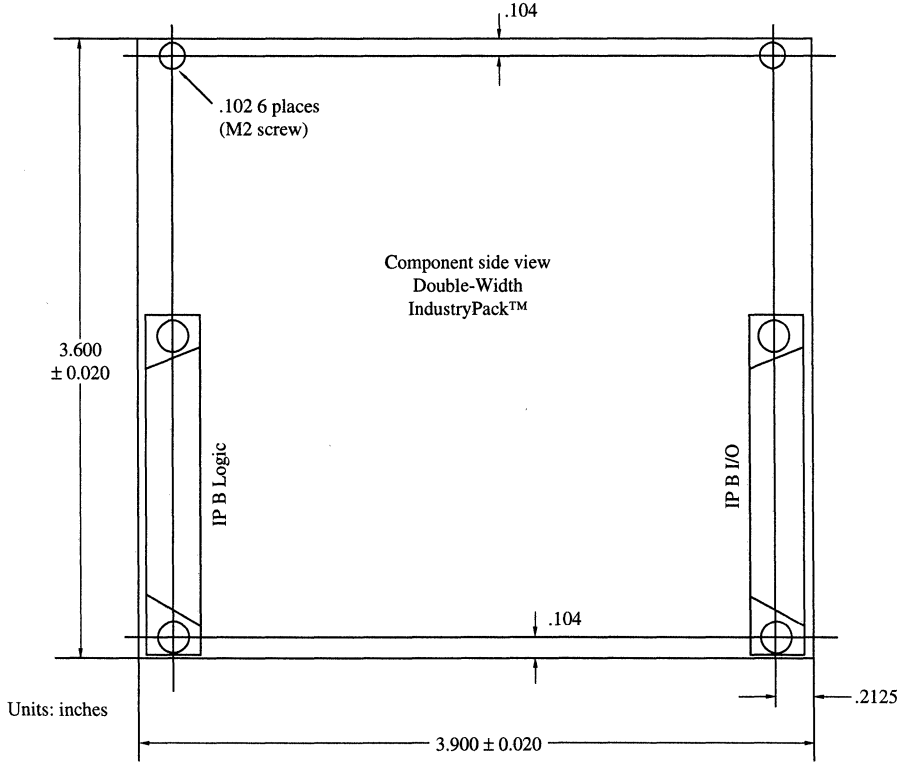


Figure 17: Board Size and Connector Layout

Note 1: The board size is IP double-width.

Note 2: The connector pin layout and dimensions conform to IP standards.

Sumitomo Metals Industries has standard adapter cables available for purchase. Please request a catalog of board options (cables and connectors) from your sales representative.

Use only the cable appropriate to the particular device to connect the adapter cable.

Figure 16 shows the connectors and connector cables to connect to IndustryPack™ standard P4. Connectors and connector cables (A) to (G) in Figures 15 and 16 are explained below.

- Receptacle Connector (A)
  - KEL Co. 6230-050-601
  - Note: This connector extends slightly beyond the parts on the standard VME board. However, if the tapped strain relief (for use in disconnecting the cable) is removed, the assembly falls within the specified dimensions.
- Receptacle Connector (B)
  --KEL Co. 6230-030-601
- Flat Cable (C)
  - AWG#28 -- 30 pins
  - Distance between (A) and (B) must be 70 mm or less.
- Flat Cable or Twisted Flat Cable (D)
  - AWG#28 -- 10 pins
  - Flat cable can be up to 70 mm long, and twisted flat cable can be up to 200 mm. Use a shielded cable for lengths greater than this.
- Flat Cable or Twisted Flat Cable (E)
  - AWG#28 -- 2 pins
  - Flat cable can be up to 50 mm long, and twisted flat cable can be up to 100 mm. Use a 75-ohm coaxial cable for lengths greater than this.
- Flat Cable or Twisted Flat Cable (F)
  - AWG#28 -- 8 pins
  - Flat cable can be up to 50 mm long, and twisted flat cable can be up to 100 mm. Use a standard RS-422 cable for lengths greater than this.

Note: Connecting to camera cable
- The I/O connector does not include a power line to drive the line sensor camera; this must be purchased separately. Connect each signal line of the adapter cable (F) and the cable from the camera drive power with connector (G).

For connector (G) specifications and pin assignments, check the specifications for the line sensor camera and the camera cable.

MECHANICAL DIMENSIONS

Figure 17 shows the size of the board and the layout of the connectors, both of which are in accordance with the IndustryPack™ (IP) standards of GreenSpring Computers in the United States.