Data General

Technical Reference

microNOVA™ COMPUTER SYSTEMS

014-000073-03

NOTICE

Data General Corporation (DGC) has prepared this manual for use by DGC personnel and customers as a guide to the proper installation, operation, and maintenance of DGC's equipment and software. The drawings and specifications contained herein are the property of DGC and shall neither be reproduced in whole or in part without DGC's prior written approval nor be implied to grant any license to make, use, or sell equipment manufactured in accordance herewith.

DGC reserves the right to make changes without notice in the specifications and materials contained herein and shall not be responsible for any damages (including consequential) caused by reliance on the materials presented, including but not limited to typographical or arithmetic errors, company policy and pricing information. The information contained herein on DGC software is summary in nature. More detailed information on DGC software is available in current released publications.

NOVA, SUPERNOVA, ECLIPSE and **NOVADISC** are registered trademarks of Data General Corporation, Southboro, Mass.

Ordering No. 014-000073

Printed in the United States of America
Revision 03, January 1977

TABLE OF CONTENTS

SECTION I
microNOVA COMPUTER SYSTEMS

INTRODUCTION AND OVERVIEW I-1
- CPU and 2K/4K RAM Boards I-1
- Memory Boards I-2
- Peripherals I-2
- General Purpose Boards I-3
- Chassis I-3
- Power Supply I-4

PACKAGED SYSTEMS I-4
CODING AIDS I-6
RELATED MANUALS I-6

SECTION II
CPU AND 2K/4K RAM BOARD

OVERVIEW AND INTRODUCTION II-1
SUMMARY OF CHARACTERISTICS II-2
BLOCK DIAGRAM II-2
- CPU to Memory Bus II-2
- CPU to I/O Bus II-4
- Power Up/Fail and Run Control II-4
- On Board Memory II-4
- Special Function Jumpers II-4
- CPU and Console II-5
- Auto-restart II-5
- Automatic Program Load Option II-5

TABLE OF CONTENTS (Continued)

PROGRAMMING II-5
FIXED POINT ARITHMETIC II-6
LOGICAL OPERATIONS II-6
STACK MANIPULATION II-6
PROGRAM FLOW ALTERATION II-7
INPUT/OUTPUT II-7
CENTRAL PROCESSOR FUNCTIONS II-8
REAL TIME CLOCK II-8

SECTION III
RANDOM ACCESS MEMORY BOARDS

OVERVIEW AND INTRODUCTION III-1
SUMMARY OF CHARACTERISTICS III-1
BLOCK DIAGRAM III-3

SECTION IV
PROGRAMMABLE READ-ONLY MEMORY BOARDS

OVERVIEW AND INTRODUCTION IV-1
SUMMARY OF CHARACTERISTICS IV-1
BLOCK DIAGRAM IV-2

SECTION V
HAND-HELD CONSOLE SUBSYSTEM

OVERVIEW AND INTRODUCTION V-1
SUMMARY OF CHARACTERISTICS V-1

TABLE OF CONTENTS (Continued)

BLOCK DIAGRAM V-3
I/O Interface V-3
HHC Memory V-3
Hand-held Programmer's Console V-3
Display and Control V-3
Keyboard and Control V-4

PROGRAMMING V-4
PROGRAMMING SUMMARY V-4
Instructions V-5
Read Switches V-5
Light Decimal Point V-5
Read Function V-5

SECTION VI
ASYNCHRONOUS INTERFACE BOARD

OVERVIEW AND INTRODUCTION VI-1
SUMMARY OF CHARACTERISTICS VI-1
BLOCK DIAGRAM VI-3
Asynchronous Serial Interface VI-3
Console Debug Option VI-3

PROGRAMMING VI-5
PROGRAMMING SUMMARY VI-5
Instructions VI-6
Read Character VI-6
Write Character VI-6
Read Modem Status VI-6
Control Data Terminal Ready VI-7

TABLE OF CONTENTS (Continued)
SECTION VII
DISKETTE SUBSYSTEM

OVERVIEW AND INTRODUCTION VII-1
SUMMARY OF CHARACTERISTICS VII-1
BLOCK DIAGRAM VII-3
- Subsystem Interface and Controller VII-3
- Data Format VII-3
- 2F Data Transcription VII-3
- Dual Drive Multiplexors VII-3
- Information Flow and Control Sequences VII-3
  - Setup and Status Transfers VII-3
  - Head Positioning VII-3
  - Address Verification VII-4
  - Data Writing VII-4
  - Data Reading VII-4
- Drive Interlocks and Status Flags VII-5

INSTRUCTIONS VII-5
PROGRAMMING SUMMARY VII-5
- Specify Command VII-6
- Load Memory Address Counter VII-6
- Read Current Address VII-6
- Read Status VII-7
- Read Memory Address Register VII-7

PROGRAMMING VII-8
- Phase I: Select the Drive and Determine Status VII-8
- Phase II: Position the Head VII-9
- Phase III: Read or Write VII-11
  - Read VII-11
  - Write VII-13

TIMING VII-13
FORMAT VII-14
ERROR CONDITIONS VII-14
- During Initial Selection VII-14
- During Head Positioning VII-14
- During Reading VII-14
- During Writing VII-15
- Illegal Indications VII-15

TABLE OF CONTENTS (Continued)

SECTION VIII
PROM PROGRAMMER BOARD

OVERVIEW AND INTRODUCTION VIII-1
SUMMARY OF CHARACTERISTICS VIII-1
BLOCK DIAGRAM VIII-2
PROGRAMMING VIII-5
PROGRAMMING SUMMARY VIII-5
- Instructions VIII-6
- Specify Address VIII-6
- Loading Programming Buffer VIII-6
- Verify VIII-6

SECTION IX
GENERAL PURPOSE INTERFACE BOARDS

OVERVIEW AND INTRODUCTION IX-1
SUMMARY OF CHARACTERISTICS IX-1
BLOCK DIAGRAM IX-3
PROGRAMMING IX-3
- Instructions IX-3
- No I/O Transfer IX-3
- Data In A IX-3
- Data In B IX-4
- Data In C IX-4
- Data Out A IX-4
- Data Out B IX-4
- Data Out C IX-4
- I/O Skip IX-4
- I/O Reset IX-4
- Mask Out IX-4
- Interrupt Acknowledge IX-4

TABLE OF CONTENTS (Continued)

SUMMARY OF GPIO BUS SIGNALS IX-9
Data IX-9
Programmed I/O IX-9
Program Interrupt IX-9
Data Channel IX-10
System Control IX-10
Interface Timing IX-10
Programmed Transfers IX-10
Data Channel Transfers IX-11
Jumpers IX-12
Data Lines and Drive Capability IX-13
Busy/Done IX-13
Interface Wire Wrap Pins IX-13

SECTION X
MISCELLANEOUS PRINTED CIRCUIT BOARDS

OVERVIEW AND INTRODUCTION X-1

SECTION XI
microNOVA CHASSIS

OVERVIEW AND INTRODUCTION XI-1
SUMMARY OF CHARACTERISTICS XI-1
CHASSIS XI-2
CARDFRAME XI-3
POWER SUPPLY XI-5
Power Supply with Battery Backup XI-6

TABLE OF CONTENTS (Continued)

SECTION XII
POWER SUPPLY ASSEMBLIES

OVERVIEW AND INTRODUCTION XII-1
SUMMARY OF CHARACTERISTICS XII-2
BLOCK DIAGRAM XII-2
- AC to DC Conversion XII-2
- Reference Voltage Generation XII-2
- +5V Regulation XII-2
- +15V Regulation XII-4
- -5V Regulation XII-4
- Power-fail Detection XII-4
- Battery Backup Implementation XII-4

APPENDICES

I/O BUS CONSIDERATIONS A-1

microNOVA - NOVA CONSIDERATIONS B-1

This page intentionally left blank.

SECTION I
microNOVA
COMPUTER SYSTEMS

INTRODUCTION AND OVERVIEW

The microNOVA computer systems are organized around the microNOVA microprocessor, a general-purpose, user-programmable processor which resides on a single LSI N-channel MOS chip. The microNOVA processor has a word length of 16 bits, four accumulators for arithmetic and logical operations, and the capability to address up to 32,768 words of dynamic RAM and/or programmable ROM (PROM) memory. Two independent buses are used to communicate with memory and with I/O devices. The memory bus is used to both access and refresh memory, and consists of sixteen bidirectional data lines and four control lines. The I/O bus is capable of driving I/O devices as far as 100 feet from the microNOVA processor, and consists of a 2-bit bidirectional differential data bus, timing lines, and control lines. The microNOVA computer is software-compatible with Data General's NOVA line of computers, and is similar to the NOVA 3 instruction set (see Appendix for greater detail).

The microNOVA computer is available as a complete computer system, as assemblies such as the CPU and 2K/4K RAM board, and as chips such as the CPU and the I/O Controller (IOC). The availability of the separate assemblies and chips gives the customer the flexibility either to construct a computer system tailor-made to his particular applications from the parts available or to purchase the complete microNOVA computer system packaged by Data General. The assemblies which can be purchased separately or combined to form a complete microNOVA computer system are as follows:

CPU and 2K/4K RAM Board
4K/8K MOS RAM Board
Programmable Read-Only Memory Board
Hand-held Console Subsystem
Asynchronous Interface Board
Diskette Subsystem
PROM Programmer Board
General Purpose Interface Board
General Purpose Wiring Board
Extender Board
9-slot/18-slot Main Chassis
Card Frame Assembly
Power Supply Assemblies

Each of the boards in the system is a 7.5 inch wide printed circuit board suitable for installation in the microNOVA main chassis or the microNOVA card frame assembly. The boards vary in length from 9.9 to 11.7 inches.

On the following page a microNOVA functional block diagram is displayed to show the System user how the various modules are implemented together.

CPU and 2K/4K RAM Boards

A microNOVA CPU and 2K/4K RAM board is a single 7.5 by 10.4 inch printed circuit board containing a microNOVA CPU, 2K or 4K words of dynamic N-channel MOS random access memory, a timing generator, and interface circuitry for the memory bus and the I/O bus. The word length for both the CPU and the memory is sixteen bits. The CPU contains four accumulators, a program counter, a stack pointer, a frame pointer, and a real-time clock. A general-purpose instruction set, which includes multiply and divide functions as standard features, is available for programming the microNOVA processor.

The memory buffering and control circuitry contained on the CPU and 2K/4K RAM board can support up to 32K words of memory within a microNOVA computer system. Memory refresh operations for dynamic RAM are totally hidden, since the memory is refreshed in parallel with processor operations during cycles in which memory is neither read nor written.

The I/O capabilities supported by the CPU and 2K/4K RAM board are programmed data transfer, 16-level programmed priority interrupts, and a data channel capability. The I/O bus can support up to 30 I/O devices, and can be up to 100 feet in length.

Edge-mounted controls and indicator lights are available as an option with the CPU and 2K/4K RAM board. The controls are start, halt, continue, and lock, and the indicator lights are run, AC power, and battery power. These controls and lights provide a means for controlling the operation of the CPU when the board is not part of a complete microNOVA computer system.

A power fail/auto-restart capability is standard with the CPU and 2K/4K RAM board. An automatic program load capability is optional. The starting address and the device code for automatic program load procedures used are determined by jumper settings on the board. Two features are available to provide user flexibility in start and program load procedures: 1) For program start capability, memory location 77777_8 contains the contents of the Jumper Word Register to specify the starting address. 2) For the automatic program load (APL) option, 64 words of local ROM can be enabled to override memory locations 77700- 77777_8 , while associated memory location 77776_8 contains the contents of the Jumper Word Register with the option device code.

Memory Boards

The microNOVA 4K/8K MOS RAM board contains either 4K or 8K 16-bit words of dynamic random access memory and associated logic. The memory has a cycle time of 960 nanoseconds and must be refreshed once every 1.8432 milliseconds. Each refresh operation will refresh a single block of 64 words. The block of addresses associated with the memory is selected with jumpers.

The programmable read-only memory (PROM) boards contain sockets for 512, 1024, 2048, or 4096 16-bit words of programmable read-only memory and associated logic. The 512 word and 1024 word PROM boards contain eight sockets and 16 sockets respectively for 1024-bit (256 x 4) PROM chips. The 2048 word and 4096 word PROM boards contain 32 sockets for either 1024-bit (256 x 4) or 2048-bit (512 x 4) PROM chips. The block of addresses associated with the memory is selected with jumpers.

Within a microNOVA computer system, RAM and PROM can be mixed in any combination up to 32K words of memory.

Peripherals

The hand-held console subsystem consists of a hand-held programmer's console, an I/O controller on a 7.5 by 9.9 inch printed circuit board, and a 10-foot 16-conductor ribbon cable which connects the console to the controller. The hand-held console is calculator-like in appearance with 20 push-button keys and a 6-digit display area. The I/O controller contains 256 words of ROM and RAM, corresponding to and taking precedence over any other RAM associated with addresses 77400-77777_8. The controller memory contains standard software which enables the hand-held console to be used to examine and modify the contents of memory locations and internal CPU registers, and to control the operation of the processor with functions such as start, stop, and reset. The standard software also includes an automatic program load capability. User software can be used in place of the standard software to redefine the functions associated with the console keys.

The asynchronous interface board consists of a controller for a single asynchronous terminal or communications line and, optionally, a console debugger. The controller performs full-duplex character assembly and disassembly, with the input and output signals conforming to either EIA RS232-C (CCITT V-24) or 20mA current loop specifications. Any standard transmission speed from 50 baud to 19,200 baud is available. The transmission speed, the number of data bits and stop bits per character, the type of parity associated with each character, and the line specifications (EIA RS232-C or 20mA current loop) are selected with jumpers. In addition, the controller includes a modem control capability.

The console debug option available with the asynchronous interface board allows any ASCII terminal to be used as a system debugging console. With this option, the terminal can be used to examine and modify the contents of memory locations and internal CPU registers, to control the operation of the processor with functions such as start and continue, and to set and clear breakpoints. An automatic program load capability is also included with the console debug option. The console debug option program resides in 256 words of ROM and RAM on the asynchronous interface board. This ROM and RAM correspond to and take precedence over any other RAM associated with addresses 77400-77777_8. Since both the console debug option and the hand-held console subsystem contain ROM associated with addresses 77400-77777_8, they are mutually exclusive options within a microNOVA computer system.

The diskette subsystem provides removable, direct access, moving head disc memory for the microNOVA computer systems. The subsystem consists of one or two drive units in a rack-mountable chassis, a controller on a 15" square printed circuit board contained within the drive unit chassis, and an external I/O bus cable for connection to the I/O bus of a microNOVA computer. Each diskette can store up to 157,696 16-bit words in blocks of 256 words. The maximum data transfer rate is 15,625 words per second.

The PROM programmer board provides the capability to program and verify a complete PROM board, one word at a time, under program control. The PROM programmer board has two female board connectors along one 7.5 inch edge of the board for connection to any of the microNOVA PROM boards. The memory on a PROM board is programmed by inserting PROM chips in the sockets of the PROM board, inserting the PROM board in the PROM programmer board connector, and issuing the appropriate I/O instructions to program and verify the contents of memory.

General Purpose Boards

The general purpose interface board provides the basic logic required by a customer building an I/O device controller for inclusion in a microNOVA computer system. The board contains an IOC, an I/O bus transceiver, and supporting logic for a generalized I/O interface including programmed I/O, program interrupts, and a data channel capability. In addition, the board has drilled holes and etched conductors to accommodate chips in 14, 16, 18, 20, 22, 24, 28, and 40-pin packages. The maximum number of chips that can be added to a general purpose interface board is 32 14-pin chips. The general purpose interface board is also available with sockets and wire wrap pins to accommodate a combination of up to 27 14-pin and 16-pin packages.

The general purpose wiring board is a 7.5 by 10.4 inch printed circuit board containing only drilled holes and etched conductors to wire wrap pins. The board is designed to be used by the customer building his own assembly for inclusion in a microNOVA computer system. The board can accommodate up to 50 chips in 14-pin packages, and can accommodate a lesser number of chips in combinations of 14, 16, 18, 20, 22, 24, 28, and 40-pin packages.

The extender board provides the capability to gain direct access to the components of another printed circuit board in a microNOVA computer system for debugging and maintenance purposes. The board has two female board connectors along one 7.5 inch edge of the board for connection to any microNOVA computer system board. The extender board is inserted into the chassis slot of the board to be tested, and the board to be tested is plugged into the extender board. Etched conductors on the extender board carry all the back panel connections to the extended printed circuit board.

Chassis

The microNOVA main chassis is a 19" rack-mountable chassis available in both 9-slot and 18-slot configurations. The 9-slot chassis contains a 9-slot card frame and a power supply mounted side-by-side across the front of the chassis. The battery backup option, if present, is mounted on a support behind the power supply. The 18-slot chassis contains two 9-slot card frames mounted side-by-side across the front of the chassis, and a power supply mounted behind the left card frame. The battery backup option, if present, is mounted behind the right card frame. Both chassis have a front panel which swings aside to provide front access to the printed circuit boards. The front panel contains a power-on keylock switch, a rocker switch, three indicator lights, and a place to attach the hand-held console. I/O device cables connect directly to the I/O controller printed circuit boards at the rear of the card frames. Strain relief is provided for the I/O device cables at the rear of the chassis.

The microNOVA card frame assembly contains 9 slots for 7.5 inch wide printed circuit boards. The assembly consists of a stamped metal frame with stiffener and card guides, an etched backpanel, and female board connectors to the backpanel. The backpanel carries the memory bus signals and I/O bus signals to the boards mounted in the card frame. The memory bus signals are enabled only when a CPU and 2K/4K RAM board is mounted in the bottom slot of the card frame. The I/O bus signals can be chained from the CPU card frame to other card frames with an external I/O bus cable. Each slot has a metal retainer for attaching an I/O device cable connector. The card frame includes a DC power cable for connection to the power supply.

Power Supply

The power supply provides 18 amps at +5 volts dc, 2.5 amps at +15 volts dc, and 1.2 amps at -5 volts dc. This is sufficient power to support a fully-loaded 18-slot chassis containing standard microNOVA computer system boards. Low voltage switching regulators control the +5 volt and +15 volt portions of the power supply by independent pulse width modulation, and control the -5 volt portion of the power supply by variable pulse rate generation. An AC line filter, independent overvoltage and overcurrent sensing, and a power-fail monitor provide system protection. Any of the standard line voltages of 100 volts, 120 volts, 220 volts, or 240 volts in the frequency range of 47-63Hz may be used to drive the power supply; only the AC line cord is different for the different line voltages. The arrangement of the pins within the AC line cord/power supply connector selects the correct transformer taps for the line voltage being supplied.

A battery backup option is available with the power supply to provide sufficient power to keep the CPU and up to 32K (using three 8K RAM's) of dynamic random-access memories refreshed for approximately 30 minutes during a power failure. Normal system operation is not intended. However, since the fans do not operate on battery, time is limited by temperature rise. At normal room temperature (25 C), the system is typically refreshed for approximately 20 minutes before the thermal circuit shuts down the system.

The battery backup option consists of two 12-volt battery packs connected in series, battery switchover circuitry, and battery recharge circuitry.

PACKAGED SYSTEMS

A microNOVA computer system can be purchased in any one of several packaged system configurations. The CPU and 4K RAM board can be purchased with either a 9-slot or 18-slot chassis. In addition, a complete microNOVA development system can be purchased, where the development system includes a CPU, 16K RAM, an 18-slot chassis with the battery backup option, a hand-held console subsystem, a dual diskette subsystem, a box of 10 diskettes, an asynchronous interface board, either an ASR33 or KSR35 Teletype®, a 72" equipment cabinet with filler panels, and all applicable system cabling.

The microNOVA development system is fully supported by Data General software. This software includes the diskette-based Disc Operating System (DOS), FORTRAN IV with real-time extensions and library routines, an editor, a macro assembler, a relocatable loader, and a symbolic debugger.

Teletype® is a registered trademark of Teletype Corporation, Skokie, Illinois. All references to teletypes in this manual shall apply to this mark.


DATA GENERAL

CODING AIDS

In the descriptions of the separate instructions, the general form in which the instruction is coded in assembly language is given along with the instruction format and the description of the instruction. The general form in which an instruction may be coded has the following format:

MNEMONIC *[optional mnemonics]* **OPERAND STRING**

The mnemonic must be coded exactly as shown in the instruction description. Some instructions have optional mnemonics that may be appended to the main mnemonic if the option is desired. The operand string is made up of the operands for the given instruction.

Square brackets "[ ]" or *"[ ]"* along with boldface- and italic-printed symbols are used in this manual to aid in defining the instructions. These conventions are used to help describe how an instruction should be written so that it can be recognized by the assembler and translated into the correct binary, or machine language, representation. Their general definition is given below.

[ ], *[ ]* Square brackets indicate that the enclosed symbol is an optional operand or mnemonic. The operand enclosed in the brackets (e.g., *[,skip]*) may be coded or not, depending on whether or not the associated option is desired.

BOLD Operands or mnemonics printed in boldface must be coded exactly as shown. For example, the mnemonic for the MOVE instruction is coded **MOV** .

italic Operands or mnemonics printed in italics require a specific substitution. Replace the symbol with the number of a desired accumulator, or address, or with a user-defined symbol that the assembler recognizes as a specific name, address, number, or mnemonic.

The following abbreviations are used throughout this manual:

AC = Accumulator

ACS = Source Accumulator

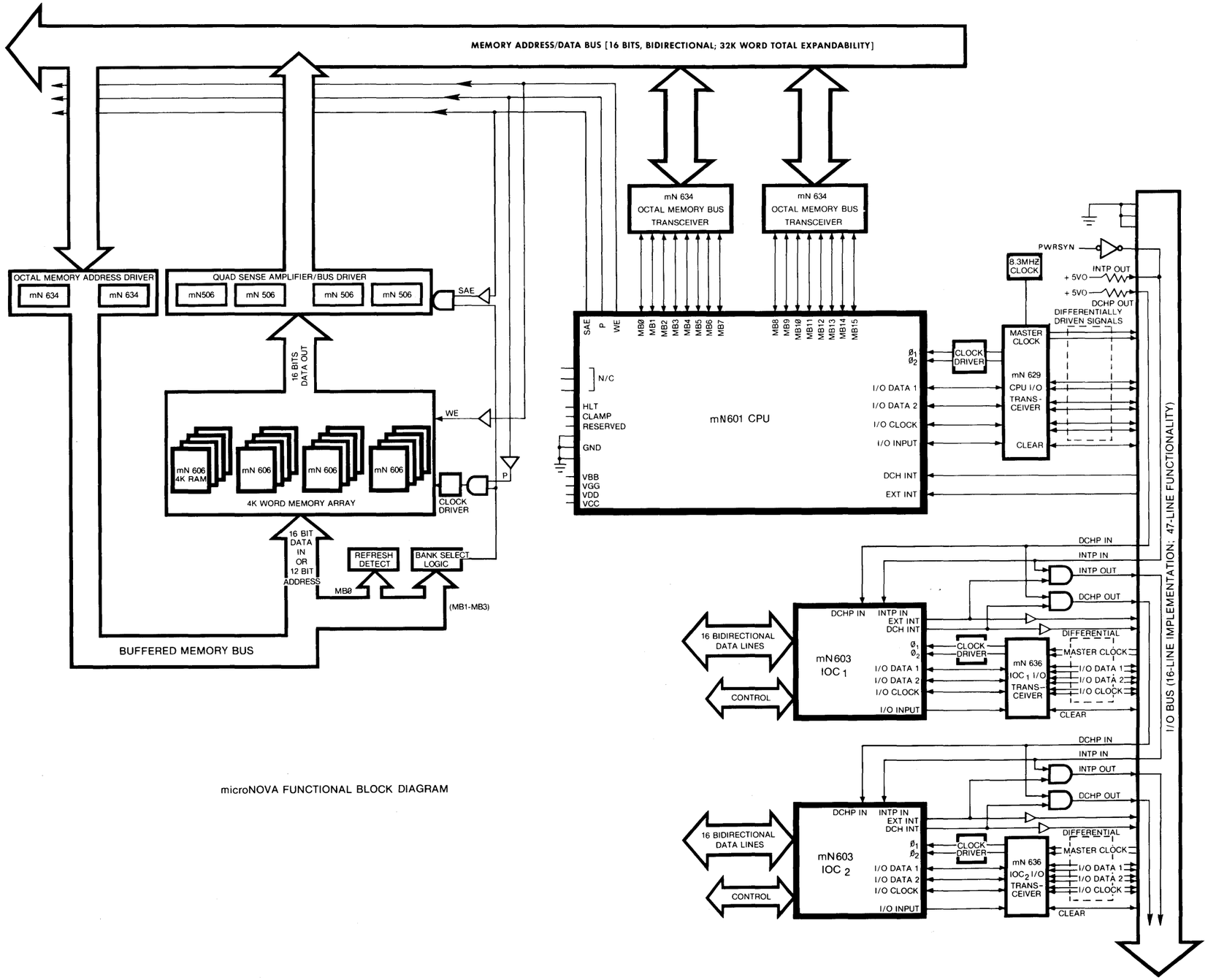
ACD = Destination Accumulator

F = Flag Control Function

When describing the format or a word involved in an information transfer between the computer and an I/O device, the various fields and bits in the word are labeled with names descriptive of their functions. Bits in the word which are not used by the I/O device controller are shaded. Shaded bits are ignored on output and are undefined on input.

RELATED MANUALS

Two other manuals describe portions of the microNOVA computer family. The Technical Reference for microNOVA Integrated Circuits *(DGC no. 014-000074)* describes the major integrated circuits which form the basis of a microNOVA computer system, including detailed timing diagrams and electrical specifications for each integrated circuit. The Programmer's Reference Manual for microNOVA Computers *(DGC no. 015-000050)* describes the instruction set of the microNOVA computers.



microNOVA FUNCTIONAL BLOCK DIAGRAM

Rev 03

SECTION II
CPU AND 2K/4K RAM BOARD

The microNOVA CPU board is an LSI minicomputer on a single 7.5 by 10.4 inch printed circuit board; it contains a complete, Data General NOVA-line, 16-bit word length, central processor in a single 40-pin package, and includes a real-time clock, Power Fail/Auto-restart, plus provision for automatic program load and on board console options. The board contains all support and interface circuits needed to connect the processor to separate, external, I/O and memory buses and to a turnkey console. It also contains 2,048 or 4,096 16-bit words of 960ns read-write memory (RAM). By itself, the CPU board can be integrated into a variety of special applications; combined with other microNOVA system pieces, it is the basis of a complete microprocessor-based computer system.

OVERVIEW AND INTRODUCTION

The quality and power of a general purpose, microprocessor-based, computer system depends to a large measure on: *i)* the flexibility of its instruction set and the power of its central processing unit (CPU); *ii)* the availability of utility and development software; *iii)* the type and structure of the information paths into and out of the CPU; and *iv)* the size and power needs of the pieces that make up the system.

Data General's microNOVA CPU board executes the proven NOVA (NOVA 3) instruction set. It is a powerful, flexible, instruction set that combines high speed, multiple operation, fixed point, logical operations with hardware multiply/divide, hardware PUSH/POP stack, and a tightly optimized I/O handling capability. Most importantly, it is an instruction set for which Data General has already developed extensive software. This means that users can develop and optimize new applications software on any NOVA line computer system with all the resources available in proven Data General operating systems, compilers, run-time libraries, and peripherals.

The microNOVA CPU board exchanges information with memory and peripherals over two independent information transfer paths: the memory bus and the I/O bus. The CPU board contains a transceiver and control buffers for the memory bus; a 16-bit wide, parallel, synchronous, TTL-level bus by which the central processor communicates with microNOVA memory boards, or with special customer-designed memories. The board also contains the drivers and control buffers for a 2-bit wide, differential, serial I/O bus by which the central processor communicates with microNOVA peripherals or customer-designed I/O interfaces containing the microNOVA IOC.

The microNOVA CPU is packaged as a complete minicomputer on a single 7.5 by 10.4 inch printed circuit board. Built around a single-chip microprocessor that is designed and fabricated by Data General's semiconductor division, the CPU board is compact, and it easily lends itself to special applications when used alone, or when it is combined with other pieces in the growing microNOVA family of computers.

SUMMARY OF CHARACTERISTICS

CENTRAL PROCESSOR		
Operations:	Instruction Execution	
	Fixed Point Arithmetic	
	Logical Operations	
	Stack Manipulation	
	Program Flow Alteration	
	Input/Output	
	Interrupts	
	Power Fail/Auto-restart	
	Real-time Clock	
	Stack Overflow	
	External	
	Data Channel Transfer	
	Input: 7.2 microseconds/16-bit word	
	Output: 5.8 microseconds/16-bit word	
	Automatic Program Load (optional)	
	Memory Refresh (hidden)	
Registers:	4 16-bit accumulators	
	1 15-bit stack pointer	
	1 15-bit frame pointer	
	1 15-bit program counter	
	1 interrupt enable flag	
	1 real-time clock enable flag	
	1 carry bit	
	1 stack overflow flag	
Buses:	Memory	
	16 address/data lines	
	4 control signals	
	I/O	
	2-bit serial, synchronous, bidirectional	
	I/O clock	
	Master clock	
	2 request lines	
	2 priority lines	
	1 clear line	
	Maximum total length: 100 ft. (30.5m)	
MEMORY		
Memory Type:	Dynamic MOS N-channel RAM.	
Board Capacity:	2,048 or 4,096 16-bit words.	
Cycle Time:	Read	960ns.
	Write	960ns.
	Read/Modify/Write	1920ns.
	Refresh	960ns.
Maximum Refresh Period:	1.8432 milliseconds.	
Board Dimensions:	7.5 x 10.4 in. (19 x 26 cm.)	
Maximum Operating Temperature:	131°F (55°C)	

BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components plus the data and control paths of a microNOVA CPU board. This board contains a microNOVA microprocessor (CPU), 2,048 or 4,096 words of dynamic read-write memory (RAM), and power fail and restart controls. It has connections to the memory and I/O buses and front panel. The board may also contain the automatic program load option.

The CPU board contains two principal data paths and six functional blocks. These are:

CPU to memory bus
CPU to I/O bus
Power up/fail and run control
On board memory
Special function jumpers
CPU and front panel controls
Auto-restart
Automatic program load option

The first five items in the above list encompass the basic structure of the CPU board. The front panel controls, auto-restart, and the program load option take effect by gaining, on an exceptional basis, overriding control over that basic structure.

CPU to Memory Bus

Communication between the processor and memory take place along a memory bus. The memory bus operates at TTL levels; it is a wired OR, half duplex, 16-bit wide, address and data path. The operations which take place on it occur synchronously, under CPU direction. Such operations include Read memory, Write memory, Read-Modify-Write memory, and Refresh memory. All information that moves on the memory bus does so in bit-parallel fashion, 16-bits at a time.

A 16-bit memory transceiver and the memory control buffer interface the CPU integrated circuit to the memory bus. Four memory control signals direct the flow of information along the 16-bit bi-directional address and data path.

The CPU's memory interface supports the flow of information between the central processor's memory port and the microNOVA memory bus. The protocols by which the CPU communicates with memory are described in the Technical Reference for microNOVA Integrated Circuits *(DGC no. 14-000074)*.

microNOVA CPU AND 2K/4K RAM BOARD

* Mem Cntrl Signals

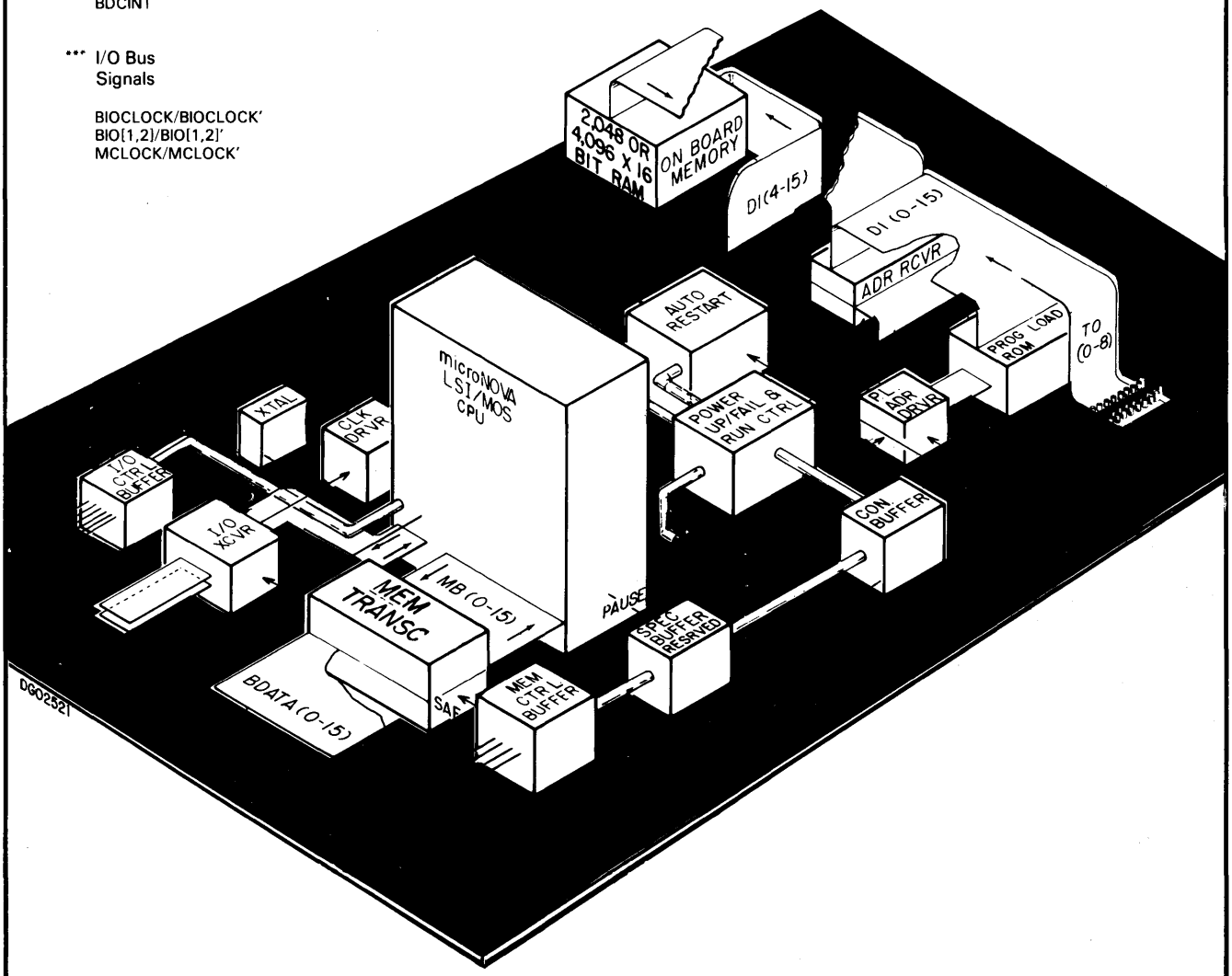
BSAE
BWE
BP
PHIL

** I/O Cntrl Signals

CLEAR'
DCHP
INTP
BEXTINT
BDCINT

*** I/O Bus Signals

BIOCLOCK/BIOCLOCK'
BIO[1,2]/BIO[1,2]'
MCLOCK/MCLOCK'



CPU to I/O Bus

Communication between the processor and peripherals take place along the I/O bus. The microNOVA I/O bus is a bidirectional, 2-bit, synchronous data path from the CPU to any device that needs direct program control or data channel service. It consists of 9 signals, four of them differential. All information is placed on and read from two differential lines in conjunction with an I/O clock signal carried on a third differential line; the remaining differential lines carry the system clock. The single-ended control lines include a reset signal, plus priority and request lines for program interrupt and data channel service. Information moves on the bus in bit-serial fashion, 2 bits at a time, with a two-bit wide coded header and 16 bits of data. The I/O bus is carried up the chassis backpanel and may extend a total of 100 feet from the CPU using 16-wire ribbon cable.

A CPU I/O transceiver, consisting of 4 differential I/O bus ports, includes 3 bidirectional lines (2 data, 1 I/O clock) and 1 system clock driver. A single-ended clear line completes the transceiver I/O interface.The master clock for a microNOVA computer system is derived from the crystal oscillator connected to the CPU's I/O transceiver.

The CPU's I/O interface supports the flow of information between the central processor's I/O port and the microNOVA I/O bus. The protocols by which the CPU communicates with peripherals are described in the technical reference cited above.

Power Up/Fail and Run Control

The CPU board includes control circuits to direct the CPU's operation when power is first applied to the microprocessor, while it its running, and as power drops. These circuits include the a power up/fail and run control, and a CPU clock driver.

The power up monitor holds the CPU in a well defined state and resets the computer during a cold startup. It then allows the processor to start, when directed to do so, by the console controls or self-starting options. The power up requirements of the microNOVA CPU are described in the technical reference cited above. The power fail monitor senses the power beginning to fail and alerts the processor, via a power fail interrupt, that there is a guaranteed 2 milliseconds of operation before power actually fails.

The CPU clock driver amplifies the system master clock which is derived from a crystal oscillator connected to the I/O bus transceiver. It provides precise, non-overlapping clock signals required to run the processor.

On Board Memory

The CPU board holds an entire read-write memory module which connects directly to the microNOVA memory bus. The memory consists of one bank of dynamic RAM with 2,048 or 4,096 16-bit words, 16-bit wide data in and data out paths, a 12-bit address path, four memory control lines, and a 16-bit memory transceiver. Memory address jumpers assign the 4K memory address boundary in which the on board memory will operate within the total system. Because of hardware restrictions, it is not recommended that the on board memory be assigned the highest 4K position.

The construction of the on board memory is similar to that of other RAM modules for the microNOVA systems. The flow of information within the module and also between it and the memory bus is described in section III.

Special Function Jumpers

The board utilizes a Jumper Word Register to configure two types of automatic operation. They are:

1. Starting Address/Auto-Restart
2. Automatic Program Load (APL)

Inserting jumper W4 enables the Jumper Word Register.

Inserting a jumper in W5 or W6 controls the function of the Jumper Word Register. If a jumper is inserted in W5, the Jumper Word Register becomes a device code for the Automatic Program Load option. If a jumper is inserted in W6, the Jumper Word Register becomes a starting address used by the start functions and Auto-Restart. W5 and W6 should not be both inserted at any time.

Inserting jumper W7 disables Auto-Restart after a power failure if no battery power is present (i.e., battery backup not provided or batteries are discharged). This jumper is normally inserted if battery backup is not installed, or a restart program in PROM is not utilized.

Inserting jumper W8 enables 64 words of ROM associated with addresses $077700\text{-}077777_8$ for the APL option.

Refer to sheet II-10 for chart configurations of these jumpers.

CPU and Console

The CPU board includes a control circuit and signal buffer for the switches and indicators of the microNOVA turnkey console; the board may optionally have a group of switches and indicator lights for console functions.

Signals from the console switches and indicators pass through the console buffer to the power up/fail and run monitor, to the program load address driver, and to the auto-restart monitor to start and continue functions of the board.

Auto-restart

Auto-restart includes a circuit that monitors the conditions of a backup battery, the position of the LOCK switch of the turnkey console, and the configuration of the setup jumpers. When power returns after a power failure, this circuit can initiate an automatic restart by gaining control over selected I/O and memory control lines.

Automatic Program Load Option

When the automatic program load option is installed, it includes a 16-bit information path through the CPU memory transceiver to the memory bus, control lines to the console buffer and the power up/fail and run control, a 64 word ROM, and a 6-bit address driver. When an automatic program load is initiated, the option gains control over selected I/O and memory control lines. The CPU executes a series of instructions stored in the program load ROM, and transfers the microNOVA bootstrap program from there to memory locations $2\text{-}37_8$.

PROGRAMMING

The microNOVA CPU processes information one word at a time, where a word is 16 bits in length. The CPU has four accumulators which are also 16 bits in length and are used for arithmetic and logical operations. Furthermore, two of the accumulators can be used as index registers. Memory can be addressed either directly or by using indirect addressing. Chains of indirect addresses may be up to eight addresses long.

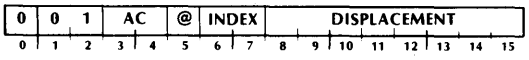
The instruction set for the CPU contains instructions that perform fixed point arithmetic and logical operations between accumulators, transfer of operands between accumulators and memory, transfer of program control, stack manipulations, I/O operations, and processor control operations. All instructions are one word in length. The instruction set of the microNOVA computer is similar to the instruction set of the NOVA line of computers (see Appendix for greater detail).

The table on the following pages summarizes the instruction set of the microNOVA computer. For a more detailed description of the instructions, consult the Programmer's Reference Manual for microNOVA Computers *(DGC no. 015-000050)*.

FIXED POINT ARITHMETIC

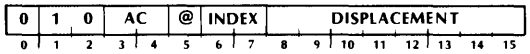
Load Accumulator

LDA *ac,[@]displacement[,index]*



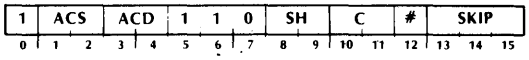
Store Accumulator

STA *ac,[@]displacement[,index]*



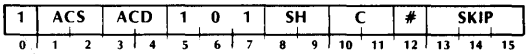
Add

ADD *[c][sh][#] acs,acd[,skip]*



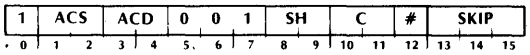
Subtract

SUB *[c][sh][#] acs,acd[,skip]*



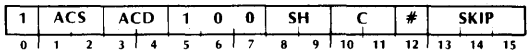
Negate

NEG *[c][sh][#] acs,acd[,skip]*



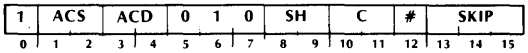
Add Complement

ADC *[c][sh][#] acs,acd[,skip]*



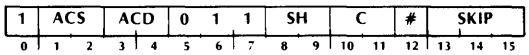
Move

MOV *[c][sh][#] acs,acd[,skip]*



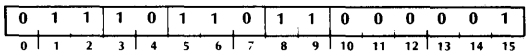
Increment

INC *[c][sh][#] acs,acd[,skip]*



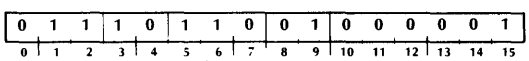
Multiply

MUL



Divide

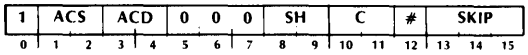
DIV



LOGICAL OPERATIONS

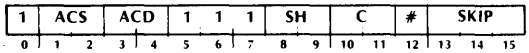
Complement

COM *[c][sh][#] acs,acd[,skip]*



And

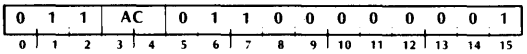
AND *[c][sh][#] acs,acd[,skip]*



STACK MANIPULATION

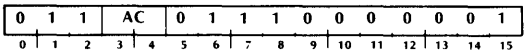
Push Accumulator

PSHA *ac*



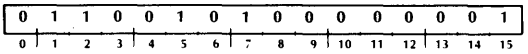
Pop Accumulator

POPA *ac*



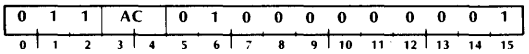
Save

SAV



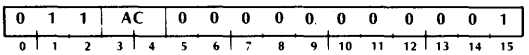
Move To Stack Pointer

MTSP *ac*



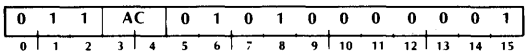
Move To Frame Pointer

MTFP *ac*



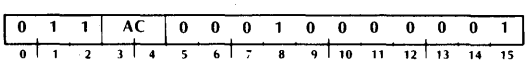
Move From Stack Pointer

MFSP *ac*



Move From Frame Pointer

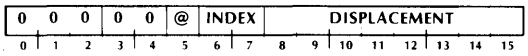
MFFP *ac*



PROGRAM FLOW ALTERATION

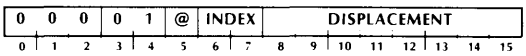
Jump

JMP *[@]displacement[,index]*



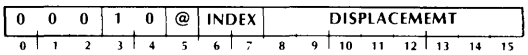
Jump To Subroutine

JSR *[@]displacement[,index]*



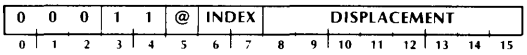
Increment And Skip If Zero

ISZ *[@]displacement[,index]*



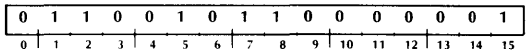
Decrement And Skip If Zero

DSZ *[@]displacement[,index]*



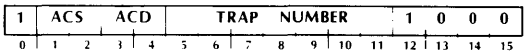
Return

RET



Trap

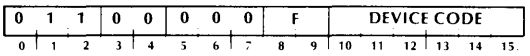
TRAP *acs,acd,trap number*



INPUT/OUTPUT

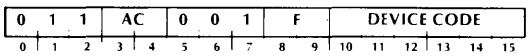
No I/O Transfer

NIO *[f] device*



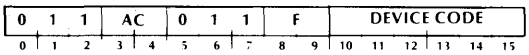
Data In A

DIA *[f] ac,device*



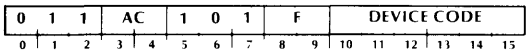
Data In B

DIB *[f] ac,device*



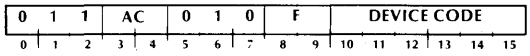
Data In C

DIC *[f] ac,device*



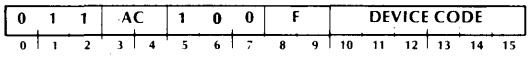
Data Out A

DOA *[f] ac,device*



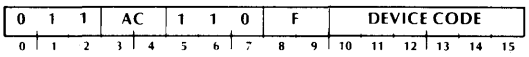
Data Out B

DOB *[f] ac,device*



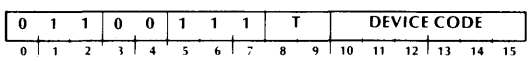
Data Out C

DOC *[f] ac,device*



I/O Skip

SKP *[t] device*

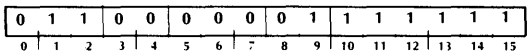


CENTRAL PROCESSOR FUNCTIONS

Interrupt Enable

INTEN

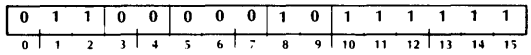
NIOS CPU



Interrupt Disable

INTDS

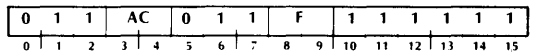
NIOC CPU



Interrupt Acknowledge

INTA *ac*

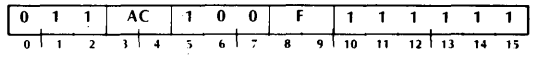
DIB *[f]* *ac*, **CPU**



Mask Out

MSKO *ac*

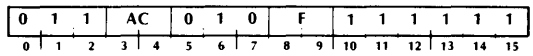
DOB *[f]* *ac*, **CPU**



Reset

IORST

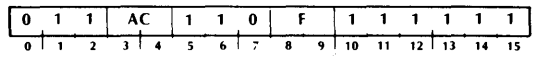
DOA *[f]* **0,CPU**



Halt

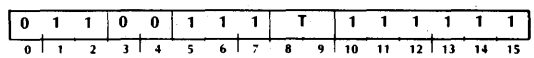
HALT

DOC 0,CPU



CPU Skip

SKP *[t]* **CPU**

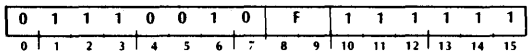


REAL TIME CLOCK

Real-time Clock Enable

RTCEN

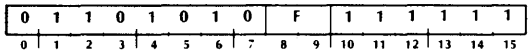
DOA *[f]* **2,CPU**



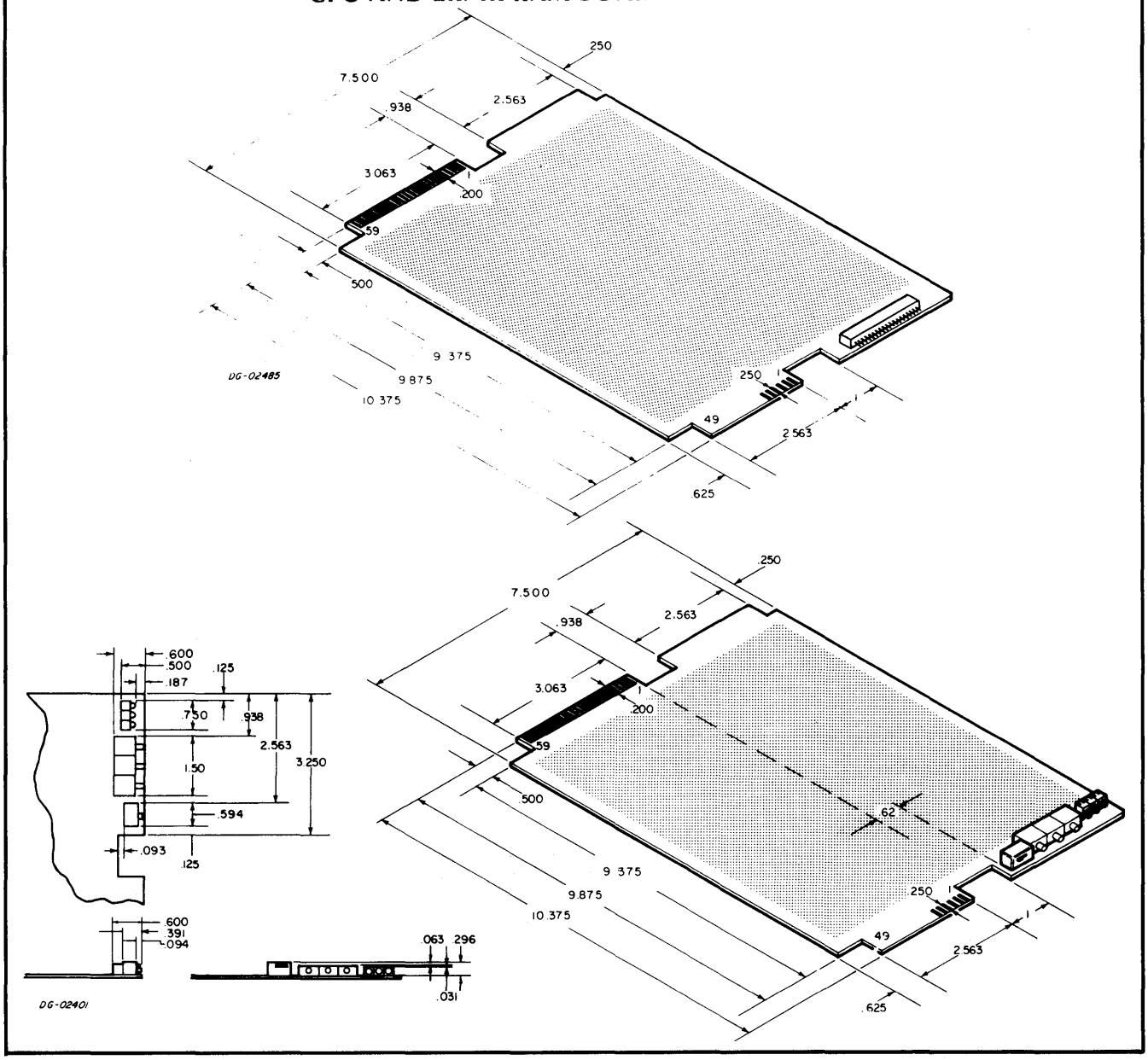
Real-time Clock Disable

RTCDS

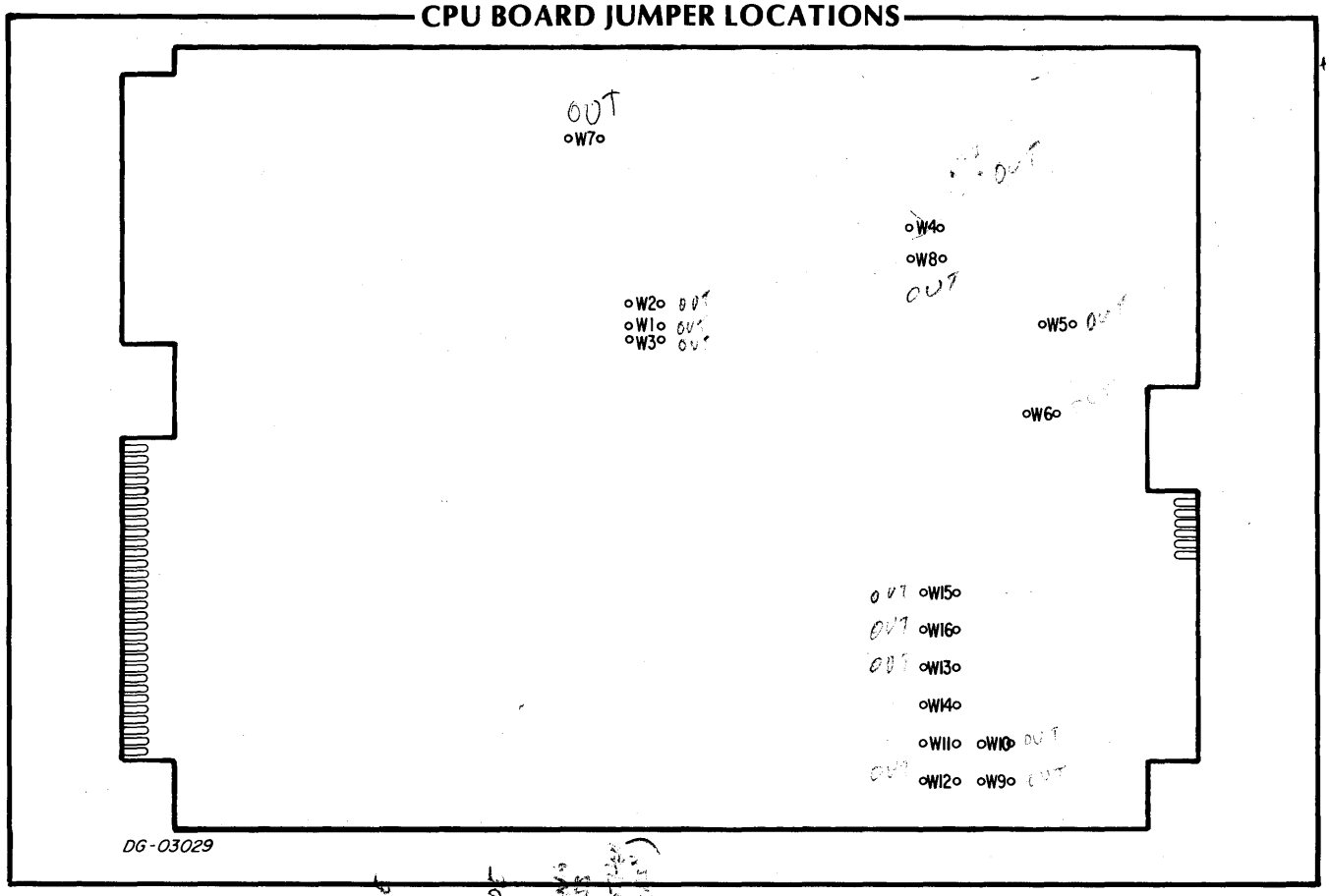
DOA *[f]* **1,CPU**



CPU AND 2K/4K RAM BOARD DIMENSIONS



CPU BOARD JUMPER LOCATIONS



JUMPER WORD REGISTER

JUMPER NUMBER	W4	W5	W6	W8	W9	W10	W11	W12	W13	W14	W15	W16
APL OPTION	IN	IN	OUT	IN	IN FOR DCH DEV	--	DS0	DS1	DS2	DS3	DS4	DS5
STARTING ADDRESS	IN	OUT	IN	OUT	A0	A1	A2	A3	A4	A5	A6	A7
HHC OR CONSOLE DEBUG OPTION	OUT	--	--	--	--	--	--	--	--	--	--	--

DG-03395

NOTE:
1. DS[0-5] AND A[0-7] INSERT JUMPER TO SPECIFY 1.
2. A[8-15] = 00 000 001.

JUMPER WORD ADDRESS

ADDRESS	INSERT JUMPER
077776_8	W5
077777_8	W6

If W5 is inserted, memory address 077776_8 shall contain the contents of the Jumper Word Register with the device address of the APL option.

If W6 is inserted, memory address 077777_8 shall contain the contents of the Jumper Word Register with the starting address of Auto-Restart.

Jumpers W1-W3 determine which 4K address group out of 32K is resident on the CPU board. Because of hardware restrictions, it is not recommended that on board memory be assigned the highest 4K position.

ADDRESS SELECTION JUMPERS

BIT POSITIONS OF STARTING ADDRESS	1	2	3
INSERT JUMPER TO SPECIFY 1	W1	W2	W3

OTHER JUMPERS

JUMPER	FUNCTION
W4	Insert jumper to enable jumper word register
W7	Insert jumper to disable auto-restart after a power failure if power is not supplied by battery back-up.
W8	Insert jumper to enable 64 words of local ROM associated with addresses 077700-077777_8. Used for APL.

SECTION III
RANDOM ACCESS MEMORY BOARDS

Data General's model 8572 and model 8573 random access memory (RAM) boards contain either 4K or 8K of dynamic MOS read/write memory and the circuitry to interface that memory to the microNOVA memory bus. Model 8572 contains 4,096 16-bit words and model 8573 contains 8,192 16-bit words of RAM. These RAM modules perform read, write, and refresh operations in 960ns. A read-modify-write operation can be performed in 1920ns.

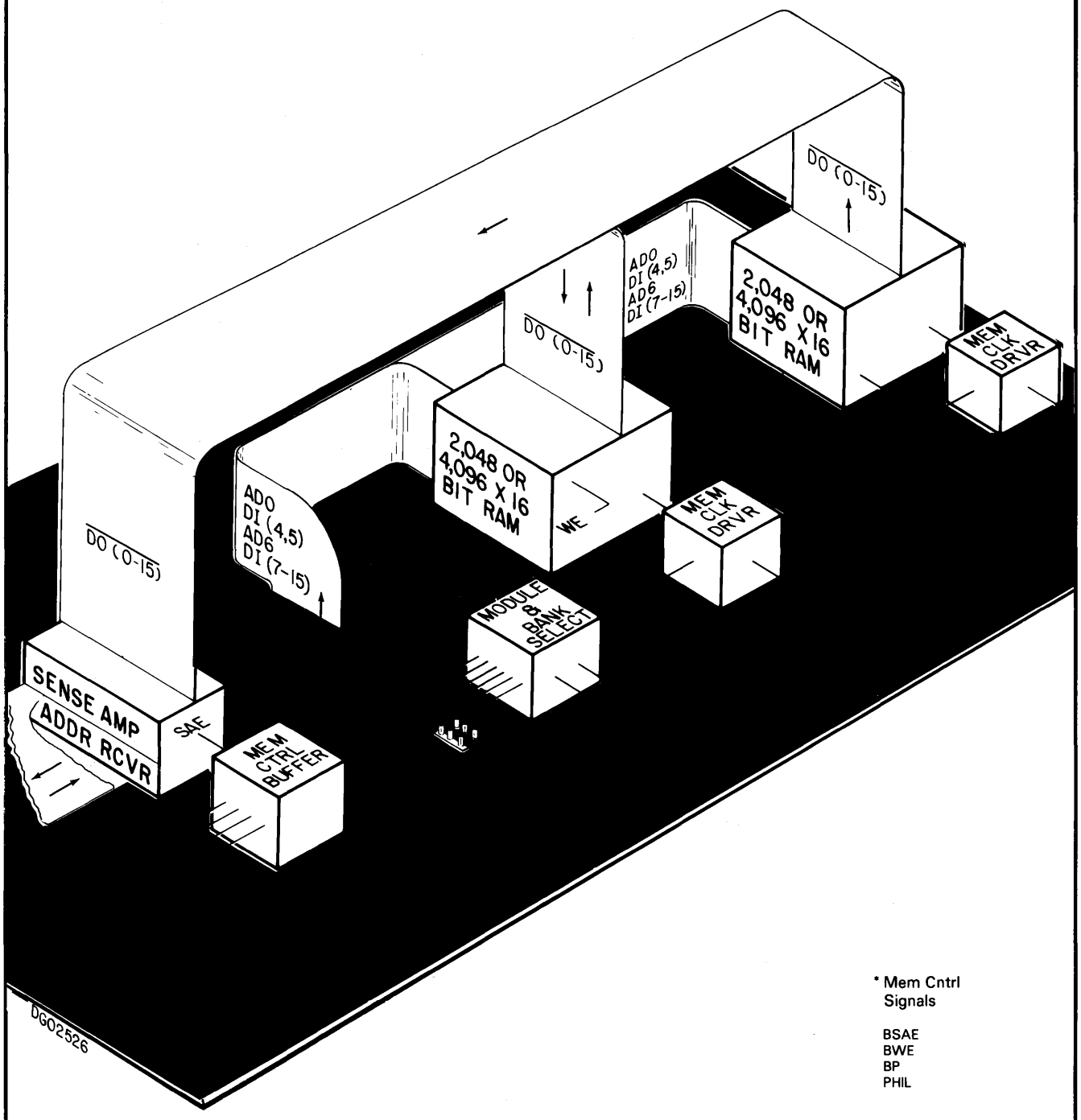
OVERVIEW AND INTRODUCTION

Random access memory (RAM) is a local storage medium whose contents can be read and modified, one word at a time. Dynamic MOS RAM is random access memory that has been constructed using metal-oxide semiconductor (MOS) technology. Dynamic MOS RAM must be refreshed at frequent intervals in order for its contents to be maintained.

The microNOVA 4K/8K MOS RAM board contains 4K or 8K 16-bit words of dynamic MOS random access memory. A microNOVA computer system can access up to 32K words of memory; all 32K words of memory may be RAM.

SUMMARY OF CHARACTERISTICS

Memory Type	Dynamic MOS N-channel RAM.	
Board Capacity	4,096 or 8,192 16-bit words.	
Cycle Time	Read	960ns.
	Write	960ns.
	Read/Modify/Write	1920ns.
	Refresh	960ns.
Maximum Refresh Period	1.8432 milliseconds.	
Location Requirement	Within Slots 2-6	
Board Dimensions	7.5 x 9.9 in. (19 x 24.9 cm.)	
Maximum Operating Temperature	131°F (55 C)	



* Mem Cntrl Signals

BSAE
BWE
BP
PHIL

BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components plus the control and data paths of microNOVA RAM boards. The board is organized into two banks of dynamic RAM memory, each with 2,048 or 4,096 16-bit memory locations. The board contains clock drivers for each memory bank, an address selection/configuration network, a memory bus sense amplifier/bus driver, a control buffer, and address data paths for moving information within the board.

The memory address receivers and sense amplifier/bus drivers and memory control buffer connect the RAM board to the microNOVA memory bus. The protocols by which the microNOVA CPU communicates with memory are detailed in the Technical Reference for microNOVA Integrated Circuits *(DGC no. 014-000074)*.

When an address appears on the memory bus at the start of a memory operation, the board and bank selection network enables the board if the address on the bus falls within the range of addresses assigned to the board by the placement of its starting address jumpers. The network enables the clock driver in the appropriate memory bank, and the twelve low-order bits of the address are driven onto the RAM address lines. The action taken next by the board depends on the type of memory operation to be performed.

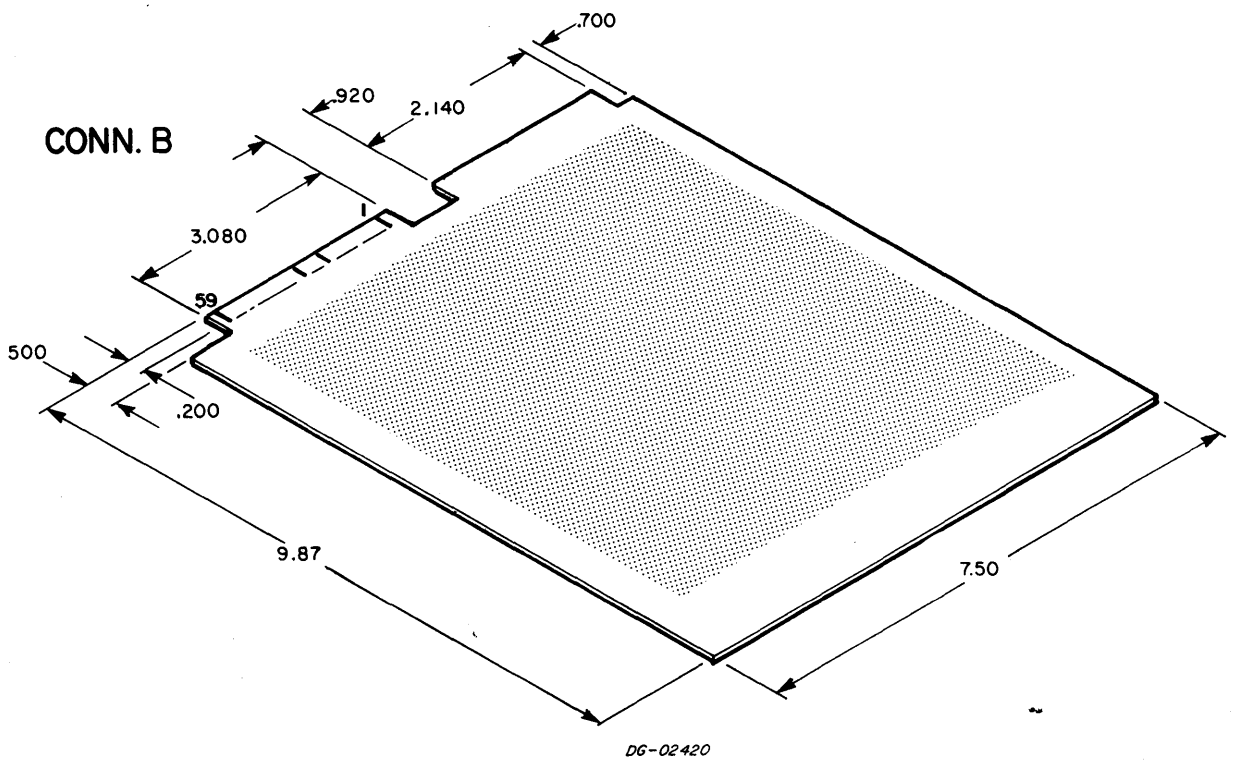
If the operation is a memory read, the memory control signals drive the contents of the addressed memory location onto the board's data out lines to the memory sense amplifier. If the RAM disable signal **PHIL** has not been asserted by a ROM board assigned to the same address, the memory bus driver places the 16-bit memory word on the memory bus.

If the operation is a memory write, the memory control signals drive the 16-bit word to be written in memory onto the the board's data in **(DI)** path and load it into the addressed memory location.

If the operation is a memory read-modify-write, the memory control signals first drive the contents of the addressed memory location onto the data out **(DO)** path and onto the bus if no ROM is assigned to the same address. The location read remains active until the new data for that address arrives on the bus, and moves onto the boards's data in path; then, the memory control signals load the new data into the memory location just read.

If the operation is a memory refresh, the memory control signals initiate a refresh operation simultaneously on 1/64 th of the memory; no data moves on either the data input or output paths.

RAM MEMORY BOARD DIMENSIONS



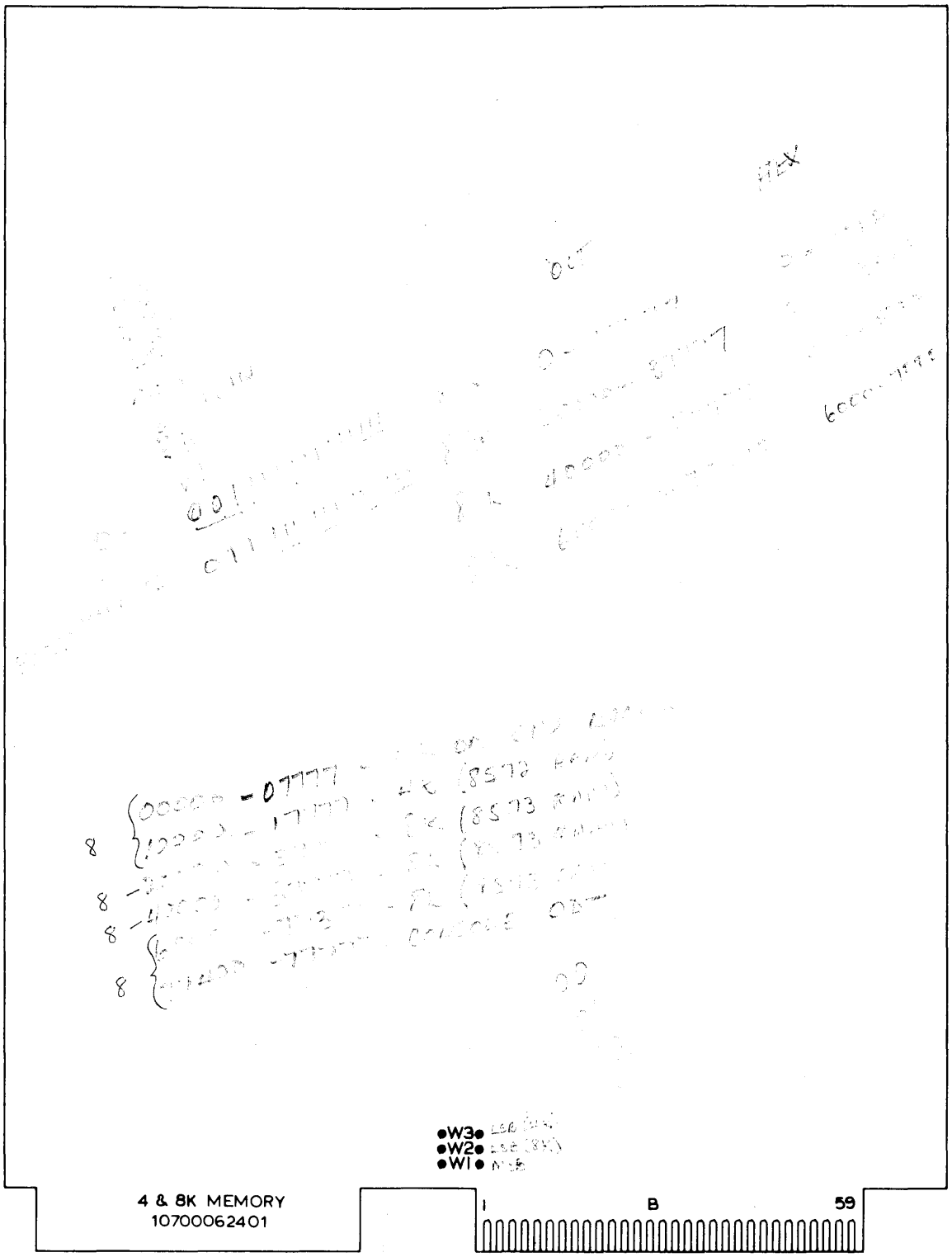
ADDRESS SELECTION JUMPERS

BIT POSITIONS OF STARTING ADDRESS	1	2	3
INSERT JUMPER TO SPECIFY 1	W1	W2	W3*

*Jumper W3 is ignored on the 8K RAM board.

The Address Selection Jumpers control the group selection of 4K/8K RAM memories on each board. System maximum is 32K. Jumper W1 is the MSB and W3 is the LSB for implementation of the address selection. For 8K group selection, jumper W3 is ignored.

4K MEMORY BOARD JUMPER LAYOUT



This page intentionally left blank.

SECTION IV
PROGRAMMABLE READ-ONLY MEMORY BOARDS

Data General's microNOVA read-only memories are programmable, 200ns, semiconductor, PROM boards containing from 512 to 4096 16-bit words of directly accessible memory. Any amount of PROM, up to the maximum system memory size of 32K words, may be placed in a microNOVA system; board starting addresses can be assigned to any 2K boundary in memory. PROM may overlay any alterable (RAM) memory locations, where a RAM/ROM priority line arbitrates memory conflicts in favor of ROM. ROM/ROM conflicts (e.g. 4K PROM at the top of memory conflicts with both the hand-held console and the console debug option) can be avoided by programming all ones in the conflicting locations in one of the memories.

OVERVIEW AND INTRODUCTION

Certain programs and data contained in a computer system are never changed, but are used frequently enough to justify their being permanently located within the address space of the CPU. Read-only memory (ROM) provides a medium on which such information can be stored. The contents of a ROM location cannot be changed under program control while the ROM is in the address space of the CPU. However, since the contents of ROM cannot be changed, the information contained in ROM is protected from software and hardware malfunctions.

Some ROM is field programmable; that is, the contents of the ROM are initially all zeros or all ones, and the appropriate ones or zeros can be "programmed" or "burned" into the ROM after the memory is purchased. Such programmable ROM is called PROM.

The microNOVA programmable read-only memory boards contain sockets for 512 to 4096 16-bit words of PROM available in 256 by 4 bit or 512 by 4 bit packages. The contents of the memory on a microNOVA PROM board can be programmed, one word at a time, using the microNOVA PROM programmer board.

SUMMARY OF CHARACTERISTICS

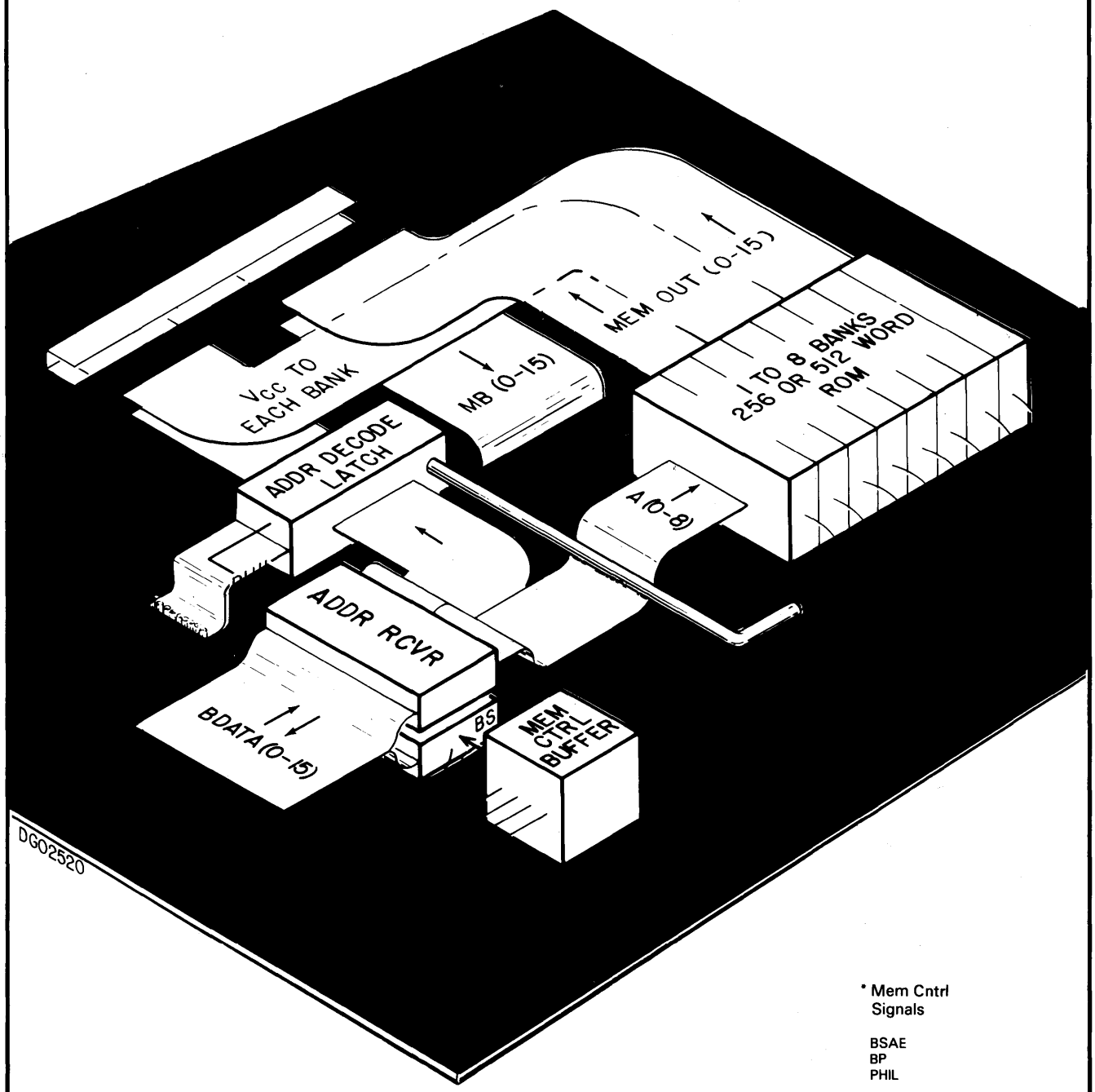
PROM Type:	Signetics 82S126 or 82S130
Board Capacity:	512; 1,024; 2,048; or 4,096 16-bit words.
Access Time:	200ns.
Board Dimensions:	7.5 x 9.9 in. (19 x 24.9 cm.)
Maximum Operating Temperature:	131°F (55°C)

BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components plus the control and data paths of microNOVA ROM boards. The board contains a memory bus address receiver and latch, from 1 to 8 banks of PROM, each with 256 or 512 16-bit words, a memory driver, and a memory control buffer. For normal PROM operation, there is also a shorting block located on connector "A".

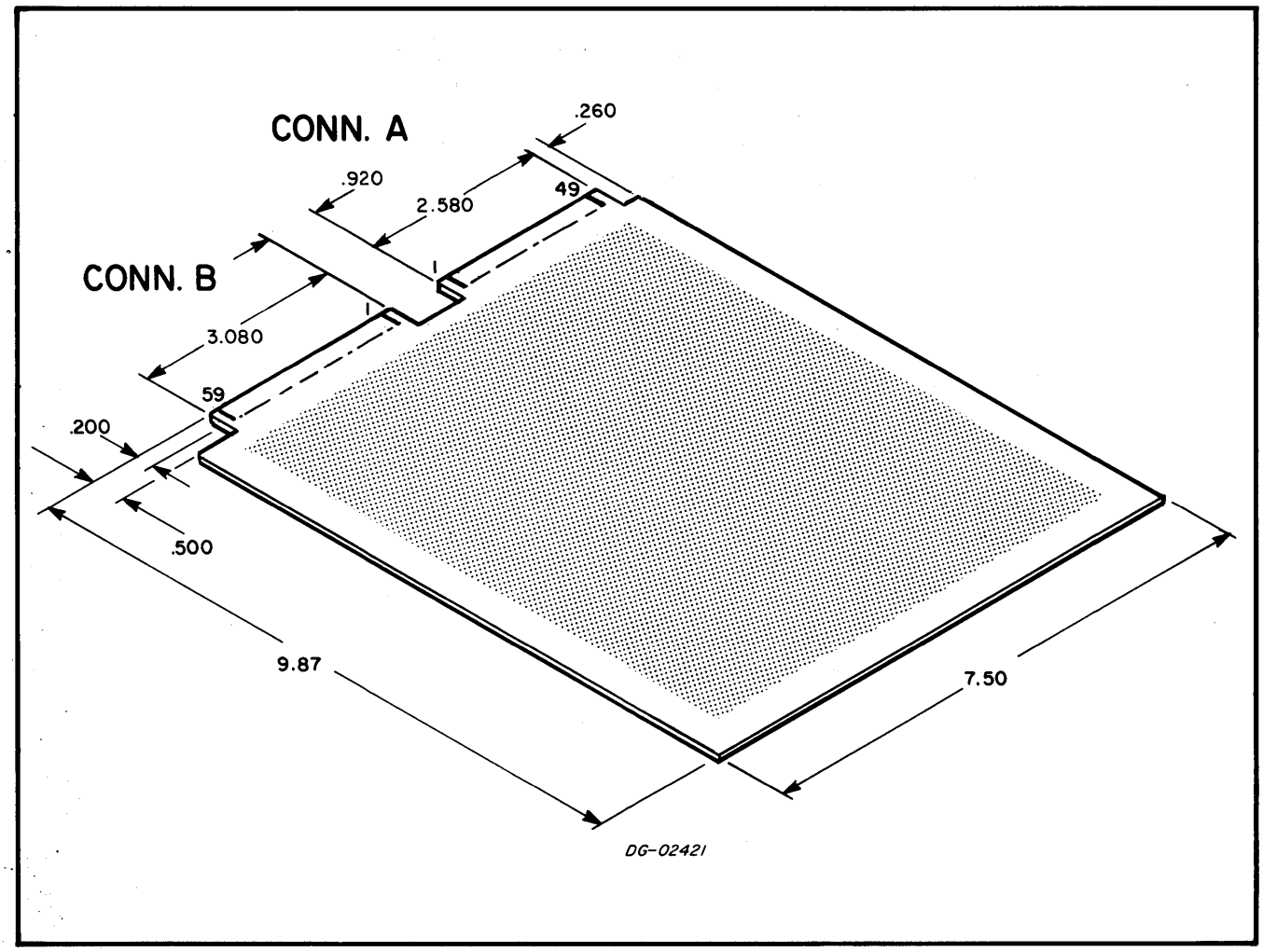
When an address appears on the memory bus at the start of a Read operation, the address latches in the address receiver. The high-order bits select one of the eight banks by applying power (V_{CC}) through the shorting block to that bank, while the low-order bits of the address pass directly to all PROM banks. (To reduce dc power dissipation, the PROM banks remain unpowered when not being read). The addressed word is driven onto the memory bus while the memory control buffer asserts the signal **PHIL** to disable any RAM in the system from placing data on the bus.

The PROM board is designed so that PROMS can be programmed on the board using the microNOVA PROM programmer. (The programmer plugs directly into the microNOVA chassis, and the PROM board plugs directly into the programmer.) Two groups of signals, the bank select signals and taps from the data out lines of the PROM banks, are carried to connector "A" because they are used by the microNOVA PROM programmer when programming PROMS. For the PROM board's normal use, a shorting plug must be installed on the "A" connector to close the internal paths carried to that connector.



* Mem Cntrl Signals

BSAE
BP
PHIL



PROM SIZE JUMPERS

SIZE OF PROM CHIPS	INSERT JUMPERS
256 x 4 bits	W1, W3, W6
512 x 4 bits	W2, W4, W5, W7

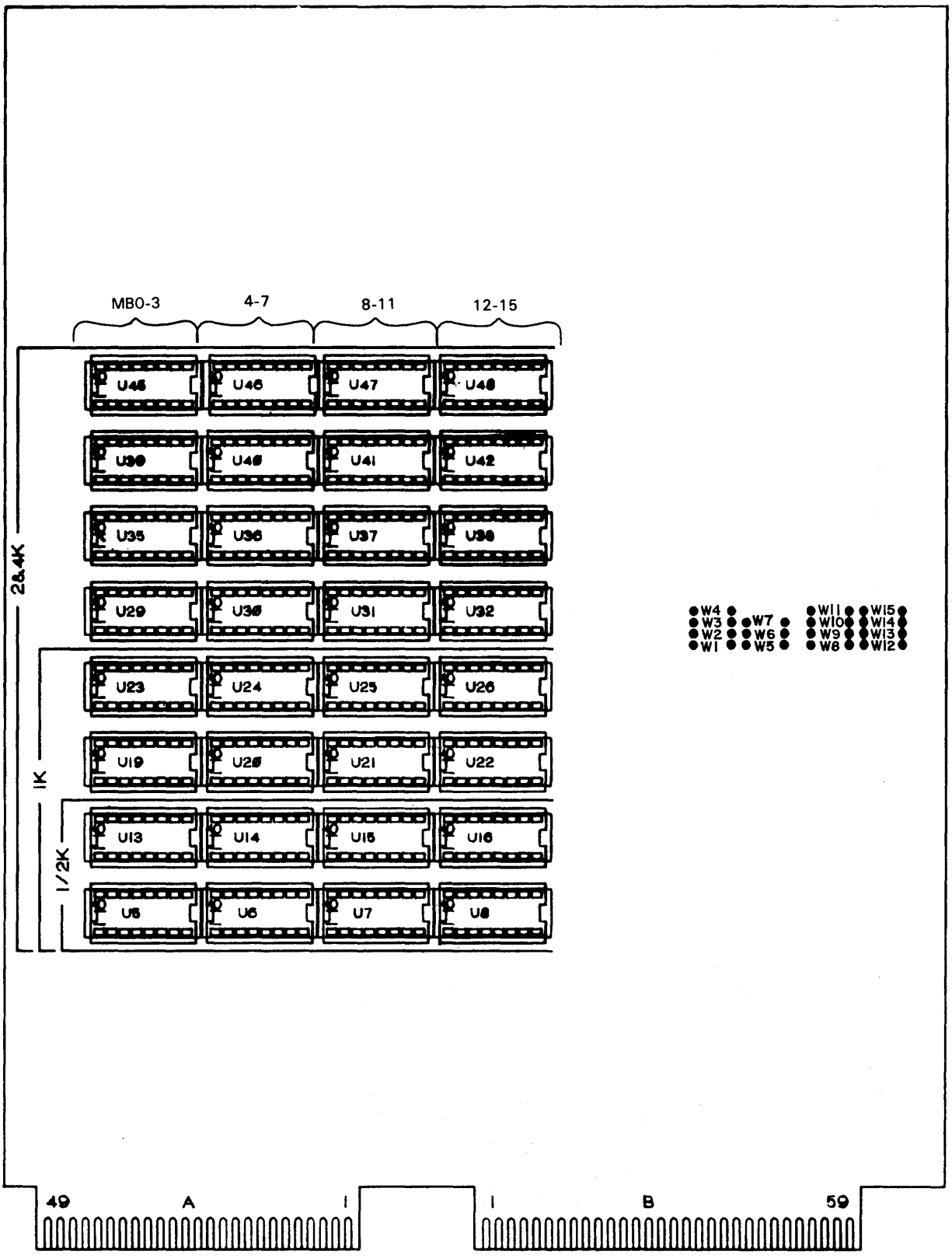
ADDRESS SELECTION JUMPERS

BIT POSITIONS OF STARTING ADDRESS	1	2	3	4*
INSERT JUMPER TO SPECIFY 1	W13	W15	W11	W9
INSERT JUMPER TO SPECIFY 0	W12	W14	W10	W8

*Jumpers W8 and W9 are removed on the 4K PROM board.

NOTE *All of the address selecton jumpers must be removed when the board is being programmed with the microNOVA PROM programmer.*

PROM BOARD PHYSICAL LAYOUT



This page intentionally left blank.

SECTION V
HAND-HELD CONSOLE SUBSYSTEM

Data General's model 8564 hand-held console is a portable device that mounts on the chassis front panel and is connected by a 10' ribbon cable to its IOC-based controller, which occupies a single slot in the main chassis. Data is retrieved from the 20-key pad via programmed I/O on an interrupt-per-key basis. The contents of memory location 077576$_8$ are constantly displayed as 6 octal digits via the 7-segment LED display. The standard console software, supplied in 256 16-bit word of ROM/RAM on the controller board, provides reset, program load, start/stop/continue, and octal examination and modification of memory and CPU registers. With suitable software, the actions of all keys may be redefined by the user.

OVERVIEW AND INTRODUCTION

Computer consoles usually are hard-wired extensions of the central processor used to monitor system performance and to aid in fault finding when the system malfunctions. For large computer systems, the processor console is the province of the field serviceman. With the advent of minicomputers, the processor console has also been used by programmers both as an aid in debugging programs and as an I/O device. As an I/O device, the minicomputer processor console is sometimes used to enter information into the system and to display the status of an application program.

The microNOVA hand-held console, when used with the standard console software, provides functions normally found on a minicomputer processor console and can be used both for trouble shooting and program debugging. The fact that the hand-held console is fully user-programmable means that it is extremely versatile and may be used in many application environments as a key-entry/visual-response I/O device.

The hand-held console subsystem consists of a lightweight, portable console which contains a 20-key pad and a digital readout which displays the contents of one 16-bit word of memory as 6 octal digits. The console controller is designed around the mN603 IOC circuit and occupies one slot in the main computer chassis. The controller contains the standard console software in 256 16-bit words of ROM/RAM.

SUMMARY OF CHARACTERISTICS

Key Pad:	20-key, calculator format. Keys are 0.425in(w) by 0.250in(h) on centers of 0.625in.
Display:	6-digit, 7-segment LED. Each digit is 0.3in(h).
Interface Requirements:	IOC-based controller occupies 1 slot in main chassis. Controller connects to console via 16-conductor 10' ribbon cable.
Packaging:	Injected molded, 2-piece case with living hinge key pad as part of top.
Location Requirement:	Within slots 3-8
Mounting:	Attaches to depression in front panel with Dzus fastener. Cable is held behind front panel.
HHC Dimensions:	4.85in(h), 3.30in(w), 1.2(d). 12.3cm(h), 8.4cm(w), 3cm(d).
Board Dimensions:	7.50 x 9.9 in (19 x 24.9 cm)
Maximum Operating Temperature:	131°F (55°C)

microNOVA HAND-HELD CONSOLE SUBSYSTEM

* Mem Cntrl Signals

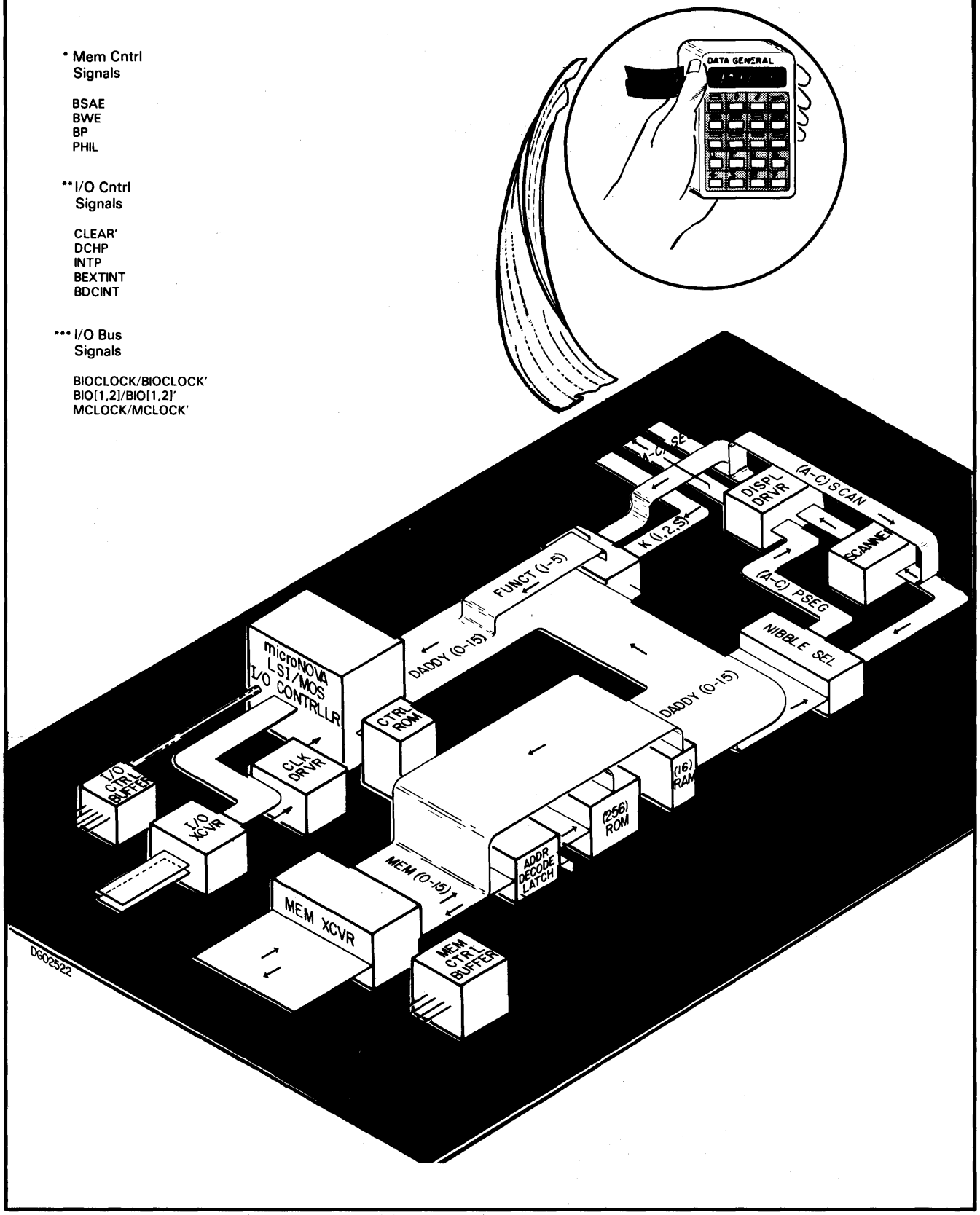
BSAE
BWE
BP
PHIL

** I/O Cntrl Signals

CLEAR'
DCHP
INTP
BEXTINT
BDCINT

*** I/O Bus Signals

BIOCLOCK/BIOCLOCK'
BIO[1,2]/BIO[1,2]'
MCLOCK/MCLOCK'



BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components plus the data and control paths of the microNOVA hand-held console (HHC) subsystem. The subsystem functions as three distinct units: the I/O interface, HHC memory, and the display and keyboard unit and controller.

I/O Interface

The I/O Interface comprises the IOC and its support circuits, a 5-bit key function register, and a 16-bit data path whereby the contents of a selected RAM location in the memory section can be transferred in bit-parallel fashion to the IOC.

The IOC and its clock driver, the I/O bus transceiver and the I/O control buffer interface the HHC subsystem to the I/O bus of a microNOVA CPU. The protocols by which the CPU communicates with peripherals is detailed in the technical reference cited above.

The control ROM contains control codes that translate IOC function codes into command signals that initiate transfer operations in all sections of the HHC subsystem. It initiates a control sequence whereby the memory section places a single word from RAM on the 16-bit input path to the IOC, and it initiates a program interrupt request when the keyboard controller places the function code resulting from a keystrike into the key function register. It also places the contents of the key function register on the input lines of the IOC when a Read Function command is received from the I/O bus.

HHC Memory

The memory section of the HHC subsystems consists of a 256 16-bit word memory module containing ROM overlaid with sixteen 16-bit words of RAM. The section includes memory, an address decoder and latch, and a memory transceiver and control buffer.

The memory transceiver and control buffer interface the memory section of the HHC subsystem to the memory bus of a microNOVA CPU. The protocols by which the CPU communicates with memory are detailed in the technical reference cited above. The lowest address in the memory section is 077400_8; any RAM elsewhere in the microNOVA system assigned to addresses 077400_8 - 077777_8 is overridden by the HHC memory section and will not return any data in Read or Read-Modify-Write operations.

When a memory address appears on the bus at the beginning of a memory operation, the address decoder latches the 8 low-order bits of that address. If the operation is a memory read, the latched address is presented to the ROM module and driven onto the module's data out lines through the memory transceiver and onto the memory bus. When the 8 bits latched lie in the range 160_8 - 177_8, the 16 word RAM is also enabled. The low-order 4 bits from the address latch select a word from RAM, and it is driven onto the module data out lines at the same time as the word retrieved from ROM. As the 16-bit data word is driven onto the memory bus, the HHC asserts the signal **PHIL** which disables any RAM in the system having the same addresses as those used by the HHC. If the memory operation is a Write operation, the operation takes place normally only if the address lies within the 16 word overlay; otherwise the operation is ignored. Because the 16 word overlay is built from static memory, the Refresh operation is ignored.

Hand-held Programmer's Console

The hand-held console is a portable device that mounts on the chassis front panel and is connected by a 10' ribbon cable to its IOC-based controller, which occupies a single slot in the main chassis. The standard console software, supplied in 256 16-bit words of ROM/RAM on the controller board, provides reset, program load, start/stop/continue, and octal examination and modification of memory and CPU registers. The hand-held console provides the functions normally found on a minicomputer processor console and can be used for both troubleshooting and program debugging.

Display and Control

The console digital display is a 6 digit 7-segment optoelectric readout, driven one digit at a time by an octal to seven-segment decoder; a 6-step scanner in the subsystem sequentially selects a 3-bit nibble from the word to be displayed, and simultaneously enables the appropriate readout cell in the display for that nibble.

The display control repetitively displays the contents of memory location 77576_8. It frequently retrieves that word from HHC memory independently of the CPU

when the memory is not performing a memory operation with the CPU. In synchronism with an internal clock the control initiates a sequence wherein memory places the contents of location 77576_8 on its memory's data out lines and into the nibble selector.

As the scanner steps through each digit, the scan number as well as the octal nibble for that scan latch into the display line driver. The HHC decodes the scan number and enables one readout cell, while it decodes the octal nibble and lights appropriate segments in that cell.

Keyboard and Control

The keyboard is a 4 column by 5 row switch matrix with its row lines driven by the display scanner; 4 column lines are encoded to a 2-bit function code and presented with a strobe signal to the key control. The key control debounces the strobe signal and latches the value of the scan number and the column code into the key function register. The key control then sets HHC Done which initiates a program interrupt request. The key control latches the value of the 5 function lines each time a key strobe is detected.

PROGRAMMING

The microNOVA hand-held console is a light-weight, portable, key-entry/visual-response data terminal. The console subsystem, which consists of a controller and the console itself, makes the function code of each key struck available to any applications program and also allows the console display to be easily updated. The standard console software uses these capabilities to provide a mechanism for controlling and monitoring the actions of a microNOVA computer system.

There are two well-defined entry points in the standard console software. Location 77777_8 contains the address which should receive control upon system initilization and auto-restart. Location 77776_8 contains the address which should receive control when a console key has been struck.

When using the standard console software from a system interrupt handler, restore the accumulators and the carry bit before transferring control. The console software assumes that the accumulators and the carry bit that it receives are those of the interrupted program.

The standard console software maintains a console switches register in its local RAM. Retrieve this information by issuing a DIA instruction to the hand-held console or by reading memory location 077577_8.

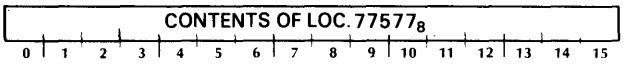
To use the hand-held console as a key-entry/ visual-response terminal, read the function code when an interrupt comes from the console and perform whatever function you want to associate with that code. To alter the 6 digits in the display, store the desired value in location 077576_8. Alternatively, location 077576_8 can be changed via the data channel from a special-purpose I/O device of your own making.

PROGRAMMING SUMMARY

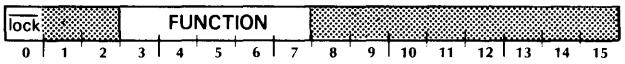
Mnemonic	HHC
Device Code	4
Priority Mask Bit	5
Display Word Location	77576_8
Console Switches Location	77577_8
Auto-restart Location	77777_8
Normal Interrupt Location	77776_8

ACCUMULATOR FORMATS

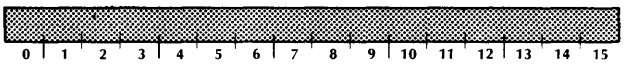
Read Switches (DIA)



Read Function (DIC)



Light Decimal Point (DOC)



The device flag commands control the hand-held console controller's Busy and Done flags in the following manner:

f=S Set the Busy flag to 1 and the Done flag to 0.

f=C Set the Busy flag to 0 and the Done flag to 0.

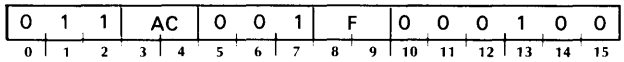
f=P Set the Done flag to 0.

Instructions

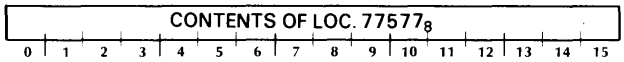
Read Switches

READS *ac*

DIA *[f]* *ac*, **HHC**



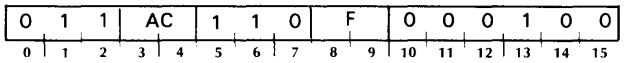
The current value of the console switches register is retrieved from memory location 77577_8 and placed in the specified AC. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



Bits	Name	Contents
0-15	Switches	Contents of memory location 77577_8.

Light Decimal Point

DOC *[f]* *ac*, **HHC**

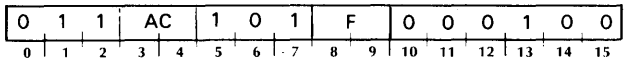


The decimal point to the right of the right-hand digit on the console is lit. The contents of the specified AC are ignored and remain unchanged. After the transfer, the function specified by F is performed.

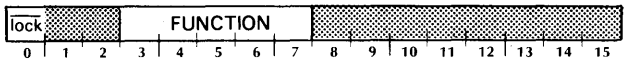
NOTE *In order to keep the decimal point visible, this instruction must be issued at least once every 4 milliseconds.*

Read Function

DIC *[f]* *ac*, **HHC**

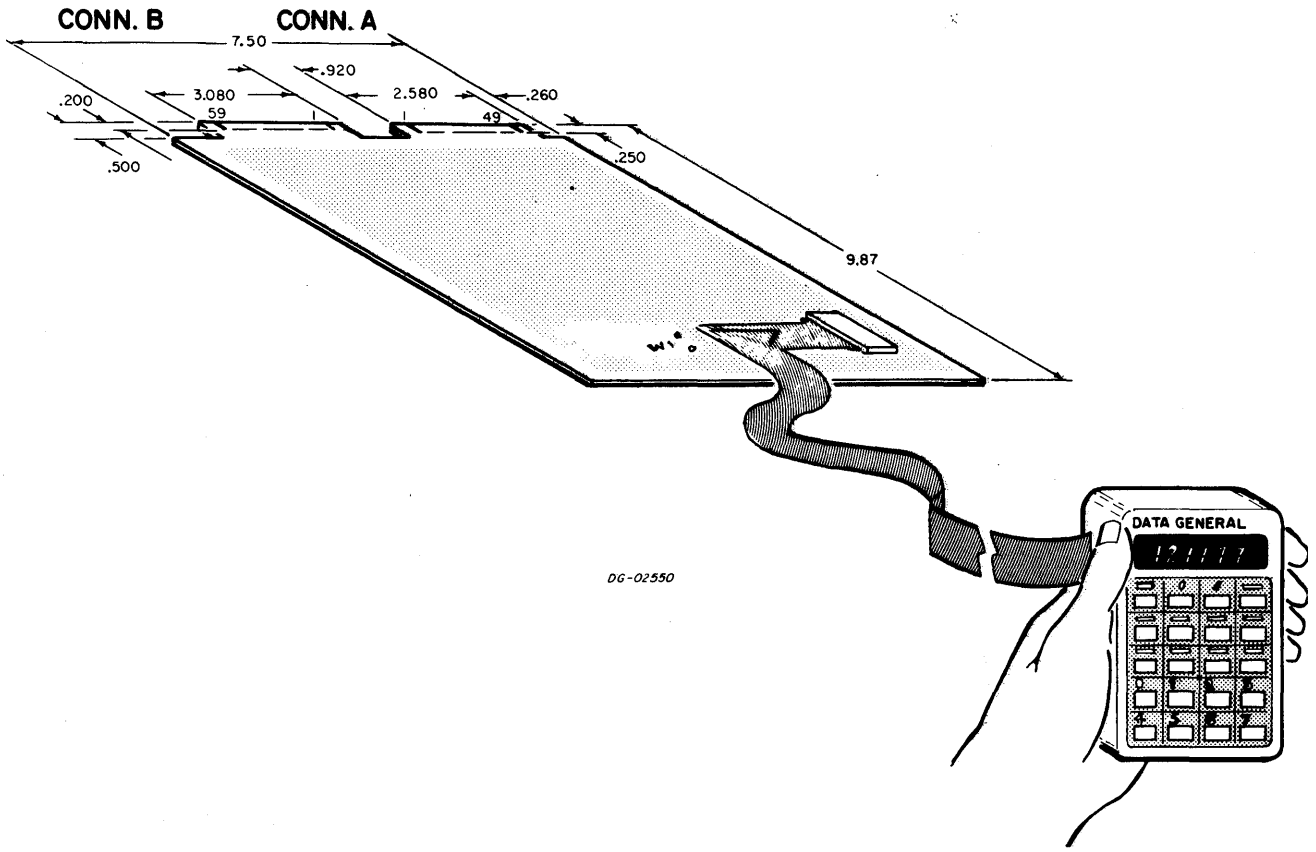


The function code of the most recently struck console key is placed in bits 3-7 of the specified AC. Bits 0-2 are set to 0. Bits 8-15 are unpredictable. After the transfer, the function specified by F is peformed. The format of the specified AC is as follows:



Bits	Name	Contents
0	Mode	Lock
1-2	----	Reserved for future use.
3-7	Function	Function code of the most recently struck console key.
	00 000	0/AC0
	00 001	1/AC1
	00 010	2/AC2
	00 011	3/AC3
	00 100	4/FP
	00 101	5/SP
	00 110	6/SWITCHES
	00 111	7/ADDR
	01 000	LAST
	01 001	NEXT
	01 010	MEM
	01 011	CLR D
	01 100	START
	01 101	STOP
	01 110	CONT
	01 111	DEP
	10 000	RESET
	10 001	*
	10 010	/
	10 011	PR LOAD
8-15	----	Reserved for future use.

microNOVA HANDHELD PROGRAMMER'S CONSOLE



W1	Function
IN	When bus signal $\overline{\text{LOCK}}$ is active, all keys are program disabled.
OUT	All keys are program enabled.

SECTION VI
ASYNCHRONOUS INTERFACE BOARD

Data General's model 4207 asynchronous interface board is an IOC- and UAR/T-based, interrupt driven, programmed I/O controller that provides the capability for full-duplex communication between a microNOVA computer system and an asynchronous terminal over either 20mA current loop or EIA RS232-C lines at speeds ranging from 50 to 19,200 baud with a character format of 1 start bit; 5, 6, 7, or 8 data bits; even, odd, or no parity; and 1 or 2 stop bits. The choice of line type, line speed, number of data bits, type of parity, and number of stop bits are all jumper-selectable. The controller has a modem control capability, including auto-answer. The model 4208 console debug option is an octal debugger contained in 256 16-bit words of ROM/RAM which are on the controller board.

OVERVIEW AND INTRODUCTION

In order to perform useful work, a computer system needs some way to communicate with the outside world. By the same token, in order to communciate with the computer system, the user needs some convenient method to give both data and commands to the system. With the advent of low-cost, asynchronous terminals, both these needs have been met by a single device. An asynchronous terminal may be used as a system console device to control the actions of the system or to interact with the user via an applications program. Using multiple terminals, one system, with suitable software, can service many users at the same time.

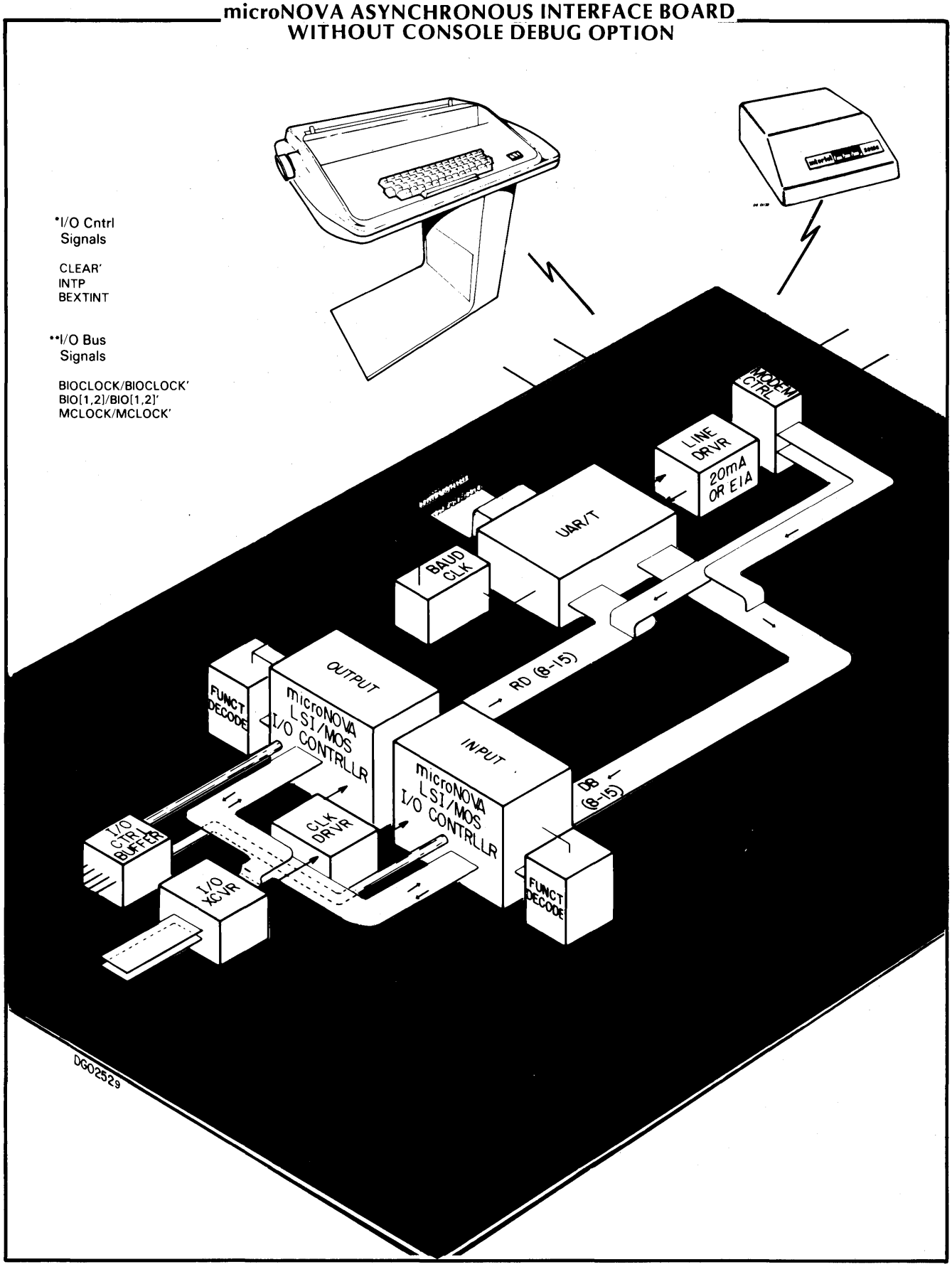
Data General's model 4207 asynchronous interface board allows a microNOVA computer system to communicate with the outside word via a wide assortment of asynchronous terminals. These terminals may range from teletypewriters and teletypewriter-compatible terminals to asynchronous serial printers to video displays.

The model 4208 console debug option available with the asynchronous interface board is an octal debugger with full memory and CPU register examination/modification capability and a software breakpoint facility. This option allows the terminal connected to the asynchronous controller to be not only a system console, but also to be a debugging and program development tool.

SUMMARY OF CHARACTERISTICS

Comm. type:	Asynchronous, bit serial full-duplex.
Line Speed:	50 to 19,200 baud, jumper-selectable.
Line Type:	20ma current loop or EIA-RS232C, jumper-selectable.
Data Structure:	1 start bit; 5, 6, 7, or 8 data bits, jumper-selectable; even, odd, or no parity, jumper-selectable; 1 or 2 stop bits, jumper-selectable.
Modem Signals Available:	Carrier On, Data Set Ready, Data Terminal Ready, Ring Indicator, Break Indicator
Board Dimensions:	7.50 x 9.9 in (19 x 24.9 cm)

microNOVA ASYNCHRONOUS INTERFACE BOARD WITHOUT CONSOLE DEBUG OPTION



BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components and the data and control paths in the asynchronous serial interface without the console debug option. The drawing on the next page shows the same block diagram with the console debug option installed. The console debug option and the serial interface sections of the board function independently of each other.

Asynchronous Serial Interface

To allow full-duplex operation, the interface responds to two I/O device codes, one for input data and one for output data. The circuit includes input and output IOC's with a clock driver and function decoders, a UAR/T, a crystal-driven baud rate clock, a line driver/receiver for a 20mA current loop or EIA RS 232-C transmission line, and modem control including automatic answer capability.

The IOC's interface the UAR/T to the I/O bus according to the protocols by which the CPU communicates with peripherals. These protocols are described in detail in the Technical Reference for microNOVA Integrated Circuits *(DGC no. 014-000074)*.

The UAR/T accepts 5, 6, 7, or 8-bit wide data bytes from the output IOC via the write bus, and serializes the data, adding start, parity, and stop bits as selected by line characteristics jumpers. The resulting bit serial data stream is passed to an external data terminal, communication modem or other device from the output connector ("A" connector) via 20mA current loop or EIA RS232-C line drivers.

Asynchronous serial input data enters via the "A" connector, and passes via the 20mA current loop or EIA line receiver to the UAR/T. The UAR/T strips format bits from the data stream and presents an 8-bit parallel data byte to the input IOC via the read bus.

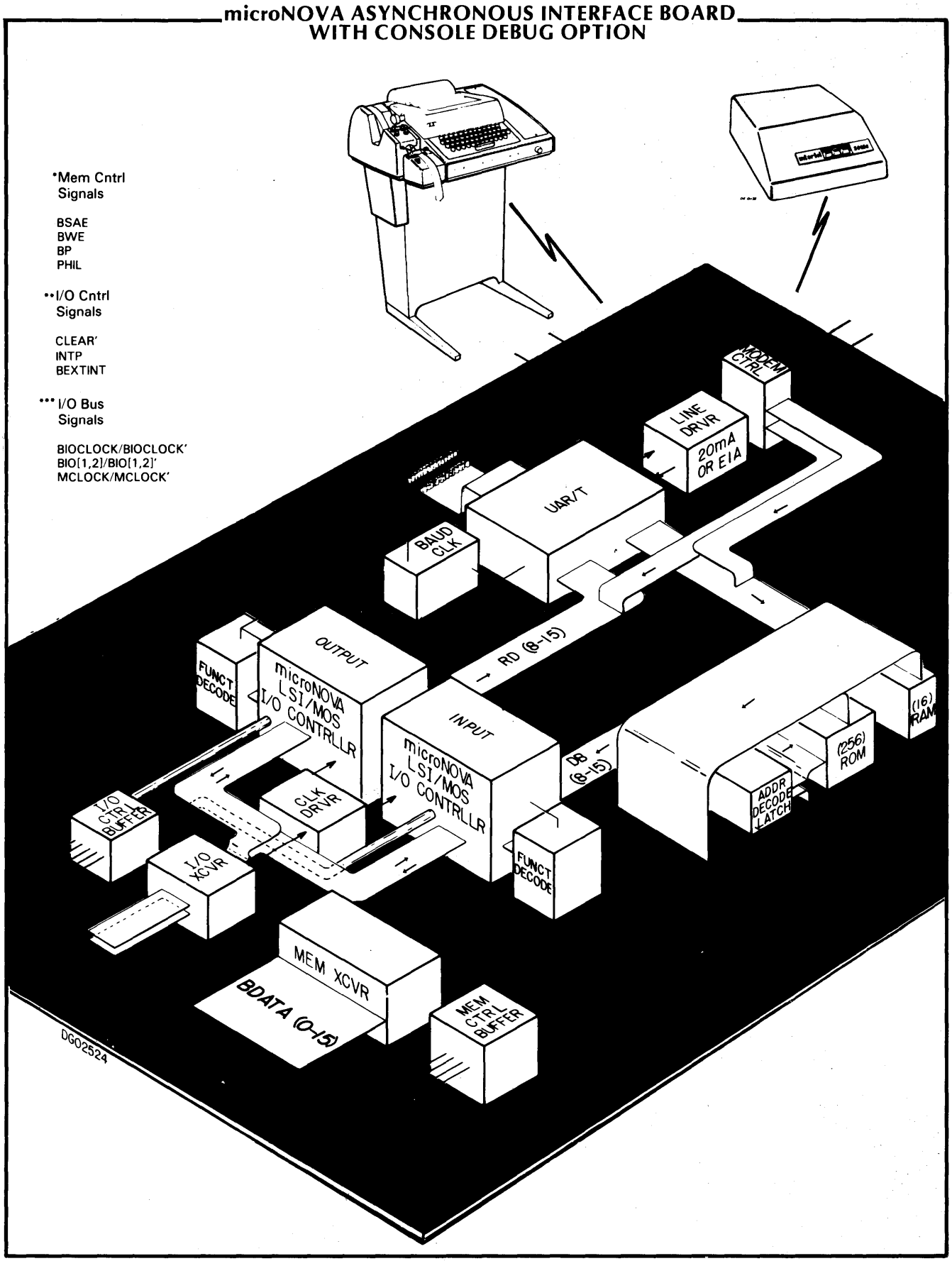
Console Debug Option

The console debug option to the asynchronous serial interface board consists of a 256 16-bit word memory module comprising ROM overlaid with sixteen 16-bit words of RAM. The option includes memory, an address decoder and latch and the memory transceiver and control buffer that connects the memory module to the microNOVA memory bus.

The memory transmitter, memory receiver, and control buffer interface the memory section of the console debug option to the memory bus of a microNOVA CPU. The protocols by which the CPU communicates with memory are detailed in the technical reference cited above.

The console debug option asserts the signal **PHIL** to disable any RAM located in the system assigned the same addresses as those used by the console debug option.

microNOVA ASYNCHRONOUS INTERFACE BOARD WITH CONSOLE DEBUG OPTION



PROGRAMMING

The microNOVA asynchronous interface board allows communication between a microNOVA computer system and a serial asynchronous terminal. When equipped with the console debug option, the asynchronous interface board gives the capability for an asynchronous terminal to be not only the system console but also a debugging and program development tool. To this end, the controller takes care of character assembly and disassembly, parity generation and checking, and character buffering. The controller allows characters to be received from and transmitted to the terminal, and provides program control of various modem status and control signals.

To transmit a character, load the character into the transmitter buffer, and set transmitter busy by issuing a Start command.The controller asserts Request to Send and waits until the modem signal Clear To Send is asserted; adds the required start, parity, and stop bits; and then transmits the character serially to the terminal. After the transmission is complete, the controller initiates an I/O interrupt request.

Reception of a character requires no initiating action. The controller detects a character being transmitted from the terminal; assembles it; strips the start and stop bits; and then initiates an I/O interrupt request. Read the character before one full character time has elapsed to avoid the character being over written by the next character on the line. To read a character from a teletypewriter paper tape reader, give a Start command to initiate the transfer of the next character.

To answer calls from a modem that is configured for auto-answer, periodically read the modem status register and check for the presence of the Ring Indicator. If the Ring Indicator is a 1, assert the signal Data Terminal Ready. This will complete the connection and allow communication to begin.

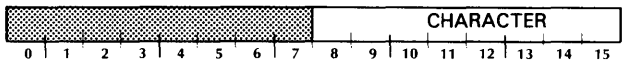
The console debug option is a stand-alone program contained in 256 16-bit words of ROM/RAM on the controller board. The console debug option cannot be programmed.

PROGRAMMING SUMMARY

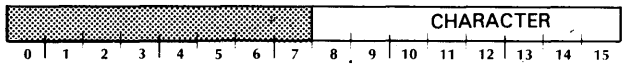
Primary Mnemonic	
Transmitter	TTO
Receiver	TTI
Primary Device Code	
Transmitter	11
Receiver	10
Secondary Mnemonic	
Transmitter	TTO1
Receiver	TTI1
Secondary Device Code	
Transmitter	51
Receiver	50
Priority Mask Bit	
Transmitter	15
Receiver	14

ACCUMULATOR FORMATS

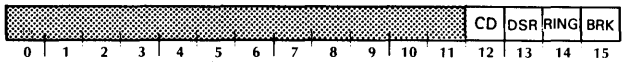
Read Character **(DIA to TTI)**



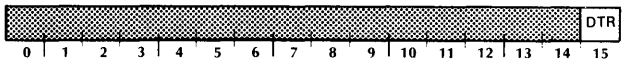
Write Character **(DOA to TTO)**



Read Modem Status **(DIB to TTI)**



Control Data Terminal Ready **(DOB to TTI)**



The device flag commands control the receiver's Busy and Done flags in the following manner:

f=S Set the Busy flag to 1 and the Done flag to 0. Start the Reader.

f=C Set the Busy flag to 0 and the Done flag to 0.

f=P Set the Done flag to 0, and set the Break Indicator in the modem status register to 0.

The device flag commands control the transmitter's Busy and Done flags in the following manner:

f=S Set the Busy flag to 1 and the Done flag to 0.

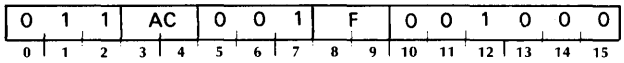
f=C Set the Busy flag to 0 and the Done flag to 0.

f=P Set the Done flag to 0.

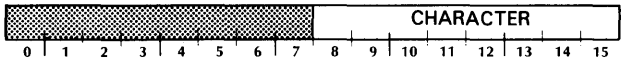
Instructions

Read Character

DIA *[f]* *ac*, **TTI**



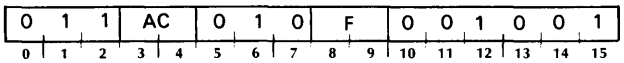
The contents of the receiver buffer are placed in bits 8-15 of the specified accumulator. If the character has less than 8 data bits, it is placed right-justified in bits 8-15. Bits 0-7 are set to 0. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



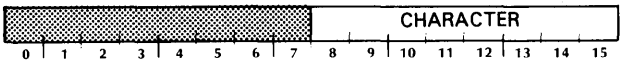
Bits	Name	Contents
0-7	----	Reserved for future use.
8-15	Character	The character most recently received (right-justified).

Write Character

DOA *[f]* *ac*, **TTO**



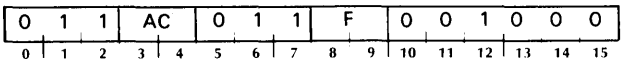
Bits 8-15 of the specified accumulator are placed in the transmitter buffer. If the character has less than 8 data bits, it should be placed right-justified in bits 8-15. Bits 0-7 of the AC are ignored. Turn on Request to Send. Begin transmitting the contents of the transmit buffer as soon as Clear to Send is received from the modem. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



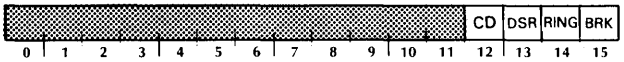
Bits	Name	Contents
0-7	----	Reserved for future use.
8-15	Character	The character to be transmitted (right-justified).

Read Modem Status

DIB *[f]* *ac*, **TTI**



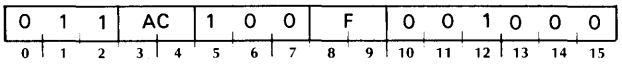
The contents of the modem status register are placed in bits 12-15 of the specified accumulator. Bits 0-11 of the AC are set to 0. After the transfer, the function specified by F is performed. The format of the specified AC is as follows.



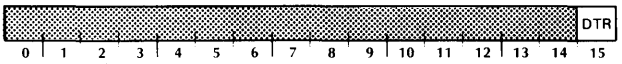
Bits	Name	Contents
0-11	----	Reserved for future use.
12	Carrier Detect	The state of the Carrier Detect signal (0=off, 1=on).
13	Dataset Ready	The state of the Dataset Ready signal (0=off, 1=on).
14	Ring Indicator	The state of the Ring Indicator signal (0=off, 1=on).
15	Break Indicator	If the bit is 1, a break condition (line open and spacing) has been detected by the receiver.

Control Data Terminal Ready

DOB *[f]* *ac,* **TTI**

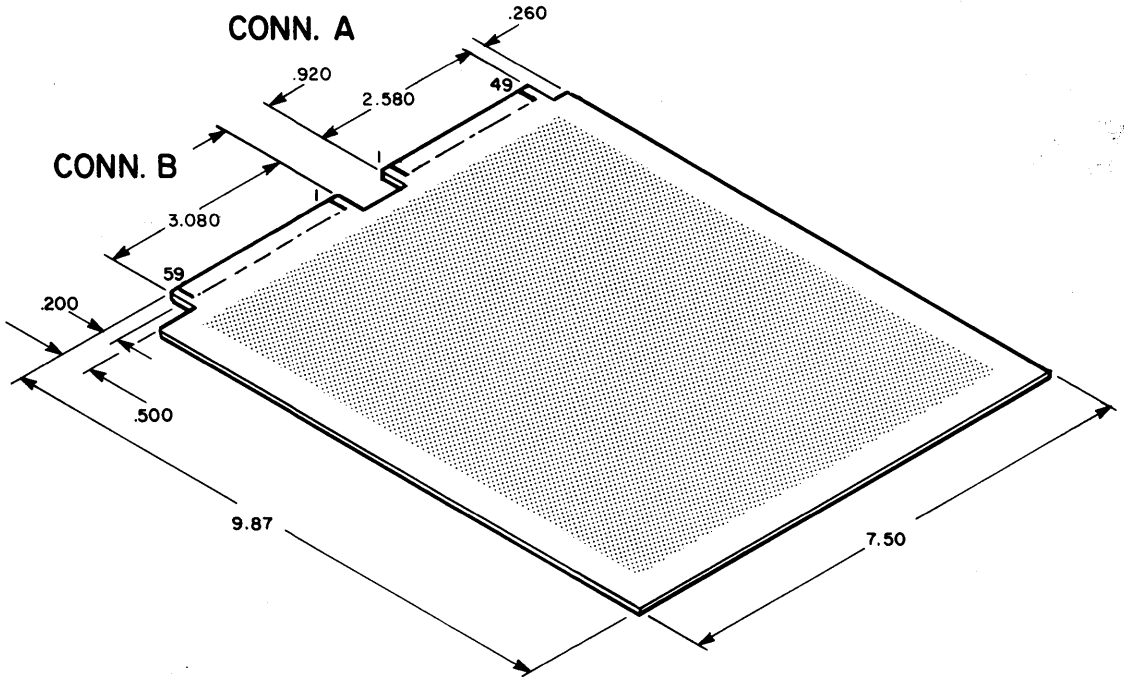


Bit 15 of the specified accumulator is used to set the Data Terminal Ready signal. If bit 15 is 1, the Data Terminal Ready signal is turned on. If bit 15 is 0, the Data Terminal Ready signal is turned off. Bits 0-14 of the AC are ignored. After the state of the signal has been set, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



Bits	Name	Contents
0-14	----	Reserved for future use.
15	Data Terminal Ready	The state of the Data Terminal Ready signal (0=off, 1=on).

ASYNCHRONOUS INTERFACE BOARD DIMENSIONS



DEVICE CODE JUMPERS

BIT POSITIONS OF DEVICE CODE (Insert jumper to specify 1)	0	1	2**	3	4	5
RECEIVER	W1	W2	W3	W4	W5	0*
TRANSMITTER	W6	W7	W8	W9	W10	1*

*The low-order bit of the device code of the receiver is 0, and the low-order bit of the device code of the transmitter is 1.

** First controller device code

CHARACTER LENGTH JUMPERS

LENGTH OF CHARACTER	W18	W19	DGC MODELS
5 bits	in	in	
6 bits	out	in	
7 bits	in	out	
8 bits	out	out	ALL

PARITY JUMPERS

TYPE OF PARITY	W17	W15	DGC MODELS
EVEN	out	in	
ODD	in	in	
NONE	out	out	ALL

BAUD RATE JUMPERS

BAUD RATE	W14	W13	W12	W11	DGC MODELS
50	in	in	out	in	
75	in	in	out	out	
110	out	out	out	out	4010
134.5	in	out	in	in	
150	out	out	out	in	
200	in	out	in	out	
300	out	out	in	out	6042-6043
600	in	out	out	in	6040-6041
1200	out	in	out	out	
1800	out	in	out	in	
2400	out	out	in	in	
4800	out	in	in	out	6012H
9600	out	in	in	in	
19,200	in	in	in	out	

STOP BIT JUMPERS

NUMBER OF STOP BITS	W16	DGC MODELS
1	in	4010
2	out	OTHERS

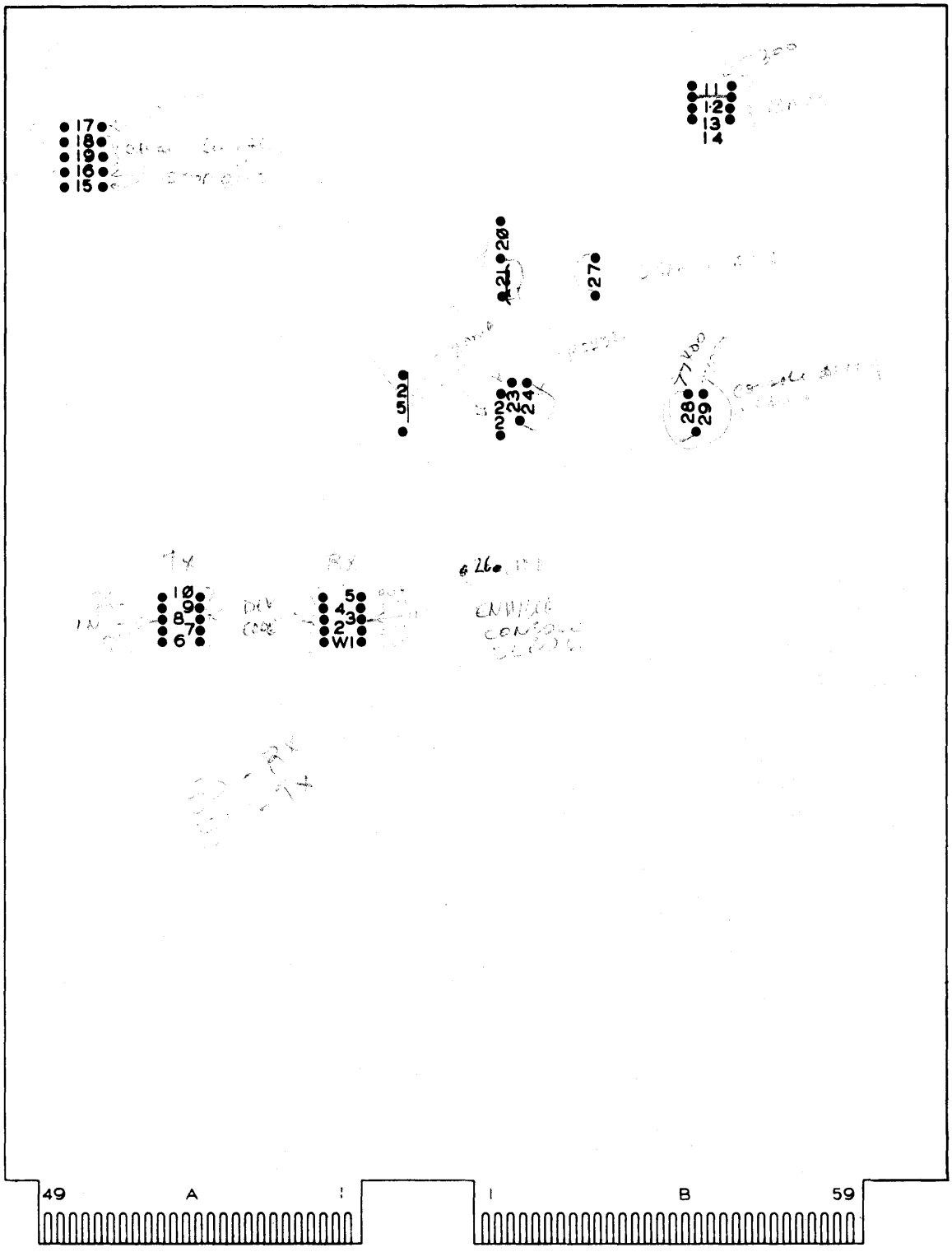
TYPE OF TRANSMISSION JUMPERS

TYPE OF TRANSMISSION	INSERT JUMPERS	DGC MODELS
20mA Current Loop	W20, W22, W23, W25	4010
EIA RS232-C	W21, W24	OTHERS

OTHER JUMPERS

JUMPER	FUNCTION	DGC MODELS
W26	Insert jumper to enable the console debug option memory.	
W27	Insert jumper to disable the use of the modem status signal Clear To Send.	ALL BUT 6040, 6041
W28	Insert for console debug address space 77400-77777	
W29	Insert for console debug address space 77000-77377	

ASYNCHRONOUS INTERFACE BOARD - JUMPER LAYOUT



This page intentionally left blank.

SECTION VII
DISKETTE SUBSYSTEM

Data General's model 6038 and model 6039 diskette subsystems are self-contained, IOC-based, mass storage, software-driven subsystems providing direct access, moving head disc memory for microNOVA computers. Subsystems store 157,696 16-bit computer words in single block (256 word) transfers at 15,625wps to hard/soft format diskettes that are fully compatible with DGC's 6030 series subsystems (for NOVA and ECLIPSE line computers). Multiple 2-drive subsystems may connect on the microNOVA's extended I/O bus (max. 100 ft bus). Software support includes a disc operating system subset of Data General RDOS.

OVERVIEW AND INTRODUCTION

Mass storage provides the capability of maintaining very large amounts of machine readable data at a fraction of the cost of main, or random access memory. Diskette subsystems provide mass storage on compact, inexpensive, easily handled, direct access media. They offer faster data access and higher transcription speed than sequential media (e.g. paper and magnetic tape and cards). They are useful for online system applications not requiring the capacity and speed of higher performance direct access systems. The direct access diskette subsystem can support a powerful operating system; this can lead to an appreciable reduction in application program development time.

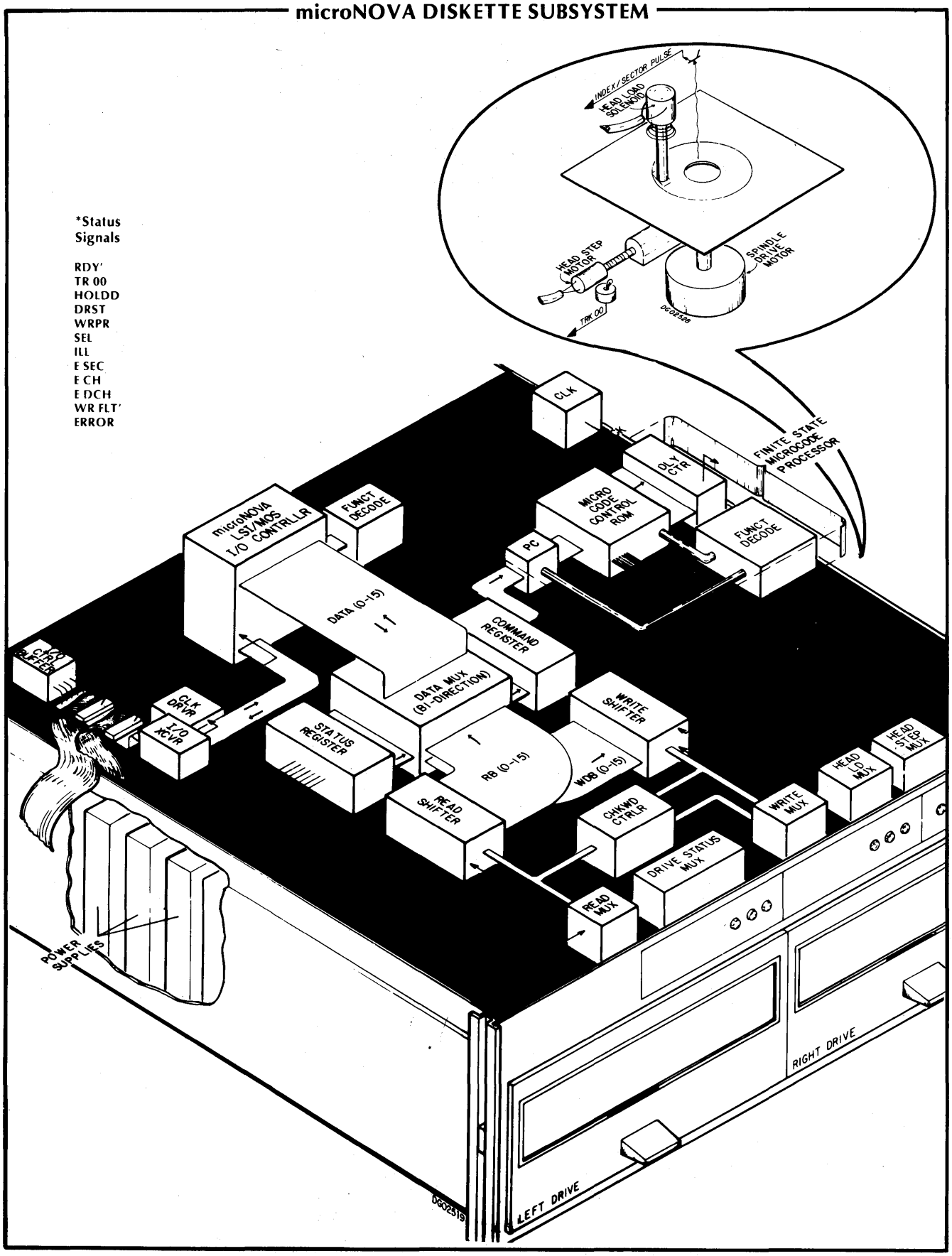
Data General's microNOVA series diskette subsystems provide powerful reliable, data storage capability for microNOVA computers. They are available in single and dual drive units.

These diskette subsystems combine a drive mechanism, power supply and packaging scheme proven in Data General's 6030 series diskette subsystems. They include an interface designed around the microNOVA system IOC.

SUMMARY OF CHARACTERISTICS

Max. Storage:	157,696 16-bit words per diskette.
Transfer Rate:	15,625 words/second via data channel.
Data Format:	77 tracks, 8 sectors each with 256 16-bit words plus address and check fields. Hard sector marks prepunched in media. Soft sector format prewritten onto media.
Packaging:	Single or Dual Drive enclosure, either drive may be assigned unit number 0.
Interface Requirements:	Couples directly to extended I/O bus of microNOVA system and requires no space in the computer chassis. Four enclosures each with 2 drives may be cabled on bus with maximum total bus length 100 feet.
Seek Time:	10ms, step in or step out per track.
Sector Access:	166ms (max) 1.4ms (min) ± 3.5% (360rpm).
Head Load/Settle Time:	60ms, minimum (automatic head load implied on all commands); 10ms, after last step in or out command; 170ms, after changing unit selection.
Head Unload Time:	249ms (nominal) after last operation; immediately, upon unit deselection.
Diskette Type:	Industry standard 7.88in dia. 3mil mylar media with Data General specified punched sector and write protect holes.
Write Current Compensation:	Two level write current compensation for head to disc speed and effective data density variation with track position.
Head Type:	Single gap, ceramic.
Data Density:	3268bpi (inner track)/1836bpi (outer).
Power Required:	100/115/220/240Vac ± 10%) 50/60Hz.
MTBF/MTTR:	4000 hours operating (after initial 200hrs)/20 minutes.
Dimensions:	19in(w), 7in(h), 24in(d) slide rail mounting in 19in NEMA cabinet.
Weight:	54lbs (single drive), 67lbs (dual).

microNOVA DISKETTE SUBSYSTEM



BLOCK DIAGRAM

The subsystem block diagram shows the principal elements in the diskette subsystem. All information transfers between the diskette subsystem and CPU occur via the extended microNOVA I/O bus. (The microNOVA serial I/O bus is described in the CPU section of this publication; its functional protocols are detailed in the Technical Reference for microNOVA Integrated Circuits *(DGC no. 014-000074)*.

Subsystem Interface and Controller

The I/O bus transceiver, the IOC, and a function decoder perform level translation, format conversion and control synchronization between the high speed, differential, serial, I/O bus and the internal control and data registers in the diskette subsystem.

The diskette command register decodes the operation specified by the program and initializes the micro-coded timing and state generator. The state generator is a ROM-controlled microcode processor that steps the diskette subsystem in an ordered sequence through all operations in synchronism with a crystal-controlled master clock. The ROM microcode enables the required paths through the multiplexor, identifies illegal commands, delineates precise time intervals, controls format and address checking during data transfers, unloads the read/write head during lapses in subsystem activity, and requests data channel and program interrupt service when needed for a diskette operation.

Data Format

Since there is only one recording suface on a diskette (numbered 0), each cylinder contains a single track. There are 77 cylinders, numbered $0\text{-}114_8$, and each cylinder is divided into 8 sectors, numbered 0-7. A sector can store 256 (400_8) 16-bit data words and contains a checkword which is hardware-verified whenever data is read. Data storage capacity is 2,048 words/track and 157,696 words/diskette. Words are transferred via the data channel to and from a diskette unit in a disc subsystem at the rate of 15,625 words/sec.

2F Data Transcription

Data transcription to and from diskette employs the 2F technique which combines both data and clocking information in a serial bit stream. The microcode processor combines serial data from the write shifter with clock pulses, forming a 2F encoded data writing stream. During reading, 2F encoded data from the read head is separated into clock and data streams in the drive's data separator. Flow of the data reading stream into the read shifter is controlled by the microcode processor.

Dual Drive Multiplexors

Principal signals to and from each diskette drive pass through dual drive multiplexors and ultimately share most of the major subsystem control and data path resources. The contents of UNIT SELECT in the subsystem command register control the state of the drive multiplexors. A selection jumper (not shown) establishes drive identities by assigning unit number 0 to the left or the right drive.

Information Flow and Control Sequences

Diskette operations include set up and status transfers (Specify Diskette Address, Specify Diskette Operation, Specify Memory Address, Read Status) head positioning (Step In, Step Out, and Settle Head), address verification (Read Preamble) and data transfers (Read Next Sector, Write Next Sector, Write Format). Each kind of operation implies a distinct sequence of events and a unique routing of information. The following summaries typify the data paths and control sequences for each category of operation.

Setup and Status Transfers

All setup and status transfers are one-step operations that involve the transfer of a single data word between a CPU accumulator and some register in the subsystem. In all cases a programmed I/O instruction specifies the transfer, and control signals from the function decoder enable the appropriate data path through the diskette multiplexor.

Destination registers in the subsystem for Setup and Status Transfers include Starting Memory Address Register (for the data channel), and Command Register. Source registers in the subsystem include Sector Preamble Register (read shifter), Memory Address Register and Status Register.

Head Positioning

A Device Flag Control signal from the function decoder, together with the contents of the diskette command register control head positioning. When the program issues any head positioning command, the read/write head loads (if unloaded) and the Step In or Step Out operation is initiated by the microcode controller; no additional operations may be initiated for another 10ms. If the program issues no additional command within two disc rotations, the controller unloads the head.

Stepping the head involves motion of the head parallel to the media. The program can instruct the subsystem to step the head of either of the two drives either forward (in or away from track 00) or backward (out or towards track 00) or to settle (no in or out movement). In all three cases, the device will unconditionally set its done flag after 10ms.

The loading or unloading of the head involves motion of the head which is perpendicular to media surface. The control board functions as described below.

When either of the two drives is addressed by the program, the control board issues a load head command to the selected drive. This command will remain active, keeping the head in contact with the spinning media, until the occurrence of the earliest of the following events:

1. The subsystem finds itself in an idle state (not busy) for a nominal time of 1 1/2 diskette revolutions, or,

2. The program selects the other drive.

During the period of time that the program is cycling on the same drive and commands are issued to the drive within 170ms of each other, the head will remain loaded.

It is mandatory that the program not issue any sector-related commands without insuring that the head is loaded. An indication of that condition is available to the program within the status word.

Address Verification

Data transfer operations may be directed only to the *next* sector to pass under the transcription head. The Read Preamble operation allows the program to determine which disc sector is presently under the head.

When the program initiates the command, the microcode processor waits for the next hard sector mark to occur, and then enables the read circuits to scan for the soft sector mark or sync bit. The processor strips the address sync bit and loads the next 16 bits (those of the sector address field) into the read shifter and terminates the operation.

The program must then initiate a register transfer operation to move the contents of the shifter (Disc Preamble) through the multiplexor to a CPU accumulator.

Data Writing

The Write operation transfers one sector (256 16-bit words) of data from memory to the next diskette sector to pass under the write head if and only if the sector address read from the preamble of that sector matches the sector address specified in the subsystem command register. Data transfers occur independently of direct program control via the microNOVA data channel.

The microcode generator directs the flow of data during the write operation. When the program initiates the operation, the controller requests a data channel transfer to place the first word from memory into the data channel output register. The microcode processor waits for the next hard sector mark, enables the read scanner to detect and strip the address field sync bit, and then shifts the sector preamble into the read shifter. If the sector address matches that specified in the command register, the microcode processor initializes the data writing circuits, the sync bit and leader generators and the checkword calculator.

Approximately 75 microseconds after the address field passes under the head, the controller begins writing the data field leader; approximately 60 microseconds later it writes the sync bit, moves the first data word from the data channel register into the write shifter, requests another word from the data channel, and then begins shifting data to the write drivers and to the checkword generator. Data moves from memory through the data channel to the data channel buffer and then into the shifter until all 256 words have been written. The 16-bit checkword in the checkword calculator is then shifted to the write drivers followed by a string of 0's. The microcode processor terminates the operation upon the appearance of the next sector mark.

Data Reading

The Read operation initiates the transfer of data from the next sector to pass under the read head if and only if the sector address of the preamble read from the address field of that sector matches the sector address specified in the subsystem Command Register. If the addresses match, the subsystem reads 257 16-bit (data plus checkword) words from that sector. The subsystem transfers 256 data words to memory via the data channel.

When the program initiates the operation, the microcode processor waits for the next hard sector mark, enables the read scanner to detect and strip the sync bit, and then reads the 16-bit address field into the read shifter. If the sector address matches the address specified in the command register, the microcode processor initializes the data reader, detects and strips the data sync bit and begins shifting data into the read shifter. It also enables the read checkword calculator. After the processor loads 16 data bits into the shifter, it loads that word into the subsystem storage multiplexor and requests service from the microNOVA data channel. The microcode processor repeatedly shifts a 16-bit word into the shifter and moves it into the temporary register; should the data channel fail to retrieve a word from the temporary register before the next word fills the shifter, data is lost and the Data Late error flag is set to 1 in the subsystem status register.

As each data bit passes the checkword calculator it recomputes the cyclic polynomial. When the 256th data word moves from the shifter to the data channel register, one additional word (the checkword) is read from the diskette, and compared with the checkword computed by the subsystem during the read operation. If the computed and retrieved checkwords do not match, the Checkword Error flag is set to 1 in the subsystem status register.

Drive Interlocks and Status Flags

A comprehensive collection of interlocks monitor the operations of the subsystem. When the program specifies an illogical or illegal operation, an interlock stops the operation; both the data in memory and that on the diskette will remain unchanged. In the event of power failure, or if an operator opens the drive door, during a data transfer, that operation halts and the appropriate error flags are set to 1 in the subsystem status register.

Status signals from the diskette drive include Write Protect, Door Open, and Write Fault. Status signals generated in the subsystem include Head Loaded, Illegal Operation, Head Home (Track 0), Not Ready, Not Ready Latch, Address Error, Checkword Error and Data Channel Late.

INSTRUCTIONS

The diskette drive controller contains four program accessible registers: a 15-bit Memory Address Counter, a 12-bit Status Register, a 13-bit combined Command/Select register, and a 10-bit Current Diskette Address Register. The Memory Address Register is self-incrementing and contains the memory location of the next 16-bit word to be either read from or written on the diskette. The Status Register contains all the information flags for the controller and the selected drive. The combined Command/Select Register contains the command last issued to the subsystem and the number of the desired sector on the diskette surface. The Current Diskette Address Register contains the track and sector numbers obtained from the last Read Preamble operation executed.

Four instructions are used to program data channel transfers to and from the diskette. Two of these instructions are used to supply all of the necessary data to the controller for any diskette operation. The remaining two instructions allow the program to determine, in detail, the current state of the diskette subsystem.

The diskette controller's Busy and Done flags are controlled using two of the device flag commands as follows:

f-S Set the Busy flag to 1; set the Done flag to 0; set the Error, Write Fault, Data Late, Checkword, Address Error and Illegal, flags to 0; and initiate the operation specified by the contents of the Command Register.

f-C Set the Busy flag to 0, set the Done flag to 0, set all error flags to 0, and the Not Ready Latch flag to 0.

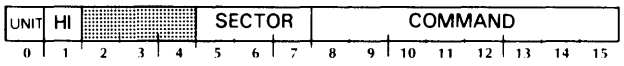
f-P No effect.

PROGRAMMING SUMMARY

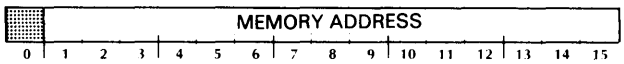
Mnemonic (first subsystem)	DKT
Device Code (first subsystem)	33
Priority Mask Bit	10
Units/subsystem	1 or 2
Surfaces/unit	1
Tracks/surface	77
Sectors/track	8
16-bit words/sector	256
16-bit words/unit	157,696
Maximum Transfer Rate (words/second)	15,625
Allowable Data Channel Latency (microseconds)	63.9
Head Step Time/track (ms)	10
Min. Head Load Time-overlaps any step command (ms)	60
Head Settle Time After Last Step (ms)	10
Head Settle Time After Unit Selection Change (ms)	170
Sector Access Time-min/max (ms)	1.4/166

ACCUMULATOR FORMATS

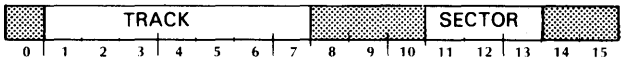
Specify Command (DOA)



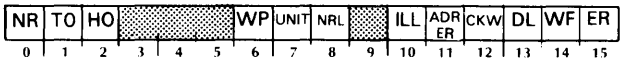
Load Memory Address Counter (DOB)



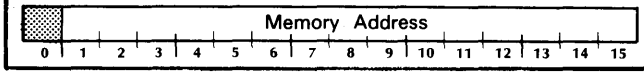
Read Current Address (DIC)



Read Status (DIA)

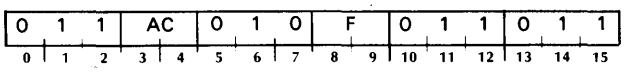


Read Memory Address Register (DIB)

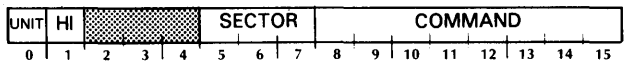


Specify Command

DOA *[f] ac,* **DKT**



Bits 0-15 of the specified AC are loaded into the diskette subsystem command register. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:

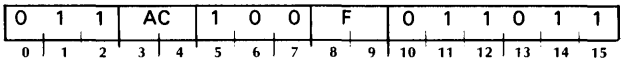


Bits	Name	Contents
0	Unit	Select the Drive (0 or 1)
1	Hi Trk	Must be set to one if the track address is greater than 43_{10} for any transfer specified by this command.
5-7	Sector	Sector address (0-7) if this command specifies a data transfer; if not, these bits are ignored.
8-15	Command	Specify the subsystem operation as follows:
	000	Settle 10ms
	001	Step out (toward Track 0)
	002	Step in (toward Track 77)
	010	Read preamble
	020	Read data, next sector *
	040	Write data, next sector *
	240	Format sector 0
	241	Format next sector

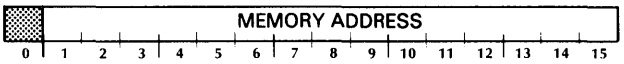
* Specified operation will not be attempted if the sector address read from the next sector does not match bits 5-7.

Load Memory Address Counter

DOB *[f] ac,* **DKT**



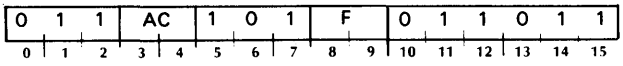
Bits 1-15 of the specified AC are loaded into the subsystem's Memory Address Register. After the transfer, the function specified by F is performed. The contents of the specified AC remain unchanged. The format of the specified AC is as follows:



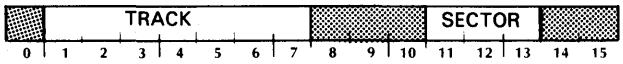
Bits	Name	Contents
0	----	Reserved for future use.
1-15	Memory Address	Location of the next word in memory to be used for a data channel transfer.

Read Current Address

DIC *[f] ac,* **DKT**



The current track and sector are placed the specified AC. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



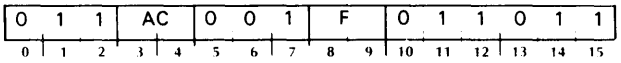
Bits	Name	Contents
0	----	Reserved for future use.
1-7	Track	Identifies the track presently under the read head ($0\text{-}114_8$)
8-10	----	Reserved for future use.
11-13	Sector	Identifies the sector presently under the read head ($0\text{-}7_8$)
14-15	----	Reserved for future use.

NOTE *The address returned by the Read Current Address instruction accurately identifies the sector presently under the read head only if certain sequence and timing restrictions are met:*

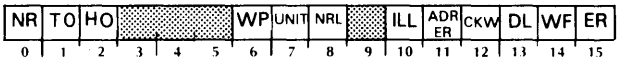
1. *A* **single** *Start flag control command is issued to the subsystem along with the Read Preamble Command; AND*
2. *The Read Current Address instruction is issued* **after** *the Done flag is set to 1 in response to (1), above, AND*
3. *The Read Current Address instruction is issued* **before** *the end of the current sector passes under the head (approximately 17-18ms after the Done flag is set to 1 in response to (1), above.*

Read Status

DIA *[f]* *ac,* **DKT**



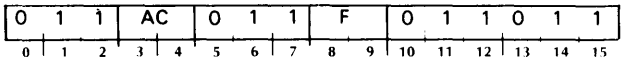
The contents of the diskette Status Register are placed in bits 0-15 of the specified AC. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



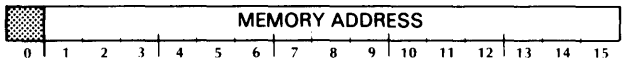
Bits	Name	Contents
0	Not Ready	The selected drive is not ready
1	Track 0	The head is positioned over track 0.
2	Head On	The head is loaded on the diskette media.
3-5	----	Reserved for future use
6	Write Protect	The diskette in the selected drive is write protected.
7	Unit	Indicates number of the selected drive (0 or 1).
8	Not Ready Latch	The Not Ready flag for the specified drive has had the value 1 at some time since the last Clear command was issued.
10	Illegal	An illegal command or sequence of commands was issued to the subsystem.
11	Address Error	The sector address read from the preamble of the sector specified for a read or write sector command does not match bits 5-7 of the command register.
12	Checkword	The checkword calculated by the subsystem did not match that read in the last sector transferred.
13	Data Late	The data channel failed to respond in time to a data channel request.
14	Write Fault	A failure in the drive during a write operation has occurred. Also set to 1 if the program attempts to write to a drive which has the door open or to a non-existent drive.
15	Error	Any of bits 10-14 is 1; or if none of the error flags are 1, no data was read from the diskette.

Read Memory Address Register

DIB *[f]* *ac,* **DKP**



The contents of the Memory Address Register are placed in bits 1-15 of the specified AC. Bit 0 is set to 0. After the data transfer, the controller's Busy and Done flags are set according to the function specified by F. When the Memory Address Register is read after a Write operation, the contents point to a memory location two greater than the location of the most recent word written on the disc. The format of the specified AC is as follows:



Bits	Name	Contents
0	----	Reserved for future use.
1-15	Memory Address	Location of the next word in memory to be used for a data channel transfer.

PROGRAMMING

A microNOVA diskette subsystem is programmed for data channel transfers in three phases: I, the drive is selected; II, the head is positioned over the correct track; and III, the Read or Write operation is started. The program should check for errors before proceeding to the next stage.

In particular, the program should check for the Not Ready condition before issuing any commands to the drive, since the drive can be in the Not Ready state for a variety of reasons, e.g., the door is open or the drive has not acheived operating speed. In the following descriptions of the programming phases, it is assumed that this check is made before each command is issued. It is also assumed that the head is in contact with the media (Head On flag is 1) before any Read Preamble or data transfer operation is attempted.

Each time a command is issued to a drive, the unit number of the drive must be specified to prevent drive deselection during command sequences.

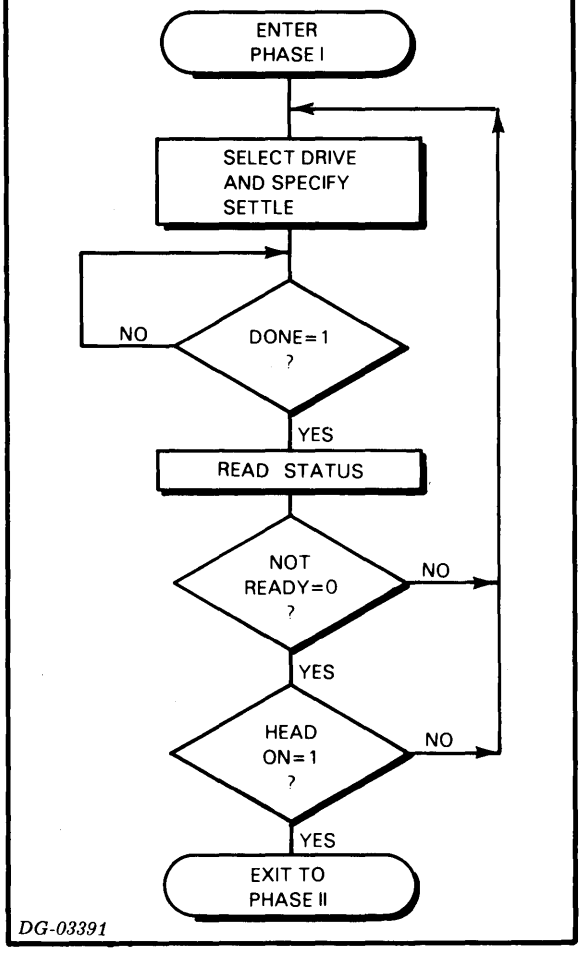
CAUTION *The only valid method of executing a command is by appending the Start pulse to the SPECIFY COMMAND instruction (DOA). The mnemonic for this command, DOAS, is used throughout the following description of programming. A Start command issued in any other manner will initiate a bootstrap loading procedure. The Start command sets the Busy flag to 1 and the Done flag to 0. Under no circumstances should the program issue a Start command when the Busy flag is set to 1.*

PHASE I: SELECT THE DRIVE AND DETERMINE STATUS

The program issues a DOAS instruction to the controller to select the drive unit and specify a 10ms Settle operation. The drive units are numbered 0 and 1 and the sectors are numbered 0-7. Although the desired sector number can also be specified in this instruction, it is not needed at this stage. Upon completion of this settle operation, the Busy flag is set to 0 and the Done flag is set to 1, thus initiating a program interrupt request. (This operation should be repeated to initialize the electronics any time a drive change is performed.)

Once the drive unit is chosen, its status must be checked to determine if it is ready to proceed. The program should issue a READ STATUS instruction (DIA) and examine the drive Not Ready and the Head On flags. If the Not Ready flag is set to 0, and the Head On flag is set to 1, the program can proceed to Phase II. If any other result is obtained, the program should continue to execute Settle operations and to check the results.

INITIAL SELECTION (Phase I)



PHASE II: POSITION THE HEAD

The head can be positioned over the desired track in two ways. First, the program can determine the current track address under the head by reading it from the address field preceeding any sector of the diskette. (If desired, the program can keep a record of the track address and continually update it.) With this information, the program can determine the direction in which the head must be moved and the number of steps it will take to position it over the desired track. This method permits rapid access to any track.

The second means by which the program can position the head is by performing a recalibrate operation before each track access. With this method, which guarantees positional accuracy, the program need not maintain a record of the current head position.

To recalibrate the drive unit, the program must issue a DOAS instruction specifying a Settle operation. Following the 10 ms settling time, the Busy flag is set to 0 and the Done flag is set to 1, thus initiating a program interrupt request. Then, a READ STATUS instruction (DIA) is issued; if the Not Ready flag is 0 and the Head On flag is 1, the program should examine the Track 0 flag. If the Track 0 flag is 1, the head is at the home or recalibrated position (track 00). If the Track 0 flag is 0, a series of Step-out commands must be executed.

A Step-out operation moves the head from its current track position to the next lower numbered track. The operation is initiated with a DOAS instruction specifying a Step-out command field with a Start command field. Upon completion of this operation, the Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated. The program should re-examine the Track 0 flag through the use of a READ STATUS instruction. If the Track 0 flag is 0, the program should repeatedly execute Step-out operations and check the results after each one until the Track 0 flag is set to 1.

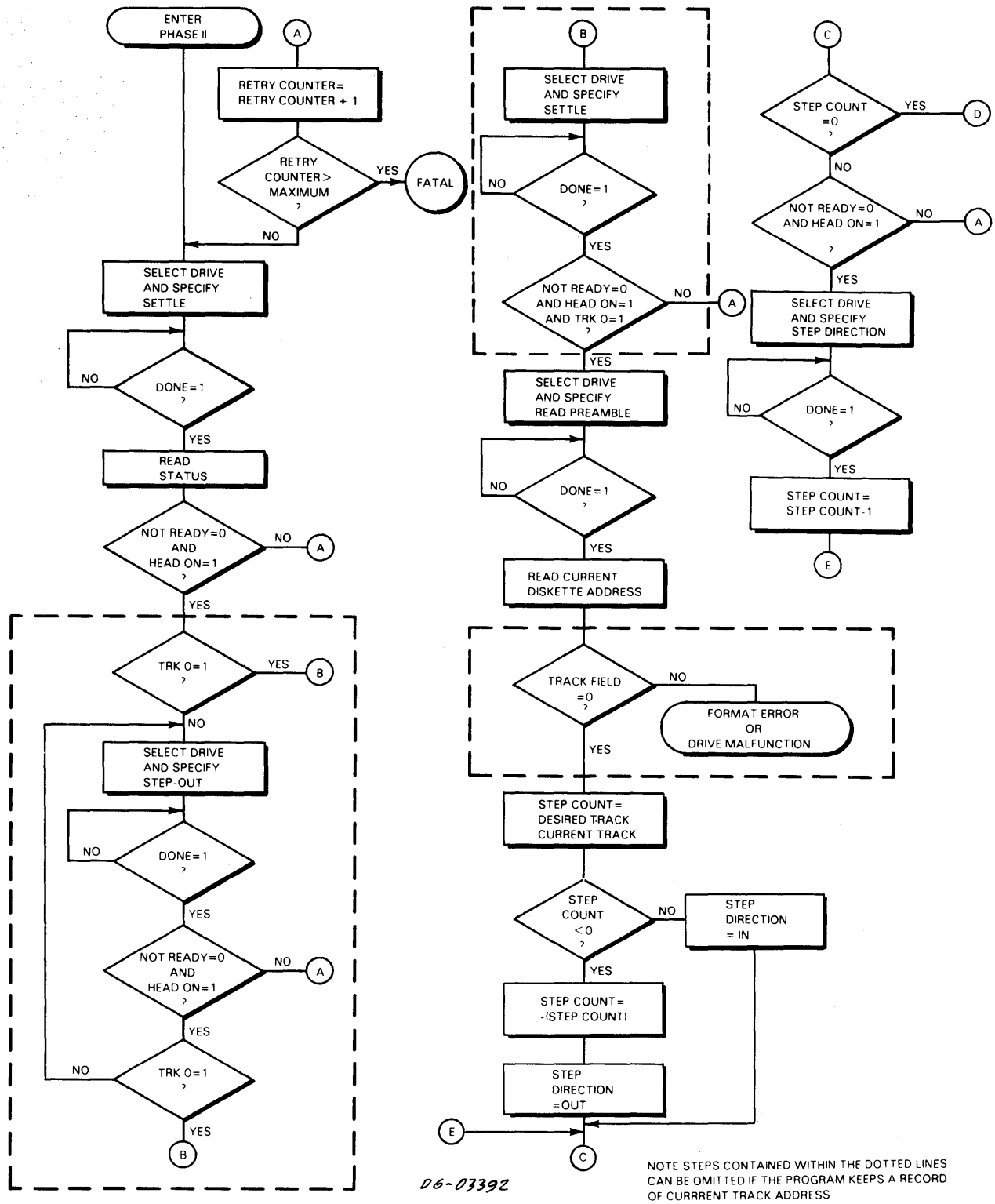
CAUTION *The program should never execute a Step-out operation when the drive is at the home position. Possible drive misalignment may result.*

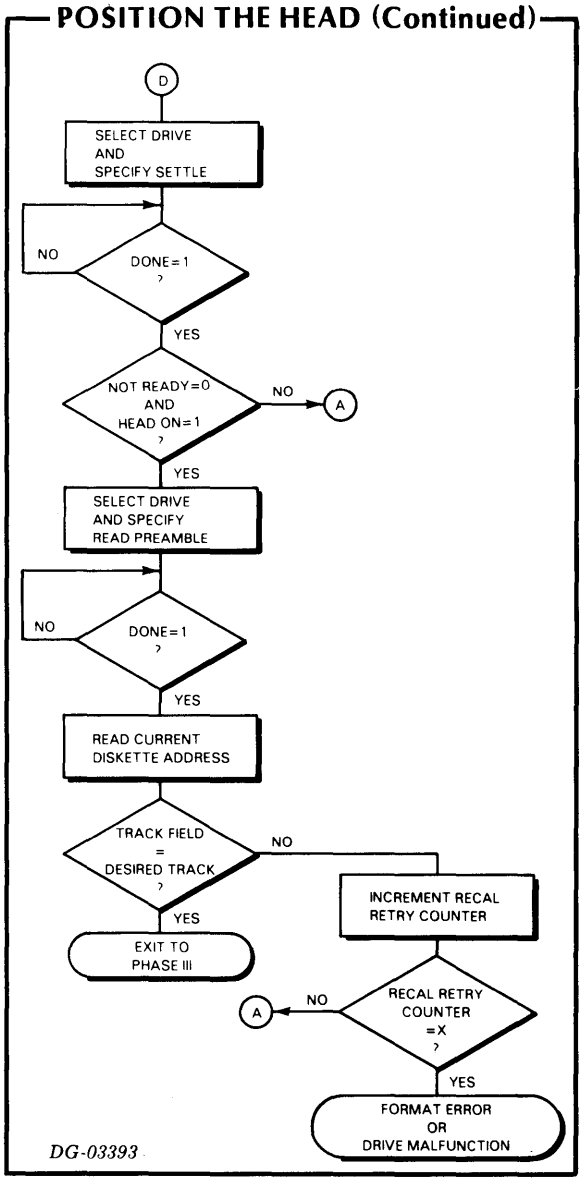
Next, the program must determine the number of tracks the head must move until it is over the correct track. For each track the head must traverse, a head positioning operation must be performed. Each head positioning operation is performed in the same way. A Step In command is issued to the controller with a DOAS instruction which specifies a Step-in command field. Upon completion of the movement to the next track, the Done flag is set to 1, thus initiating a program interrupt request. Successive head movement operations are performed until the desired track is reached.

Having reached the desired track, the program should ensure that all head movements are stopped by performing a Settle operation. Once this operation is completed, the program should check the track address by issuing a Read Preamble command as follows: A Specify Command instruction (DOAS) is issued, specifying a Read Preamble operation. When the preamble has been read, the Busy flag is set to 0,the Done flag is set to 1, and a program interrupt request is initiated. The results of the Read Preamble operation can be examined by the program through the use of a Read Current Diskette Address instruction (DIC). Both the track and sector numbers of the first sector which passed under the head, when the operation was performed, are returned. Following the completion of this operation, the program should issue a READ CURRENT DISKETTE ADDRESS instruction to determine if the desired track has been reached. If the track address read is the desired one, the program can proceed to Phase III. If not, the program should perform a recalibrate operation and try again.

CAUTION *The program should never execute a Step-in operation which would move the head beyond track 114_8.*

POSITION THE HEAD (Phase II)





PHASE III: READ OR WRITE

Once the head is positioned over the correct track, the program must determine over which sector on that track the head currently resides. Current sector position can be determined by examining the sector field returned by the last READ CURRENT DISKETTE ADDRESS instruction issued in Phase II.

The sector which must be under the head before attempting to read or write any data is the sector just before the desired sector. Thus if the program wishes to transfer data to or from sector 6, sector 5 must be under the head before any data transfer operation is initiated. The program should execute Read Preamble operations (by issuing a DOAS instruction) and check the results of each operation (after the operation is completed) by issuing READ CURRENT DISKETTE instructions until the sector immediately before the desired sector resides under the head. When this situation occurs, the data transfer operation can be executed.

Read

A LOAD MEMORY ADDRESS REGISTER instruction (DOB) specifies the location in memory into which the first word is to be read from the diskette.

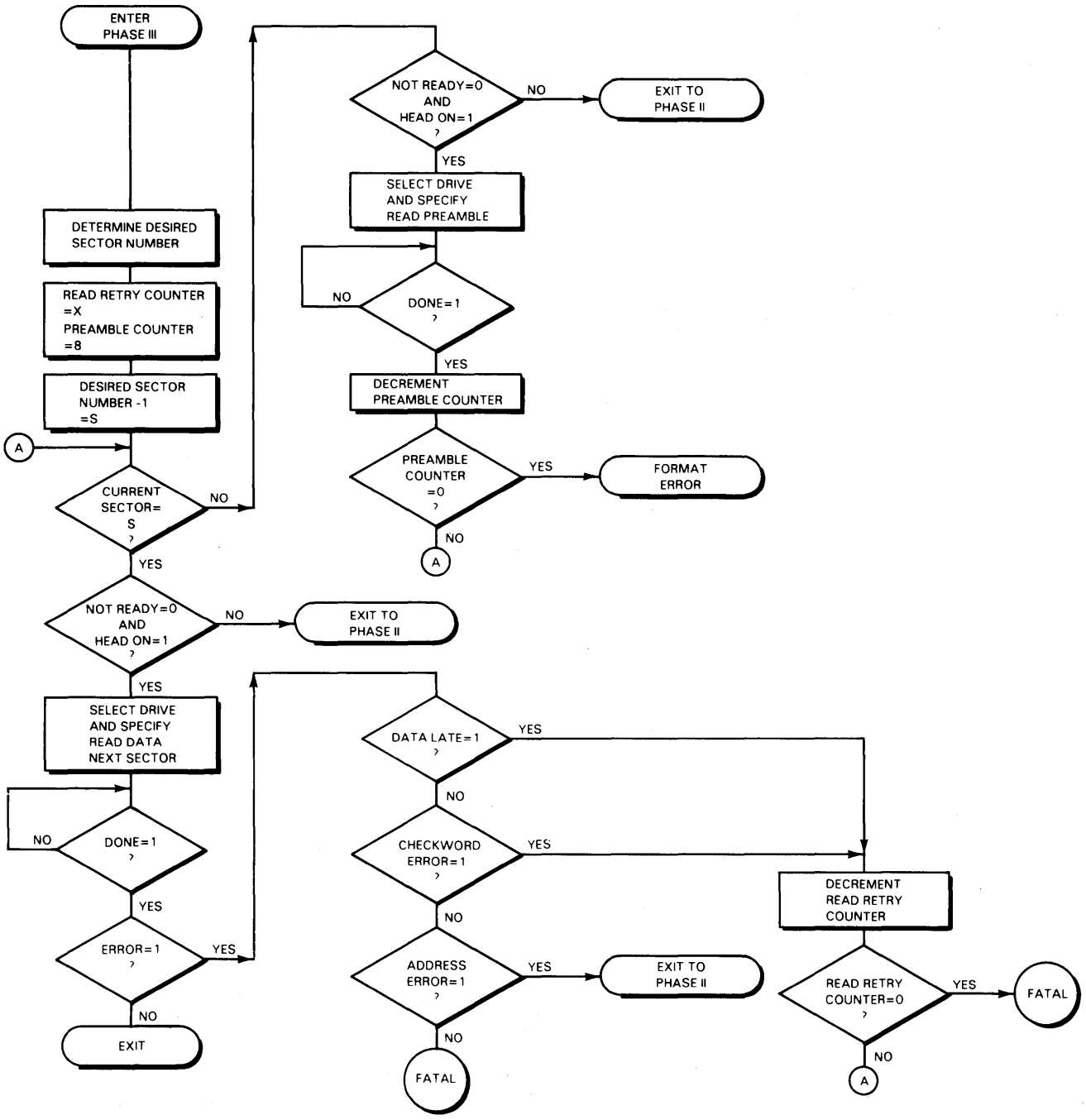
The Read Data, next sector command is loaded into the Command Register with a SPECIFY COMMAND instruction which specifies a Read Data, next sector command field (DOAS). The sector number of the desired sector must also be specified since it is used in the address check which is automatically made before each 256 word sector is read. The unit number of the drive must also be specified to prevent a deselection of the desired drive. The Busy flag is set to 1 and the Done flag is set to 0.

Once the Read operation is initiated, the drive waits until the next sector passes under the head and performs an address check. If the sector address read matches the address specified, the drive then starts reading the sequential bits of the first word in the sector. When a word is fully assembled in the controller's serial buffer register, the controller transfers the word to the computer's memory via the data channel. Each time a word is transferred to memory, the Memory Address Counter is automatically incremented.

When the 256 words from the sector have been read, the checkword at the end of the sector is compared with the checkword calculated from the data. Upon completion of this read operation, the Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated.

Sufficient time is available after a sector has been read to specify the parameters of a Read operation for the next consecutive sector on the diskette. Thus contiguous sectors can be read without waiting for the execution of a full sequence of commands providing the sectors all have the same track address.

READ (Phase III)



Write

A LOAD MEMORY ADDRESS REGISTER instruction specifies the location in memory from which the first word is to transferred to the diskette.

The Write Data, next sector command is loaded into the command register with a SPECIFY COMMAND instruction (DOAS). The sector number of the desired sector must be specified in this instruction since it is used in the address check which is automatically made before each 256 word sector is written. The unit number of the drive must also be specified to prevent deselection of the desired drive. If the track number of the track on which the data is to be written is greater than 43(sub10), the Hi Trk bit must be set to 1. The Busy flag is set to 1 and the Done flag is set to 0. If the Hi Trk bit is not set to 1, difficulty may be encountered later when attempting to write over that sector.

Once the Write operation is initiated, the controller reads a word from the computer's memory via the data channel and then waits for the next sector to pass under the head. Each time the controller reads a word from the computer's memory, the Memory Address Counter is automatically incremented. Once the next sector is encountered, the address check is performed automatically. If the sector address read matches the address specified, the bits of each word are sequentially written. When the 256 words in the sector have been written, the controller writes the checkword it calculated from the data written in that sector on the diskette.

The Write operation is completed when the next sector mark is encountered by the drive, thus preventing consecutive sectors from being written. Upon completion of the Write operation, the Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated.

TIMING

The diskette rotates at a speed of 360 rpm; a complete revolution requires 166.7 milliseconds.

Since a sector is preceded by a 1.4 millisecond gap used for address checking, the minimum sector access time is 1.4 milliseconds. The maximum is 167 milliseconds; the average is 84 milliseconds.

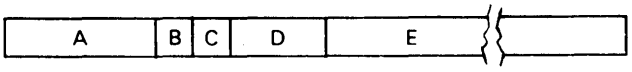
The time required to position the heads is dependant on the number of tracks which must be traversed. It takes 10 milliseconds to step the head from one track to the adjacent track. Head settle time after completion of the last head step requires 10 milliseconds.

Head loading time depends on whether or not the drive unit was selected for more than one revolution before the first command was issued to that drive. If the drive unit had been selected for more than 170 milliseconds, the head load takes a minimum of 60 milliseconds. If the drive unit was just selected, the minimum head loading time is 170 milliseconds. All head loading operations can overlap head stepping operations. This allows programs which maintain records of current track position to move to a new track while loading the head. The head will automatically unload if no commands are issued to the diskette unit for 170 milliseconds or if the drive is deselected.

A sector passes under the head in 20.8 milliseconds; while the data block in the sector passes under the head in 16.4 milliseconds. Since there are 256 data words in a sector, a data channel request occurs every 64 microseconds. This corresponds to a data transfer rate of 15,625 words/second. Since the controller has a one-word buffer, the maximum allowable data channel latency is 64 microseconds. If the data channel does not respond within this time, both the Data Late and the Error flags are set to 1.

FORMAT

The diskette used in the diskette subsystem must be formatted so that the current disk address can be read and the address check, made before every sector is read or written, can be carried out. If the diskette is not formatted, the program cannot determine current sector position and an address error terminates every read or write data operation. When the diskette is formatted, the following information is written before every sector on the diskette.



Area A is a synchronization field of 22 words (all zeros); area B contains the sync bit (formatted as a 16-bit word containing the number 1); area C contains the address field as expressed in the AC format for the READ CURRENT DISKETTE ADDRESS instruction (DIC), with all unused bits set to 0; area D contains a second synchronization field of 12 words (containing all zeros); and area E contains 220 words, the contents of which are immaterial.

To format a diskette, the program should set up a data block arranged in the above manner and transfer the block to the subsystem. The command field in the DOAS instruction should specify Format Sector 0 for the first sector on a track. Upon completion of this operation, the sector address in the data block should be incremented and the Format Next Sector command executed. Repeat until all sectors on a track have been formatted. Repeat the entire procedure for the next track after moving the head to that track. Data General Corporation supplies the the 6038/6039 Diskette Formatter Program (absolute binary tape no. 095-000443 and listing no. 096-000443) which performs the formatting necessary for diskettes. This program insures compatibility between microNOVA diskettes and NOVA/ECLIPSE line diskettes.

ERROR CONDITIONS

During Initial Selection

The Not Ready flag and the Head On flag are the only status flags of interest during Phase I. The drive unit can be in the Not Ready state for a variety of reasons, e.g., the door is open or the drive is not up to proper operating speed. This condition can occur at any time in the succeeding phases and should be checked at every stage of diskette programming.

During Head Positioning

Since the program must keep the count of the number of tracks being traversed during head positioning, care must be taken to ensure that the count is correct. If the drive goes to a Not Ready state and then returns to the ready state due to an operator opening and closing the door, the program may miss the Not Ready indication. The program should check the Not Ready Latch bit before each head positioning command to determine if any such situation has occurred. (This could occur if the operator changed diskettes.) If no data is received from the diskette during a Read Preamble operation, the Error flag is set to 1 and, in most cases, the Address Error flag is set to 1. The Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated. This situation normally occurs only if the head is not in contact with the diskette during the Read Preamble operation.

During Reading

Specifying a non-existent sector (i.e., greater than 7) will not cause any error indications since the sector field is determined only by the three low order bits specified.

Each sector on the diskette is preceeded by an address field. If the address read before the sector does not match the address specified with the Read command, both the Address Error flag and the Error flag are set to 1. The Busy flag is set to 0, the Done flag is set to 1 and a program interrupt request is initiated.

If the checkword read at the end of the sector differs from that calculated by the controller, both the Error and the Checkword Error flags are set to 1. The Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated. The Checkword Error flag indicates that at least one word in the sector contains an error.

If the data channel does not respond in time to a data channel request, both the Error and the Data Late flags are set to 1. The reading of the current sector continues, but once the sector has been read, the read operation is terminated. The Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated. The Data Late flag indicates that at least one word from the sector was not correctly transferred to memory.

During Writing

Specifying a non-existent sector (i.e., greater than 7) will not cause any error indications since the sector field is determined only by the three low order bits specified.

Each sector on the diskette is preceeded by an address field. If the address read before the sector does not match the address specified with the Write command, both the Address Error flag and the Error flag are set to 1. The Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated.

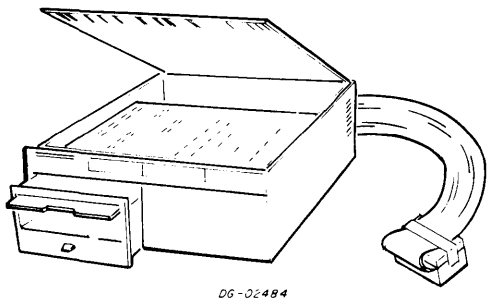
If the data channel does not respond in time to a data channel request, both the Error and the Data Late flags are set to 1. The writing of the current sector continues and on completion, the Write operation is terminated. The Busy Flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated. The Data Late flag indicates that at least one word in the sector was not written properly.

If the program attempts to write data on a diskette which is write protected (the Write Protect bit is 1), no indication of the error is returned. The operation does not occur, however. The Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated.

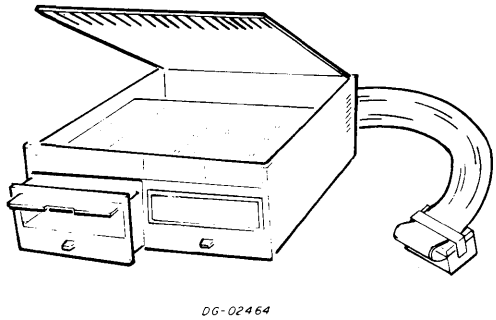
Illegal Indications

The Illegal flag is generally set to 1 when an undefined field is supplied with a DOAS instruction. In all cases of undefined command fields, the operation is terminated; the Busy flag is set to 0, the Done flag is set to 1, and a program interrupt request is initiated.

ONE DRIVE DISKETTE SUBSYSTEM WITH CABLE



TWO DRIVE DISKETTE SUBSYSTEM WITH CABLE



DEVICE CODE JUMPERS

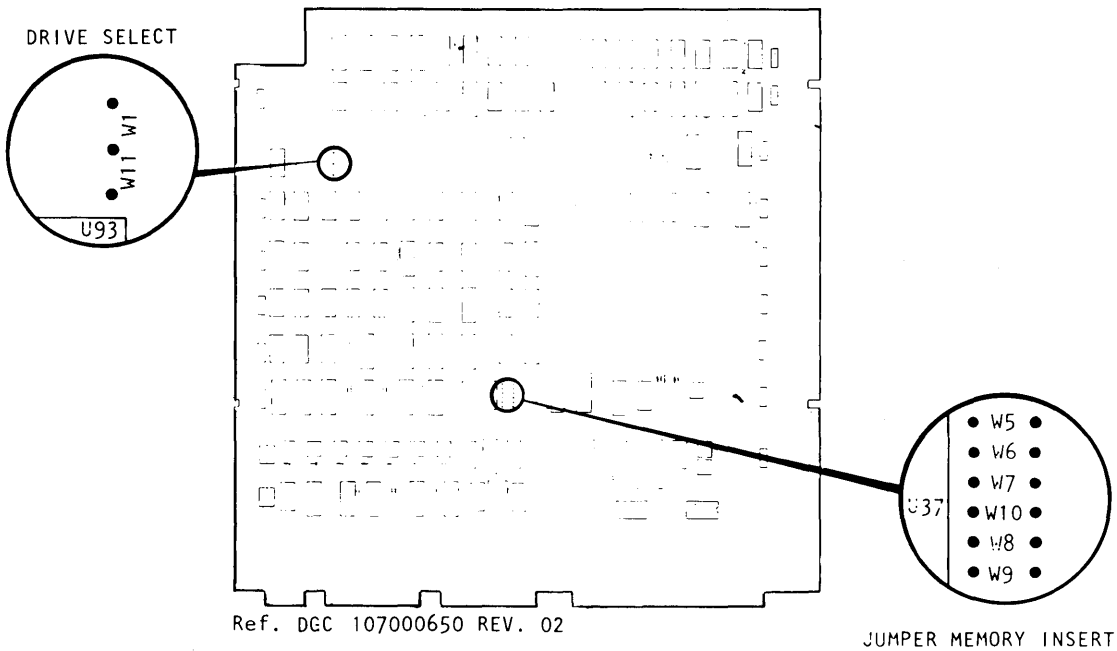
BIT POSITIONS OF DEVICE CODE	0	1	2	3	4	5
INSERT JUMPER TO SPECIFY 1	W10	W9	W8	W7	W6	W5
FIRST DEVICE CODE	OUT	IN	IN	OUT	IN	IN

DRIVE 0 JUMPERS

DRIVE 0	INSERT JUMPER
LEFT DRIVE	W1
RIGHT DRIVE	W11

NOTE *The settings of jumpers W2, W3, W4, W12, and W13 are ignored.*

DISKETTE BOARD JUMPER LOCATIONS



Ref. DGC 107000650 REV. 02

DG-03555

JUMPER MEMORY INSERTED
FOR DEVICE CODE 33

W10	W9	W8	W7	W6	W5
OUT	IN	IN	OUT	IN	IN

DRIVE 0 IF SELECTED BY INSTALLING JUMPER W1 OR W11. IF ONLY ONE DRIVE IN SYSTEM, JUMPER W11 SELECTS DRIVE 0.

SECTION VIII
PROM PROGRAMMER BOARD

Data General's model 8574 PROM programmer is an IOC-based controller capable of programming Signetics 82S126 (256x4 bits) and 82S130 (512x4 bits) PROM chips in an online mode when inserted in the sockets of a microNOVA PROM memory board. A full PROM board containing 512, 1,024, 2,048, or 4,096 16-bit words of PROM can be programmed and verified, one word at a time, under program control, using the PROM programmer. Programming a bit to 1 takes approximately 20 milliseconds, while programming a bit to 0 takes approximately 400 nanoseconds.

OVERVIEW AND INTRODUCTION

Any computer system that contains read-only memory (ROM) must have some method for initializing the contents of that memory. One method is to purchase ROM with the contents of the memory already initialized by the manufacturer. Another method is to purchase programmable ROM (PROM) and program the memory. Once a bit has been programmed to contain a value different from the value to which it was initialized, the value of that bit can no longer be changed.

The microNOVA computer system includes a PROM programmer board. The PROM programmer board provides the capability to program PROM, one word at a time, under program control. A PROM board is programmed by inserting the PROM board in the female board connectors of the PROM programmer board, and issuing the appropriate I/O instructions to the PROM programmer board to program and verify the contents of the PROM.

SUMMARY OF CHARACTERISTICS

Applicable PROM Types	Signetics 82S126 and 82S130.
Board Dimensions	7.5 x 13.75in (19 x 34.925cm)
Maximum Operating Temperature	131 F (55 C)

BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components plus data and control paths of the microNOVA system PROM programmer. It comprises an IOC and support circuits, a 16-bit data register, a 16-bit verification driver (non-latching), a 12-bit address register, and two banks of reed relays for driving the programming voltage to a PROM board. The block diagram on the following page shows a PROM board connected to a PROM programmer board.

The IOC, its clock driver, the function decoder, the I/O bus transceiver and the bus control buffer interface the PROM programmer to the I/O bus of a microNOVA CPU. The protocols by which the CPU communicates with peripherals is detailed in the Technical Reference for microNOVA Integrated Circuits *(DGC no. 014-000031)*.

Data is routed in bit parallel fashion out the IOC to latch in the address or data register, or it is routed from the data out lines of the PROM board through the verification driver into the IOC.

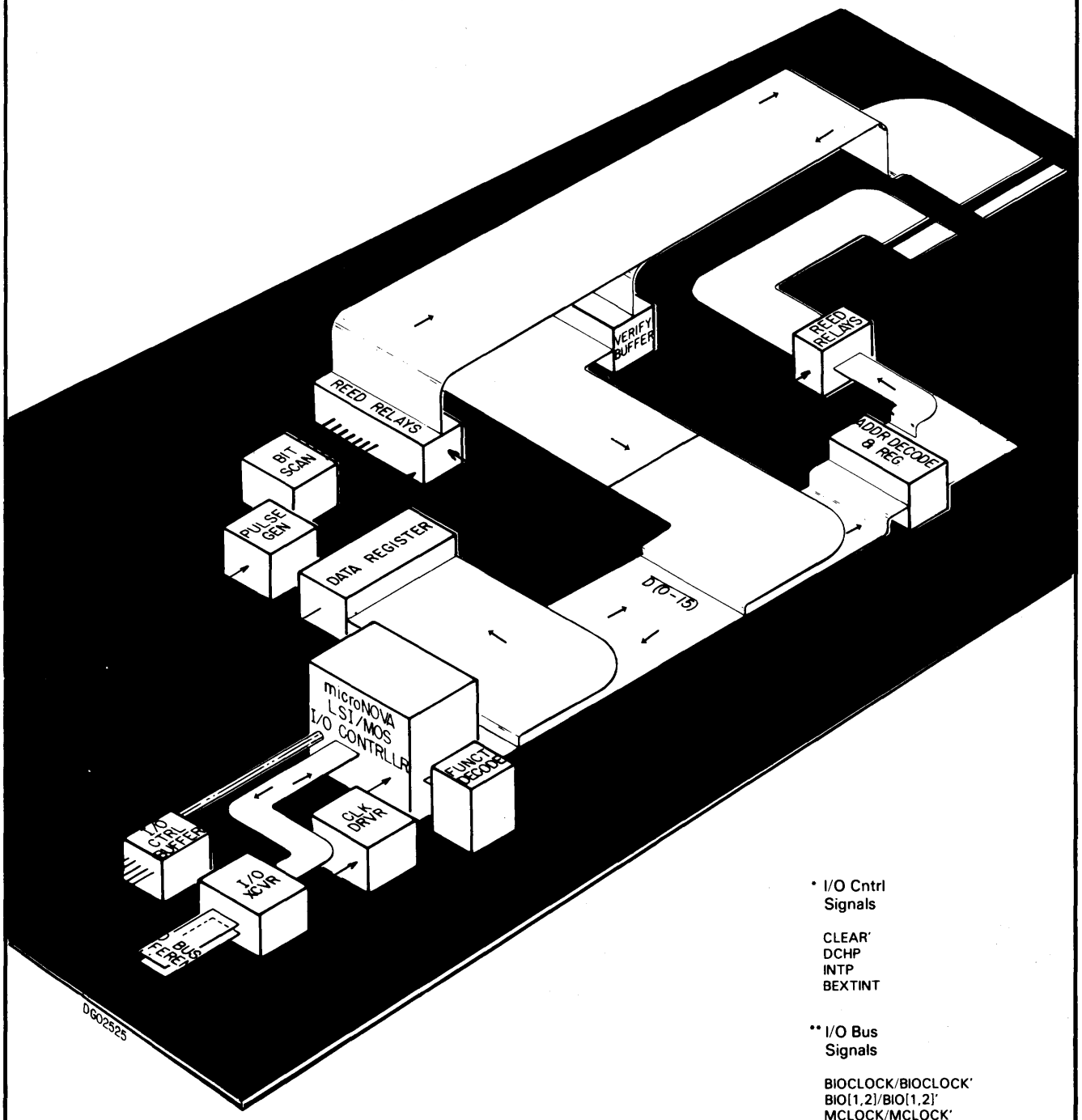
NOTE *When programming a PROM board, the address selection jumpers on that board must be removed entirely; this prevents conflicts between address decoders in the programmer and in the ROM module. When this is done, all of memory locations on the board to be programmed lie in the range 000000 through the number of memory locations on the board.*

The contents of the data register are shifted left one bit at a time while the line scanner can pulse each reed relay to apply a programming voltage to each bit in the addressed PROM location. If a 1 is shifted out of the data register, programming sequence is started. If a 0 is shifted out of the data register, no sequence is started and the next bit is shifted out.

When a programming operation begins, the high-order bits of the address register select and enable one of eight banks of PROM on the PROM board by applying a high level programming voltage to the Vcc input of that bank, a high level output voltage to the bit to be programmed, and a bank enable pulse through input connector "A" of the PROM board; simultaneously, the entire 12-bit address is passed to the memory input latch on the PROM board. The low-order bits select an address within the bank that will be programmed.

After the last bit is shifted out of the data register, the contents of the address register continue to select the word in PROM that was programmed for an additional 5ms. During that time the contents of the addressed word remains on the PROM board data out lines, at the input to the verification transmitter.

microNOVA PROM PROGRAMMER BOARD



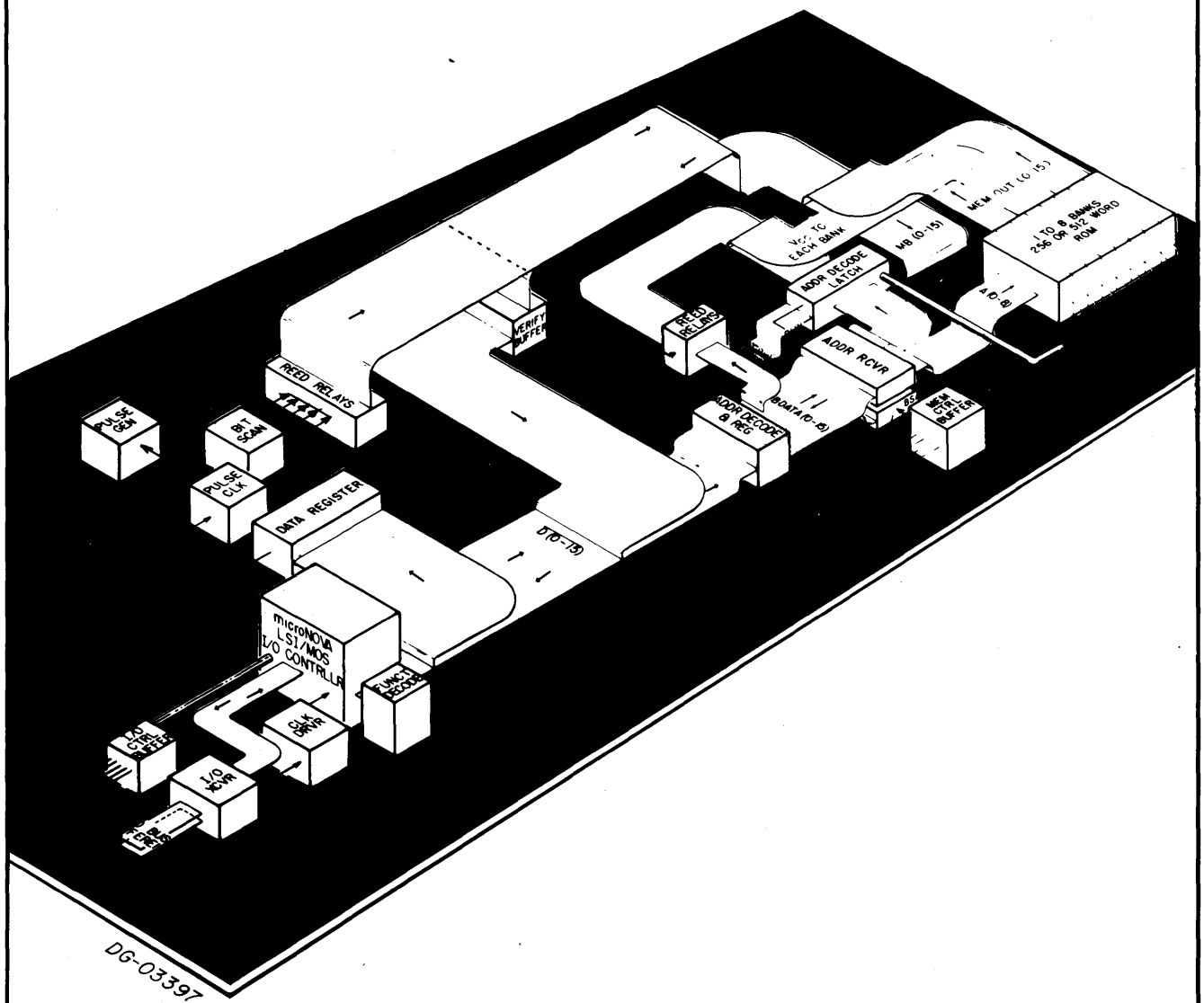
* I/O Cntrl Signals

CLEAR'
DCHP
INTP
BEXTINT

** I/O Bus Signals

BIOCLOCK/BIOCLOCK'
BIO[1,2]/BIO[1,2]'
MCLOCK/MCLOCK'

microNOVA PROM PROGRAMMER BOARD CONNECTED TO PROM BOARD



PROGRAMMING

The microNOVA PROM programmer board provides a method for online programming of PROM's on a module-by-module basis, without resorting to time-consuming programming on a chip-by-chip basis as with other PROM programmers. To achieve this goal, the PROM programmer gives control over the address to be programmed and the data to be written into that location. The programmed location can be read back so that one can verify whether or not the location was written correctly.

To program a location, specify the address and load the programming buffer. The ordering of these instructions is unimportant. Then, direct the PROM programmer to write the information into the location by giving a Start command with the second instruction. Programming of a 16-bit word takes approximately 20 milliseconds for each 1 in the word, and approximately 400 nanoseconds for each 0. After the PROM programming cycle is complete, the PROM programmer will set Done and allow 5 milliseconds to read the location just written. This gives an opportunity to verify that the location programmed correctly.

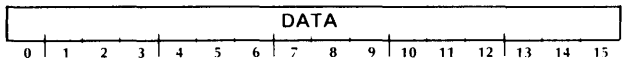
To verify a location without changing its contents, go through a complete PROM programming cycle with an all-zero data word and then read the contents of the location.

PROGRAMMING SUMMARY

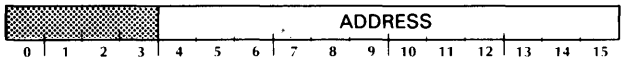
Mnemonic	PROG
Device Code	5
Priority Mask Bit	15

ACCUMULATOR FORMATS

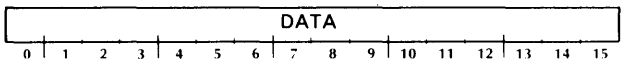
Load Programming Buffer (DOA)



Specify Address (DOB)



Verify (DIA)



The device flag commands control the PROM programmer's Busy and Done flags in the following manner:

f=S Set the Busy flag to 1, set the Done flag to 0, and initiate a PROM programming cycle.

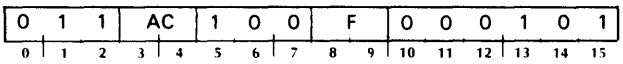
f=C Set the Busy flag and the Done flag to 0.

f=P Set the Done flag to 0.

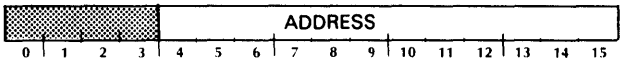
Instructions

Specify Address

DOB *[f]* *ac*, **PROG**



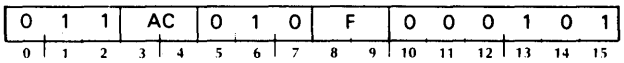
Bits 4-15 of the specified AC are placed in the PROM programmer's address register. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



Bits	Name	Contents
0-3		Reserved for future use.
4-15	Address	Address of the PROM location to be programmed.

Load Programming Buffer

DOA *[f]* *ac*, **PROG**

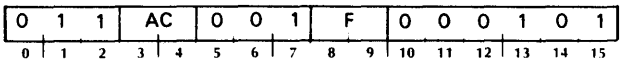


The contents of bits 0-15 of the specified AC are placed in the PROM programmer's data register. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:

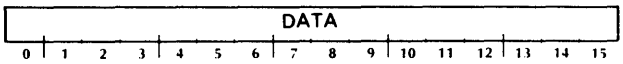
Bits	Name	Contents
0-15	Data	The information to programmed into the specified PROM location.

Verify

DIA *[f]* *ac*, **PROG**



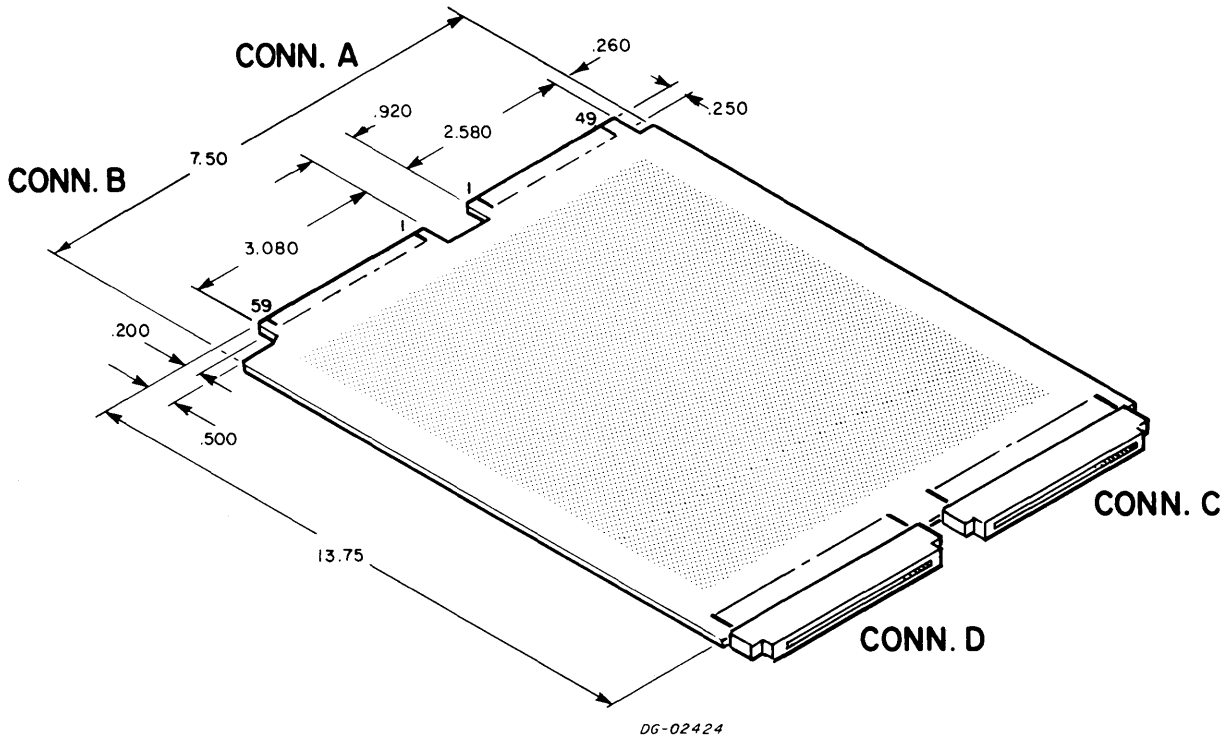
The contents of the most recently programmed PROM location are placed in bits 0-15 of the specified AC. After the transfer, the function specified by F is performed. The format of the specified AC is as follows:



Bits	Name	Contents
0-15	Data	Contents of the most recently programmed PROM location.

NOTE *For the data returned to be meaningful, this instruction must be issued within 5 milliseconds of the completion of a PROM programming cycle.*

PROM PROGRAMMER BOARD DIMENSIONS

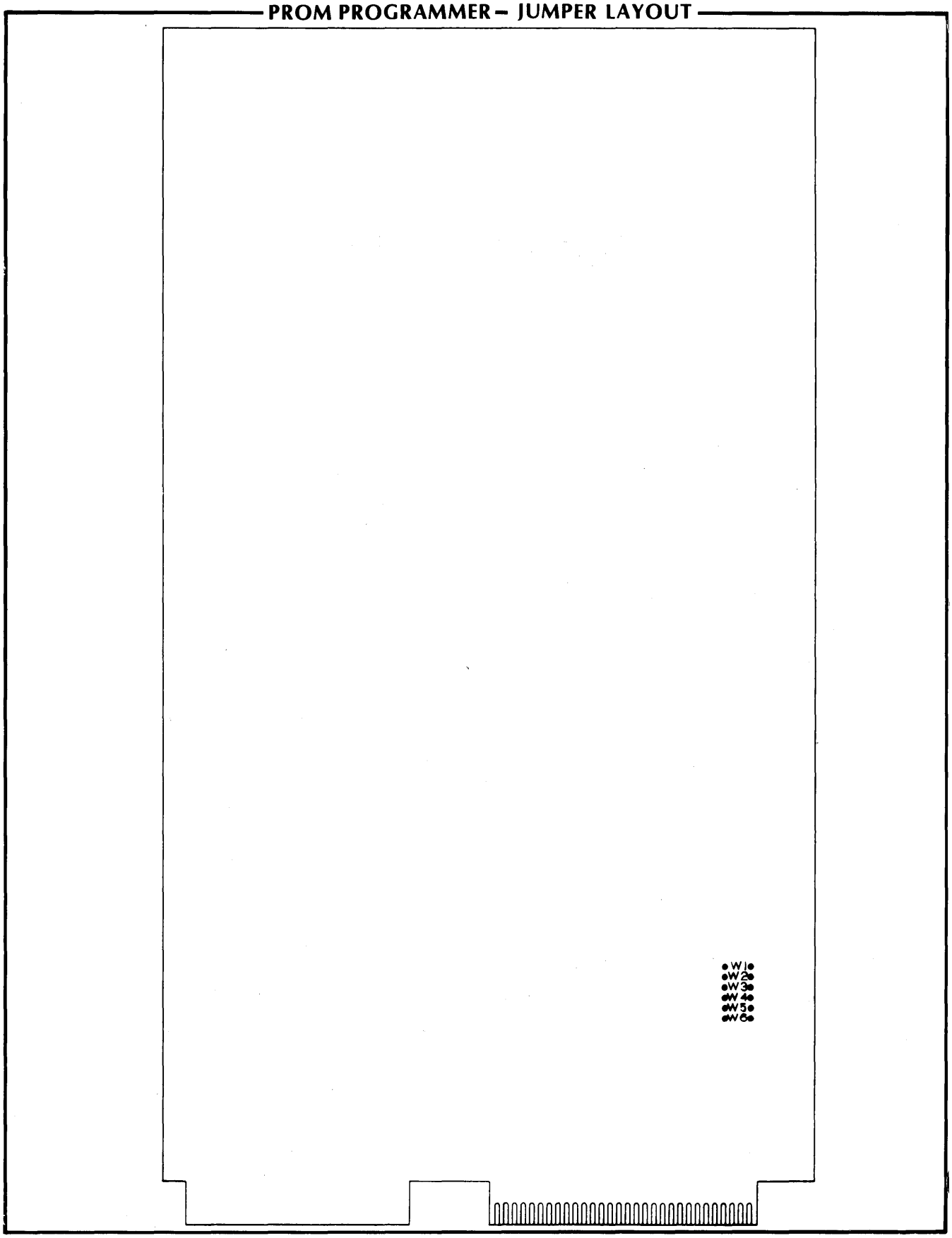


DEVICE CODE JUMPERS

BIT POSITIONS OF DEVICE CODE	0	1	2	3	4	5
INSERT JUMPER TO SPECIFY 1	W1	W2	W3	W4*	W5	W6*

* Normal Device Code

PROM PROGRAMMER – JUMPER LAYOUT



SECTION IX
GENERAL PURPOSE INTERFACE BOARDS

Data General's model 4210 general purpose interface boards is an IOC-based interfacing aid. The board contains an IOC module capable of providing a generalized front end to the microNOVA I/O bus. The IOC module handles all Busy/Done/interrupt processing and data channel interfacing. Model 4210 has pre-drilled holes and etched conductors, to assist in the building of an interface. The Model 4210 with the model 4211 option has DIP sockets and wire wrap pins installed in the pre-drilled holes.

OVERVIEW AND INTRODUCTION

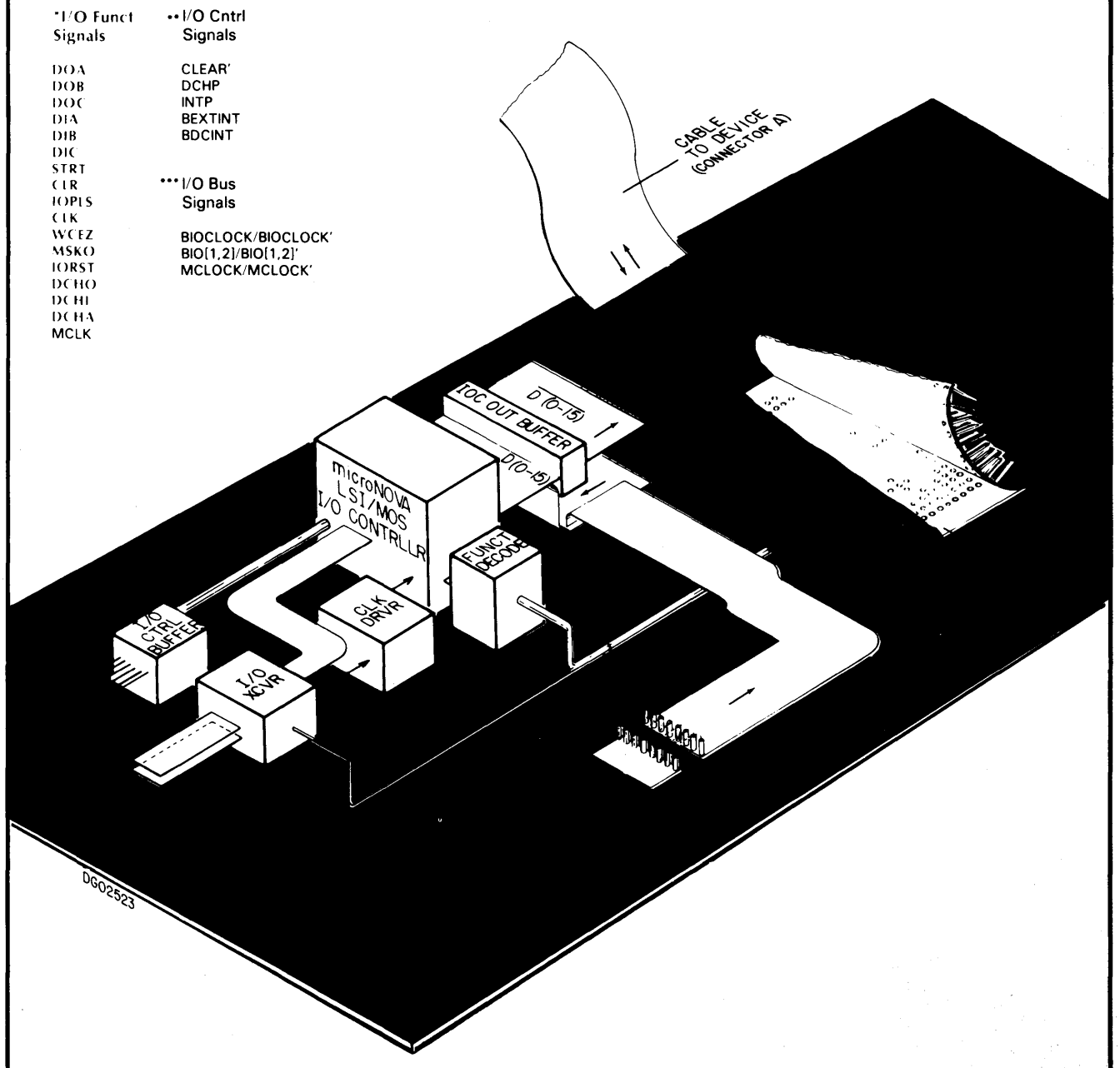
Microcomputers are used in a wide variety of applications from interactive information retrieval systems to systems for the monitoring and control of real-time processes. In many of these applications, it is necessary for the customer to interface non-standard equipment to the microcomputer. If no I/O device controller is available for a particular piece of equipment, the customer is forced to design his own controller.

Much of the logic contained in an I/O device controller is the same for every I/O device. This common logic can be designed once, laid out on a printed circuit board, and then used by customers in the construction of their own I/O device controllers.

A general purpose interface board is included in Data General's microNOVA family of computer assemblies. This board contains those logic elements that are common to most I/O device controllers, including an IOC and an I/O bus transceiver from the microNOVA family of integrated circuits. Drilled holes or sockets and wire wrap pins are provided for the customer to add those elements and interconnnections that are needed to control his particular equipment.

SUMMARY OF CHARACTERISTICS

Operations	I/O command decoding, interrupt and data channel requests, and data channel transfers.
Registers	15-bit data channel address 16-bit data channel word count Data channel request flag Interrupt request flag Interrupt disable flag External register enable flag Polarity flag Device code register Busy flag Done flag
Buses	I/O bus same as CPU.
Board Dimensions	7.5 x 10.4in. (19 x 26cm.)
Maximum Operating Temperature	131 F (55 C)


*I/O Funct Signals
DOA
DOB
DOC
DIA
DIB
DIC
STRT
CLR
IOPLS
CLK
WCEZ
MSKO
IORST
DCHO
DCHI
DCHA
MCLK
**I/O Cntrl Signals
CLEAR'
DCHP
INTP
BEXTINT
BDCINT
***I/O Bus Signals
BIOCLOCK/BIOCLOCK'
BIO[1,2]/BIO[1,2]'
MCLOCK/MCLOCK'
CABLE TO DEVICE (CONNECTOR A)
TOC OUT BUFFER
D(0-15)
microNOVA LSI/MOS I/O CONTRLLR
FUNCT DECODER
CLK DRVR
I/O CTRL BUFFER
I/O XCVR
DG02523

BLOCK DIAGRAM

The block diagram on the opposite page shows the principal components plus data and control paths for the microNOVA system General Purpose Interface (GPI). The GPI contains an IOC and its support circuits, plus a function decoder, and data buffer, plus sockets for device select and IOC initialization jumpers. All data lines, and control signals available for device use are brought to wire wrap pins. The I/O transceiver, IOC, clock driver and I/O control buffer connect the GPI to the I/O bus. The protocols by which the CPU communicates with peripherals are described in detail in the Technical Reference for microNOVA Integrated Circuits *(DGC no. 014-000074)*.

Sixteen bidirectional data lines (bit parallel) extend from the IOC in two groups. One group passes via a 16-bit buffer to wire wrap pins and normally is used for output signals from the CPU to customer designed circuits. The buffer provides TTL fanout capacity without overloading the IOC. The other group of data lines passes directly to wire wrap pins and normally is used for input signals from the customer circuit to the CPU. The input lines are designed to be driven by low-leakage, open-collector, TTL drivers.

The Interface Designer's Reference Manual, *(DGC no. 015-000031)*, provides information for the designer of special interfaces. Written chiefly to describe the I/O buses of Data General ECLIPSE and NOVA line computers, it contains detailed discussions about those I/O systems, and provides insight for the designer building special interfaces to them; since the IOC output signals emulate the the ECLIPSE and NOVA line I/O buses, the information in the interface manual will prove helpful to those designing microNOVA interfaces.

PROGRAMMING

The microNOVA general purpose interface board consists of an IOC module and space in which the rest of an I/O interface can be implemented. The IOC module contains a B register which can be loaded and retrieved and a C register which can be loaded. When the general purpose interface board is used to implement a data channel controller, the B register can be used as the Memory Address register and the the C register can be used as the Word Count register. Alternatively, the IOC module can directly control the loading and retrieving of up to three input and three output registers.

On the general purpose interface board, an I/O instruction sequence consists of asserting the appropriate I/O command line, performing the data transfer, and then performing the appropriate device flag command.

The device flag commands control the IOC module's Busy and Done flags in the following manner:

f=S Set the Busy flag to 1, set the Done flag to 0, and assert the STRT command line.

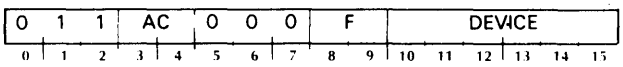
f=C Set the Busy flag to 0, set the Done flag to 0, and assert the CLR command line.

f=P Set the Done flag to 0 and assert the IOPLS command line.

Instructions

No I/O Transfer

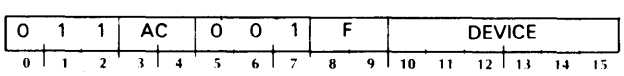
NIO *[f] ac,device*



The function specified by F is performed.

Data In A

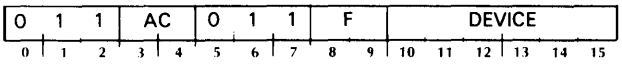
DIA *[f] ac,device*



The DIA command line is asserted and then the contents of the data lines are placed in the specified accumulator. After the transfer, the function specified by F is performed.

Data In B

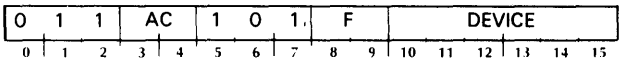
DIB *[f] ac,device*



The DIB command line is asserted and then the contents of the data lines are placed in the specified accumulator. After the transfer, the function specified by F is performed.

Data In C

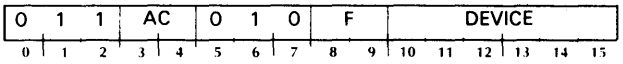
DIC *[f] ac,device*



The DIC command line is asserted and then the contents of the data lines are placed in the specified accumualtor. After the transfer, the function specified by F is performed.

Data Out A

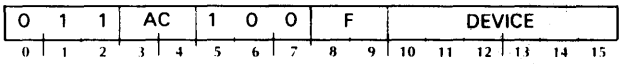
DOA *[f] ac,device*



The DOA command line is asserted and then the contents of the specified accumulator are placed on the data lines. After the transfer, the function specified by F is performed.

Data Out B

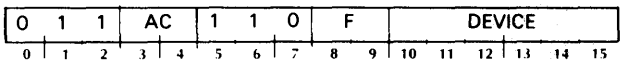
DOB *[f] ac,device*



The DOB command line is asserted and then the contents of the specified accumulator are placed on the data lines. After the transfer, the function specified by F is performed.

Data Out C

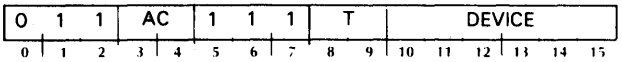
DOC *[f] ac,device*



The DOC command line is asserted and then the contents of the specified accumulator are placed on the data lines. After the transfer, the function specified by F is performed.

I/O Skip

SKP *[t] device*

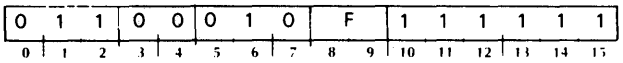


If the test condition specified by T is true, the CPU skips the next sequential instruction.

I/O Reset

IORST

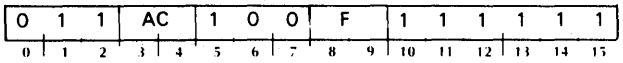
DOA *[f]* **0,CPU**



The Busy and Done flags in the IOC module are set to 0. The IOC module initializes its device code and logic polarity.

Mask Out

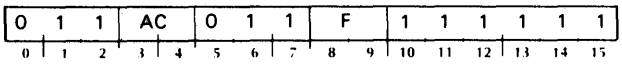
DOB *[f] ac,* **CPU**



The MSKO command line is asserted. Then the IOC module AND's the contents of its input data lines with the contents of the specified AC. If any of the result bits is a 1, the IOC module will not initiate an I/O interrupt request.

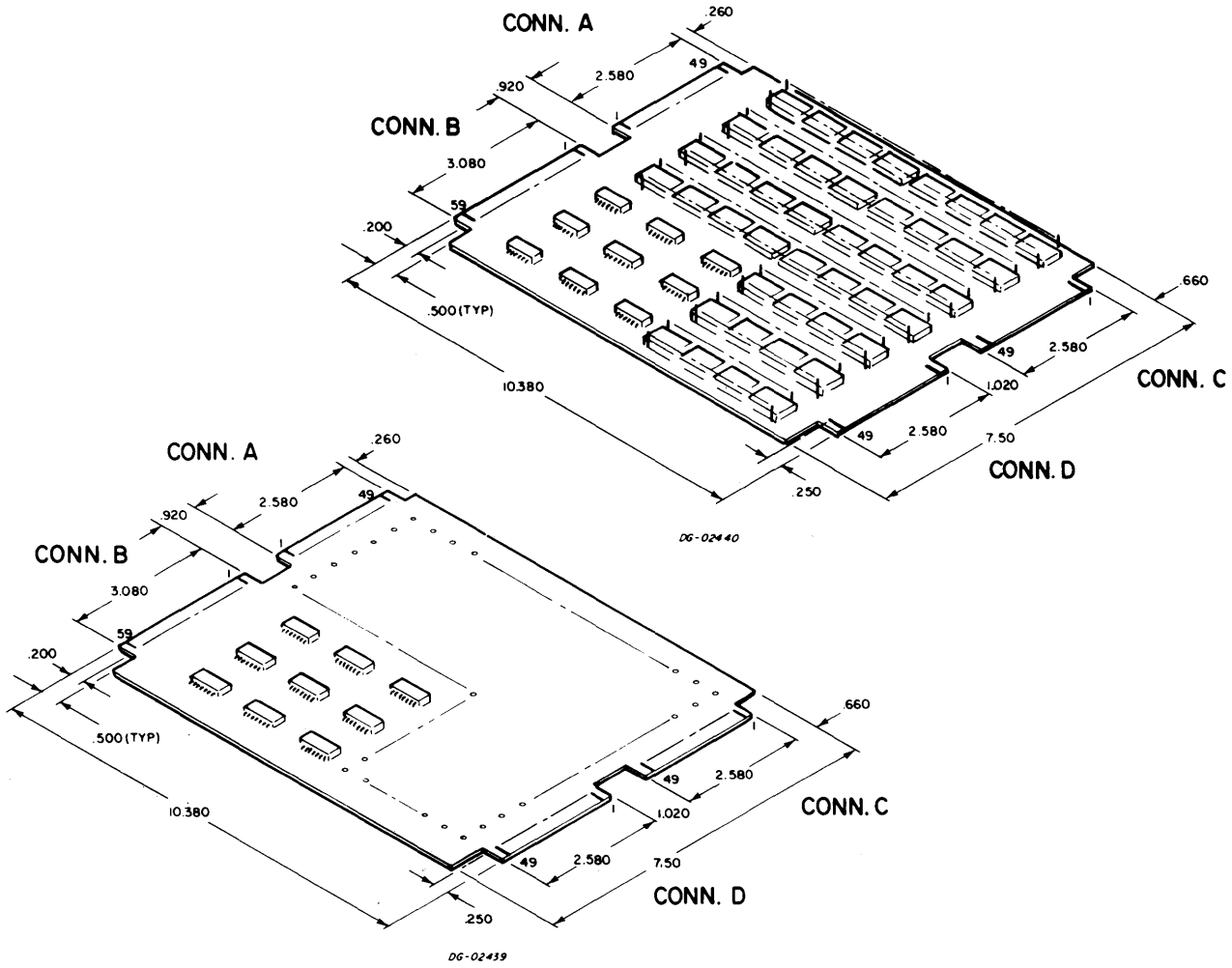
Interrupt Acknowledge

DIB *[f] ac,* **CPU**



The IOC module places its device code in the specified AC if it is requesting an interrupt and it is the highest priority device that is requesting an interrupt.

GENERAL PURPOSE INTERFACE BOARD DIMENSIONS

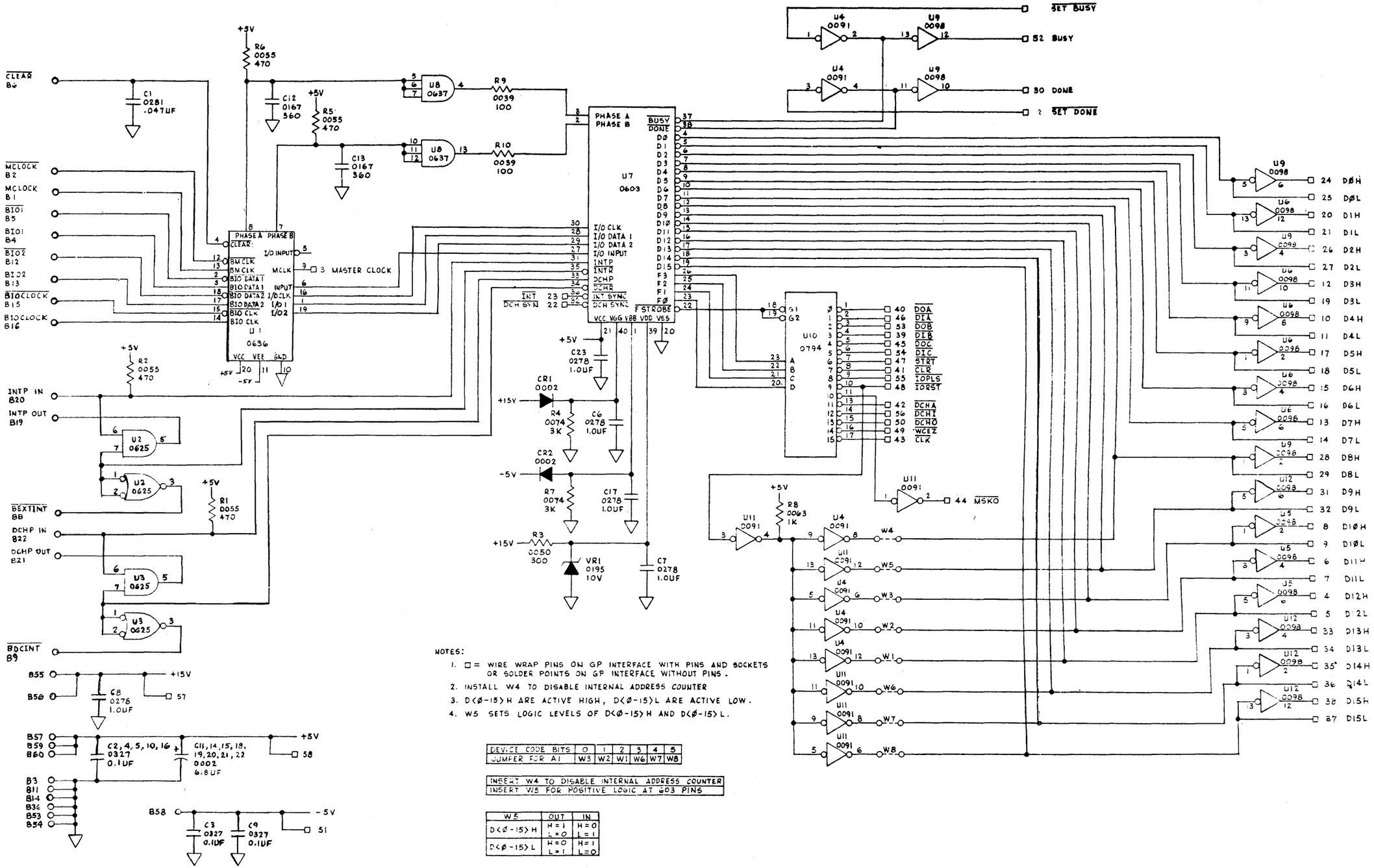


DEVICE CODE JUMPERS

BIT POSITIONS OF DEVICE CODE	10	11	12	13	14	15
INSERT JUMPER TO SPECIFY 1	W3	W2	W1	W6	W7	W8

OTHER JUMPERS

JUMPER	NAME	FUNCTION
W4	EXTERNAL	Omit jumper if the address and register internal to the IOC is to be used in the data channel operations and with the DIB instruction
W5	POLARITY	Insert jumper if positive logic is to be used in interfacing to the IOC.



NOTES:

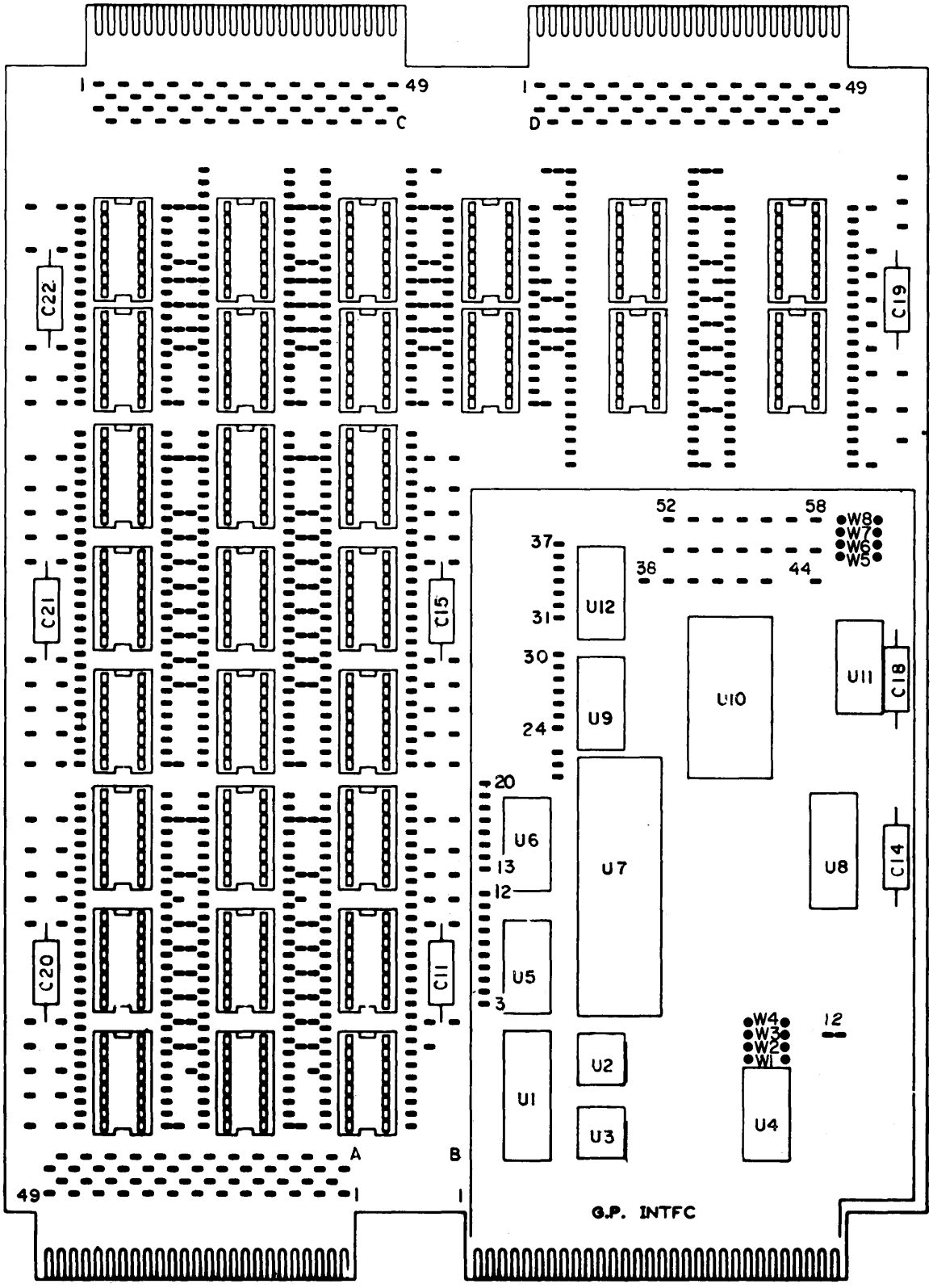
1. □ = WIRE WRAP PINS ON GP INTERFACE WITH PINS AND SOCKETS OR SOLDER POINTS ON GP INTERFACE WITHOUT PINS.
2. INSTALL W4 TO DISABLE INTERNAL ADDRESS COUNTER
3. D<Ø-15>H ARE ACTIVE HIGH, D<Ø-15>L ARE ACTIVE LOW.
4. W5 SETS LOGIC LEVELS OF D<Ø-15>H AND D<Ø-15>L.

DEVICE CODE BITS	0	1	2	3	4	5
JUMPER FOR A1	W3	W2	W1	W6	W7	W8

INSERT W4 TO DISABLE INTERNAL ADDRESS COUNTER
INSERT W5 FOR POSITIVE LOGIC AT 603 PINS

W5	OUT	IN
D<Ø-15>H	H=1 L=0	H=0 L=1
D<Ø-15>L	H=0 L=1	H=1 L=0

GENERAL PURPOSE INTERFACE
PHYSICAL LAYOUT



SUMMARY OF GPIO BUS SIGNALS

The fifty-five signals which comprise the GPIO bus can be divided into five groups:

Data

D(0-15)H Data Out. All data for both data channel and programmed I/O are transferred from the IOC to the device interface via these 16 lines. Each line is buffered to drive 10 standard TTL loads. The contents of the polarity bit (controlled by jumper W5) determines whether a low level should be interpreted as a 0 or a 1.

D(0-15)L Data Input. All data and addresses for both data channel and programmed I/O are transferred from the device interface to the IOC via these 16 input lines. The interrupt disable mask bit is determined by one of these lines when the MSKO signal is asserted. The device code, external register select bit, and the polarity bit are carried on these lines when the signal **IORST** is asserted (see IORST, MSKO, and Jumpers). The device interface should drive these lines with open collector gates. The contents of the polarity bit determines whether a low level should be interpreted as a 0 or a 1.

Programmed I/O

The following control signals are asserted low (1=0V). They can drive up to 10 TTL loads.

DIA Data In A. Asserted by the IOC upon receipt of its DIA instruction.To be used by the device interface to place the contents of its A input buffer on **D(0-15)L**.

DIB Data In B. Asserted by the IOC upon receipt of its DIB instruction.To be used by the device interface to place the contents of its B input buffer on **D(0-15)L** if external registers are enabled (see Jumpers).

DIC Data In C. Equivalent to DATIA, except that it applies to the C input buffer.

DOA Data Out A. Asserted by the IOC upon receipt of its DOA instruction.To be used by the device interface to load the contents of **D(0-15)H** into its A output buffer.

DOB Data Out B. Asserted by the IOC upon receipt of its DOB instruction. To be used by the device interface to load the contents of **D(0-15)H** into its B output buffer

DOC Data Out C. Equivalent to DATOB, except that it applies to the C output buffer.

STRT Start. Asserted by the IOC upon the receipt of any of its non-skip I/O instructions in which bits 8 and 9 = 01 (i.e., instructions in which the S control function is specified). Asserted during DIA, DIB, DIC, DOA, DOB, and DOC instructions after the data transfer has occurred. Usually used to initiate the device interface by setting the Busy flag to 1 and the Done flag to 0.

CLR Clear. Asserted by the IOC upon the receipt of any of its non-skip I/O instructions in which bits 8 and 9 = 10 (i.e., instructions in which the C control function is specified). Asserted during DIA, DIB, DIC, DOA, DOB, and DOC instructions after the data transfer has occurred. Usually used to terminate device operation by setting the Busy and Done flags to 0.

IOPLS I/O Pulse. Asserted by the IOC upon receipt of any of its non-skip I/O instructions in which bits 8 and 9 = 11 (i.e., instructions in which the P control function is specified). Asserted during DIA, DIB, DIC, DOA, DOB, and DOC instructions after the data transfer has occurred. Usually used to initiate special device operations.

SET BUSY Asserted by the interface when it is busy and should not be disturbed by the IOC. Sets the Busy flag in the IOC to 1.

SET DONE Asserted by the interface to notify the IOC that it has completed its operation. In the IOC it sets the Done flag to 1 and the Busy flag to 0.

Program Interrupt

INT SYNC Interrupt Synchronize. Asserted by the interface to notify the program that it has completed its operation. In the IOC it directly initiates a program interrupt request without disturbing either the Busy or Done flags.

MSKO Mask Out. Asserted by the IOC during the execution of a MSKO instruction. Loads the selected Data line into the priority mask bit register.

Data Channel

DCH SYNC Data Channel Synchronize. Asserted by the device interface to request data channel service.

DCHA Data Channel Acknowledge. Asserted by the IOC at the beginning of each data channel cycle in response to a data channel request from the device interface. The interface should respond by placing the memory address on **D(1-15)L** and the direction of transfer on **D(0)L**. A logical 1 on **D(0)L** indicates an input transfer, and a logical 0 indicates an output transfer.

DCHI Data Channel Input. Asserted by the IOC during a data channel input cycle. To be used by the interface to place the contents of its data register onto the **D(0-15)L** lines.

DCHO Data Channel Output. Asserted by the IOC during a data channel output cycle after the IOC has placed the contents of its data register onto the **D(0-15)H** lines. To be used by the interface to load the contents of this bus into its data register.

WCEZ Word Count Equals Zero. Asserted by the IOC during the data channel cycle that overflows the word count register. Can be used to clock the Done flip-flop in the interface.

CLK Clock. Follows **FSTROBE** when the IOC is not performing an operation.

System Control

IORST I/O Reset. Asserted by the IOC during the IORST instruction or when the Hand Held Console reset switch is released (provided the keylock switch on the front panel of the console is in the RUN position). **IORST** is also asserted when the keylock switch is turned from RESET to RUN or to OFF. This signal initializes the IOC by loading the device code, the polarity bit and the external register bit. It should also initialize the interface.

Interface Timing

Timing diagrams for programmed I/O and data channel operations are given below. Each signal, or group of signals, is represented by a horizontal line with a raised section. For control signals the raised section represents the time they are asserted. For data signals the raised section indicates the time during which the information is held on the bus. The level of a line in the diagram has no connection with the voltage level of the signal. All times are in nanoseconds.

Programmed Transfers

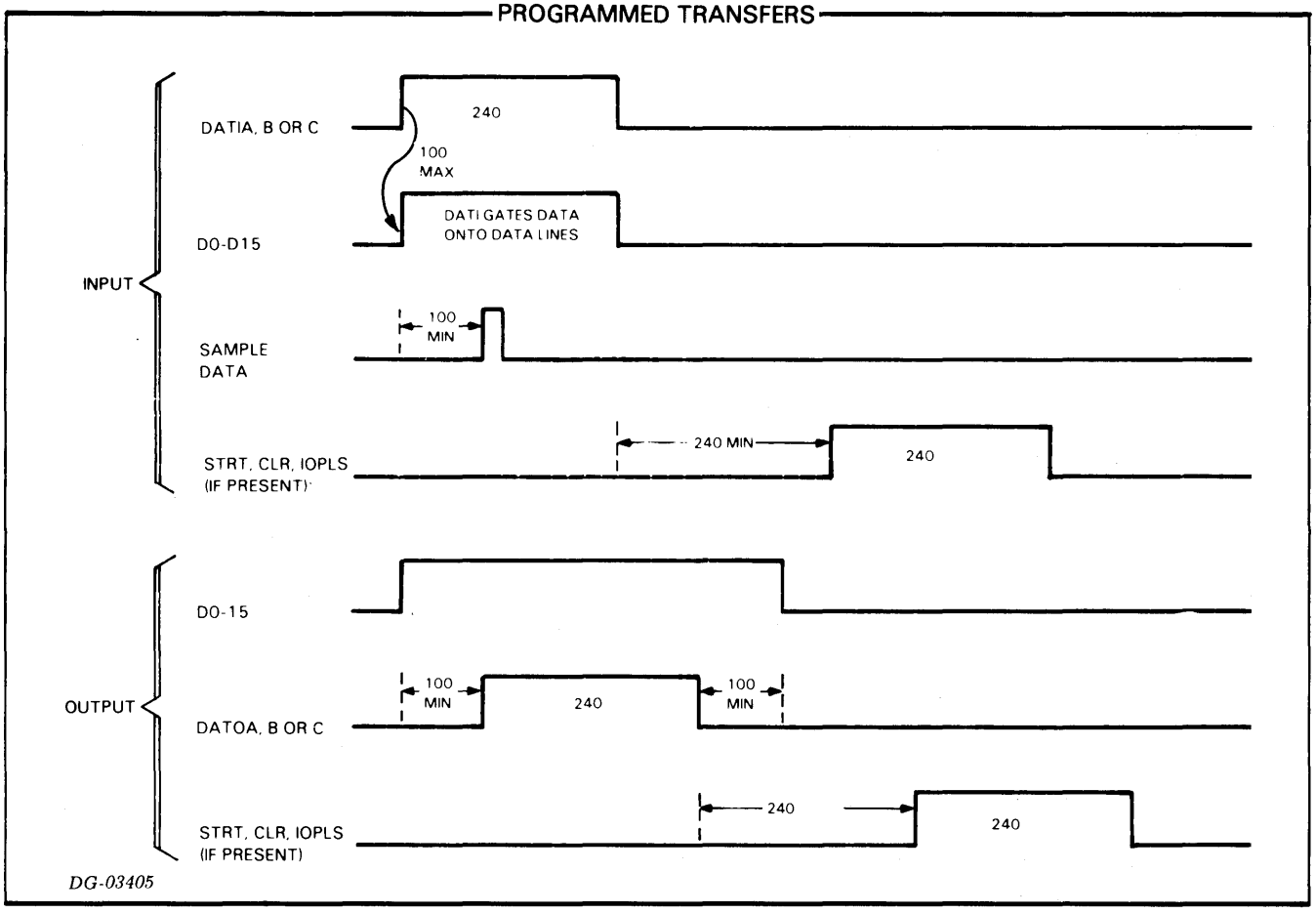
A transfer occurring under direct program control moves a word or part of a word between the IOC and a register in the device interface.

Data In - The I/O controller asserts **DATIA, DATAIB,** or **DATIC.** It also asserts **STRT, CLR** or **IOPLS** if they are specified by the I/O instruction. When the signal **SAMPLE DATA** occurs, any data on the GPIO data bus (D0-D15)L lines will be gated from the interface, through the IOC, and onto the I/O data bus. This signal, however, is internal to the IOC and is shown for reference only.

Data Out - The I/O controller (IOC) places the data received from the CPU onto the GPIO data bus **D(0-15)H** lines and asserts **DATOA, DATOB,** or **DATOC.** It also asserts **STRT, CLR** or **IOPLS** if they are specified by the I/O instruction.

Data Channel Transfers

An information transfer occurring under data channel control moves a block of data, one word at a time, between the IOC and the device.



Data In - The IOC generates **DCHA** and the interface responds by placing the memory address on **D(1-15)L** if the external register is enabled, and by placing the direction of transfer on **D(0)L**. If the word count register in the IOC overflows, **WCEZ** is generated to indicate that this is the last word in the data block. The IOC then generates **DCHI** and the interface responds by placing its data on **D(0-15)L**.

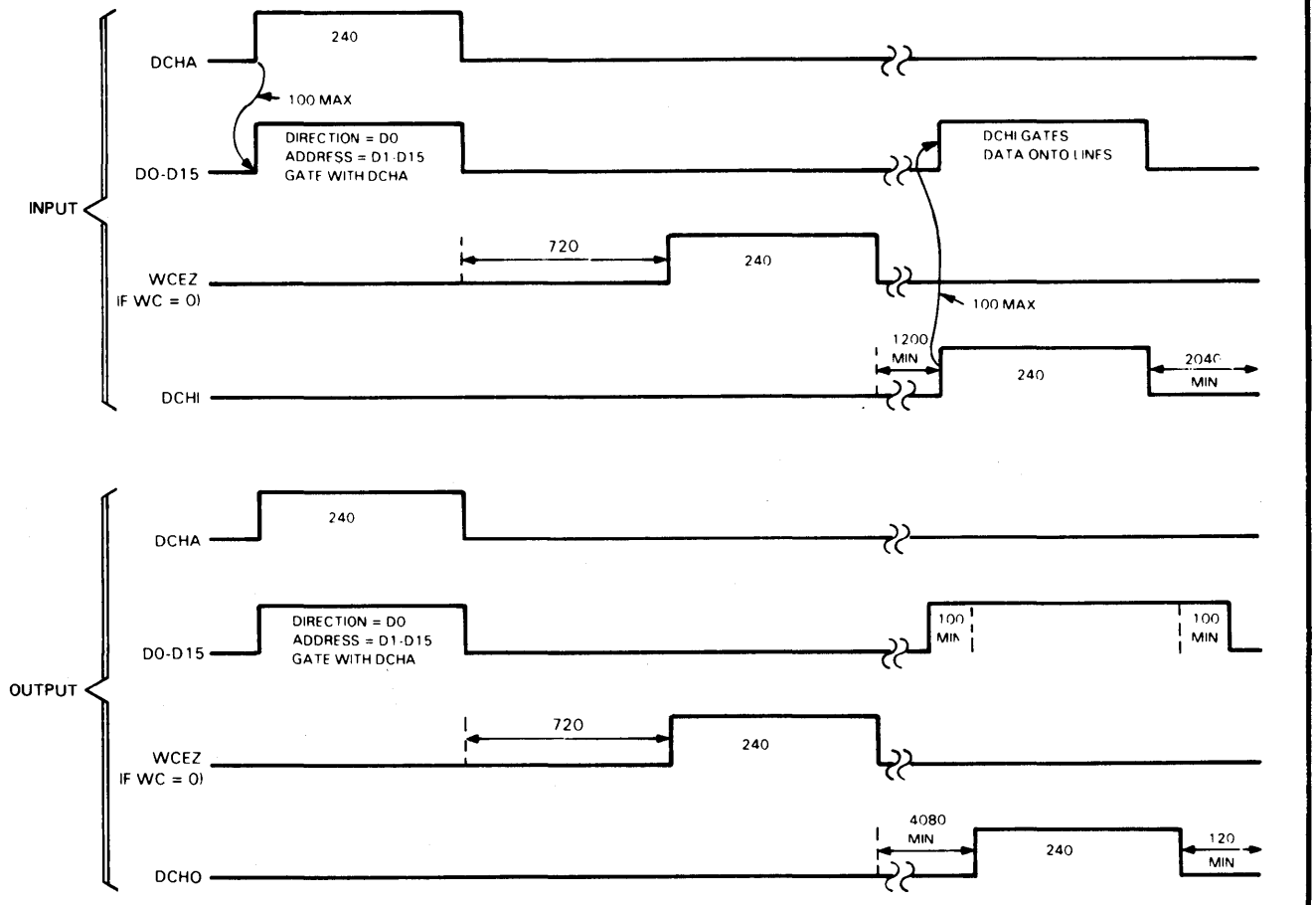
Data Out - The IOC generates **DCHA** and the interface responds by placing the memory address on **D(1-15)L** if the external register is enabled, and by placing the direction of transfer on **D(0)L**. If the word count resister in the IOC overflows, **WCEZ** is generated to indicate that this is the last word in the data block. The IOC then generates **DCHO**, and the interface responds by loading the contents of **D(0-15)H** into its data register.

Jumpers

The device code, polarity bit and external registers select bit are selected by jumpers in the IOC section of the board.

The device code is selected with jumper connections W1-W3 and W6-W8.

DATA CHANNEL TIMING



DG-03406

Jumper W5 selects the polarity bit. The polarity bit is a 1-bit register that determines the sense of the data bits transmitted and received via the IOC. If W5 is in, the polarity bit is set to a 1 and a low level (0V) on the data pins of the IOC is interpreted as a 0. A high level (5V) is interpreted as a 1. If W5 is out, the polarity bit is set to a zero and a low level on the data pins of the IOC is interpreted as a 1. The high level is interpreted as a 0. Note that the buffered outputs, **D(0-15)H**, are inverted.

W5	DATA POLARITY (GPIO BUS)	
IN OUT	D(0-15)H	ZERO=+5V, ONE= 0V ZERO= 0V, ONE=+5V
IN OUT	D(0-15)L	ZERO= 0V, ONE=+5V ZERO=+5V, ONE= 0V

Jumper W4 controls the selection of the external (device interface) or internal (IOC) memory address and word count registers.

W4	Location of Registers
IN	DEVICE INTERFACE
OUT	IOC

The interrupt priority mask bit is selected by jumpering the mask signal (**MSKO**, pin 44) to one of the **D(0-15)L** lines.

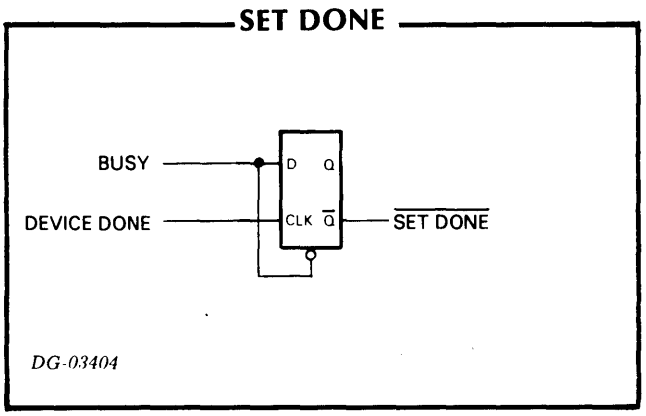
Data Lines and Drive Capability

The outputs of the I/O controller (IOC) chip are capable of driving only 1 TTL load. Therefore, all the data out lines, **D(0-15)H** have been TTL buffered, and are capable of sinking 16mA. The outputs of the 4 to 16 decoder are also capable of sinking 16mA. The data input lines, **D(0-15)L**, should be driven with open collector drivers. Each control signal to the IOC (**INTSYNC**, pin 23; **DCHSYNC**, pin 22; **SET BUSY**, pin1; and **SET DONE**, pin 2) constitute 1 TTL input load.

The supply voltages required (+5Vdc, pin 58; +15Vdc, pin 57; and -5Vdc, pin 51) must be supplied to the board by the system into which it is installed. See the section on Power Supply Assemblies for proper supply voltage sequencing. The maximum current drain on the +5Vdc should be 1 ampere.

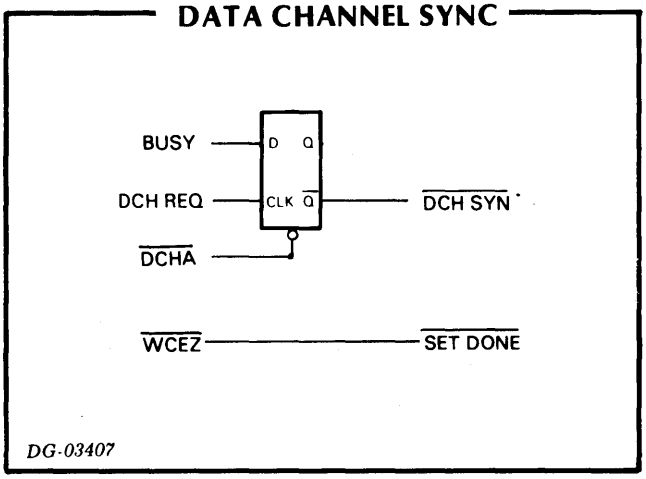
Busy/Done

A suggested circuit for generating the **SET BUSY** and **SET DONE** signals is given below.



Interface Wire Wrap Pins

Wire wrap pins are provided in the IOC section of the model 4211 board to facilitate the connection of the GPIO bus to the custom device controller. Below are listed the wire wrap pins associated with each bus signal. The location of the pins may be found by referring to the physical layout of the board. The model 4210 GPIO board does not include wire wrap pins, but features etched circuit holes in the same locations.



WIRE WRAP PINS (GPIO BOARD)

PIN	SIGNAL	PIN	SIGNAL
1	$\overline{\text{SET BUSY}}$	30	$\overline{\text{DONE}}$
2	$\overline{\text{SET DONE}}$	31	D9H
3	$\overline{\text{MASTER CLOCK}}$	32	D9L
4	D12H	33	D13H
5	D12L	34	D13L
6	D11H	35	D14H
7	D11L	36	D14L
8	D10H	37	D15L
9	D10L	38	D15H
10	D4H	39	$\overline{\text{DIB}}$
11	D4L	40	$\overline{\text{DOA}}$
12	D3H	41	$\overline{\text{CLR}}$
13	D7H	42	$\overline{\text{DCHA}}$
14	D7L	43	$\overline{\text{CLK}}$
15	D6H	44	$\overline{\text{MSKO}}$
16	D6L	45	$\overline{\text{DOC}}$
17	D5H	46	$\overline{\text{DIA}}$
18	D5L	47	$\overline{\text{STRT}}$
19	D3L	48	$\overline{\text{IORST}}$
20	D1H	49	$\overline{\text{WCEZ}}$
21	D1L	50	$\overline{\text{DCHO}}$
22	$\overline{\text{DCH SYN}}$	51	-5V
23	$\overline{\text{INT SYN}}$	52	$\overline{\text{BUSY}}$
24	D0H	53	$\overline{\text{DOB}}$
25	D0L	54	$\overline{\text{DIC}}$
26	D2H	55	$\overline{\text{OPLS}}$
27	D2L	56	$\overline{\text{DCHI}}$
28	D8H	57	+15V
29	D8L	58	+5V

DG-03408

SECTION X
MISCELLANEOUS PRINTED CIRCUIT BOARDS

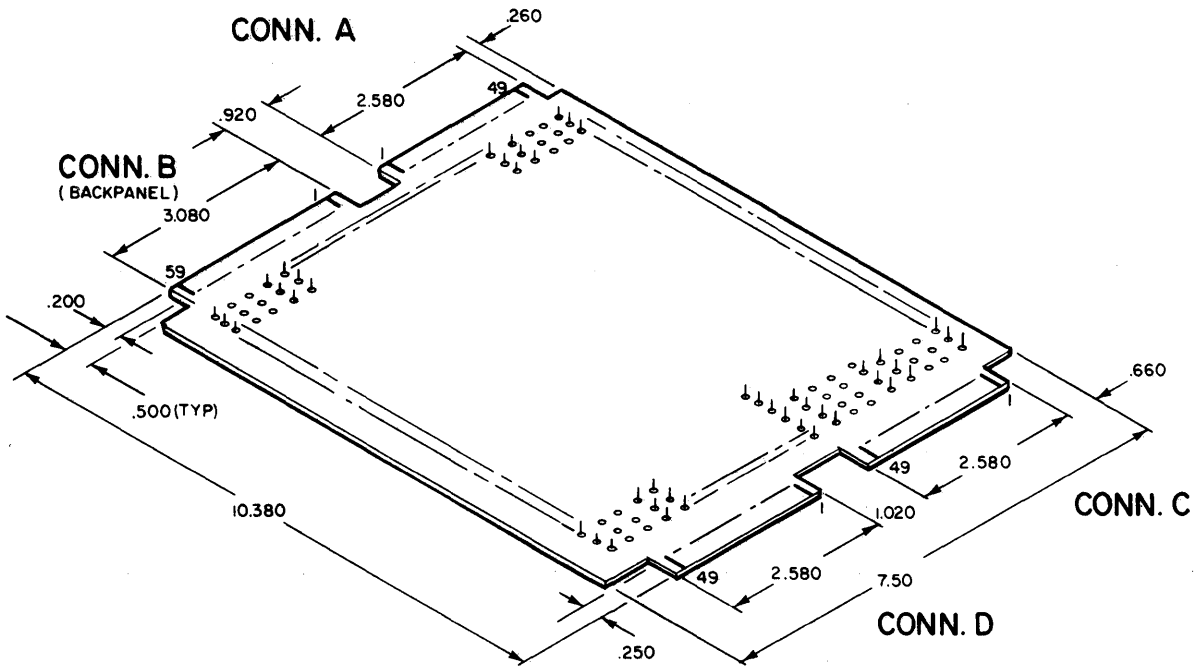
Models 1114 and 2301 boards available for custom applications and maintenance with the microNOVA computer. the model 1114 general purpose wiring board is drilled and etched, with wire wrap pins, to accept up to fifty 14-pin integrated circuits. The model 2301 extender board carries the backpanel signals to female edge connectors on the front of the board.

OVERVIEW AND INTRODUCTION

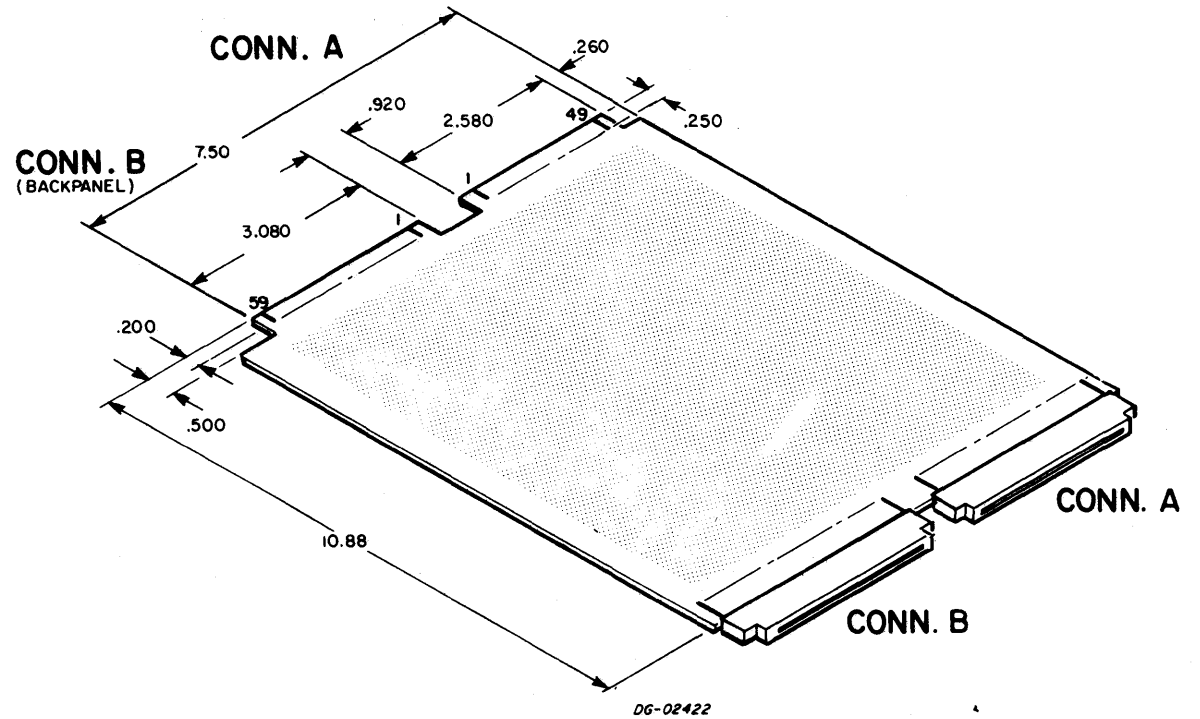
The Model 1114 Pre-Drilled Circuit Board is a general purpose wiring board suitable for the user who wishes to build a special purpose, custom interface to a microNOVA computer. The board has two male edge connectors on the backpanel edge, one of which plugs into the printed circuit backpanel socket; the other plugs into a device cable connector, in the event the board is used as an I/O device controller. The backpanel connector has 60 contacts, and the device connector has 50 contacts. Two additional 50 contact edge connectors are included on the forward edge of the board to provide flexibility to the interface designer. A series of holes in the card is designed to accept standard integrated circuit packages. Up to fifty 14-pin packages may be installed on one board, and an even larger number of smaller packages may be used. Each hole and each connector finger is connected by etch to an adjacent wire wrap pin.

The Model 2301 Extender Board is used for maintenance of any board in the microNOVA product line. The board has edge connectors which plug into the backpanel and a device cable connector. Etched conductors on the board carry backpanel and I/O signals to two female edge connectors on the forward edge of the board. Any microNOVA board may plug into the extender board and operate in this position, giving access for testing and maintenance.

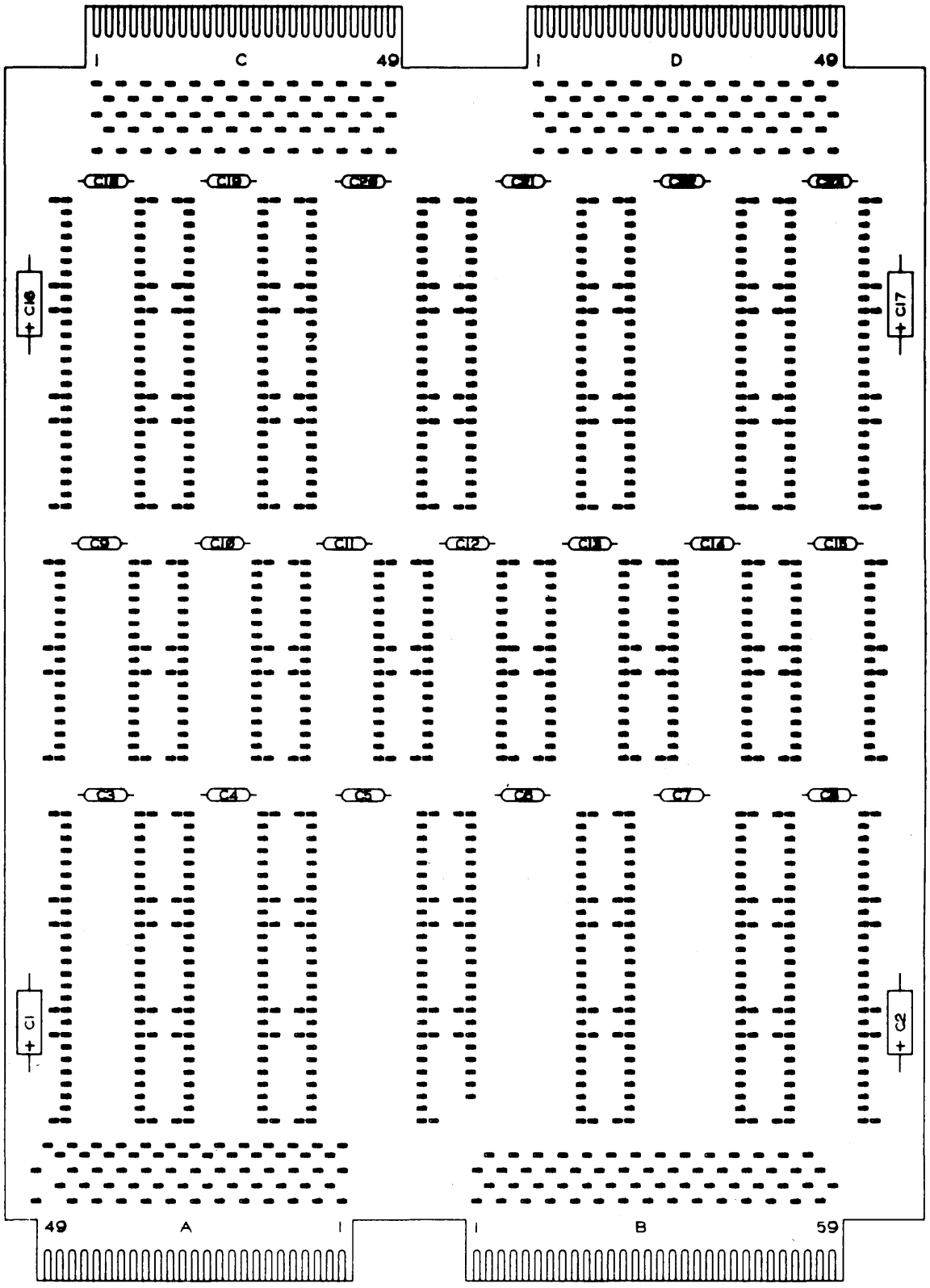
PREDRILLED CIRCUIT BOARD



EXTENDER BOARD

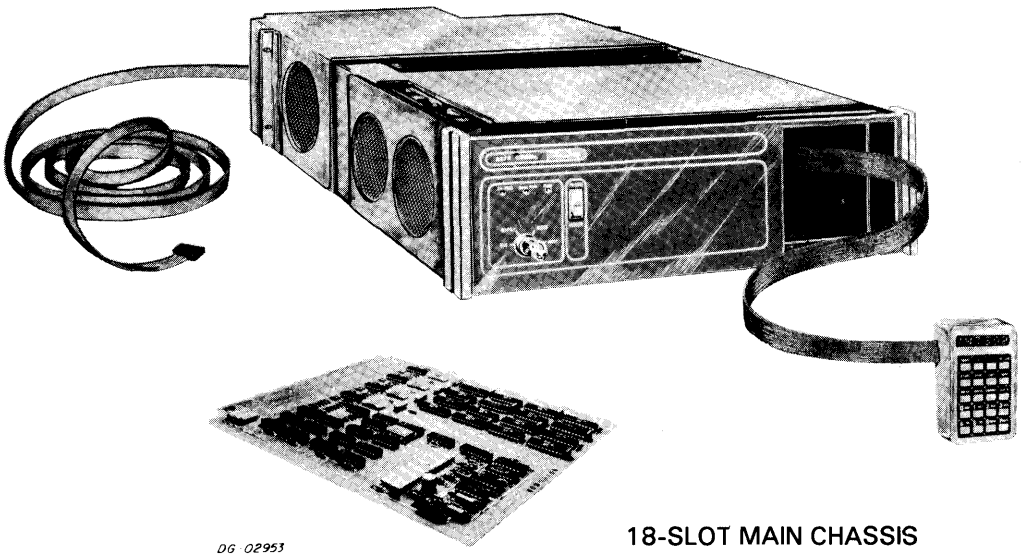


MODEL 1114 BOARD PHYSICAL LAYOUT



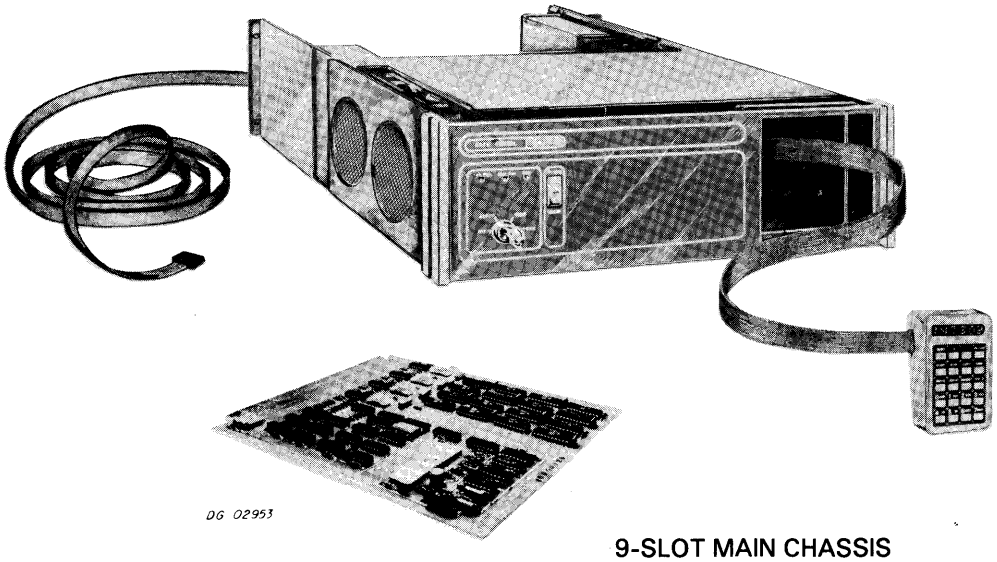
This page intentionally left blank.

microNOVA CHASSIS



DG 02953

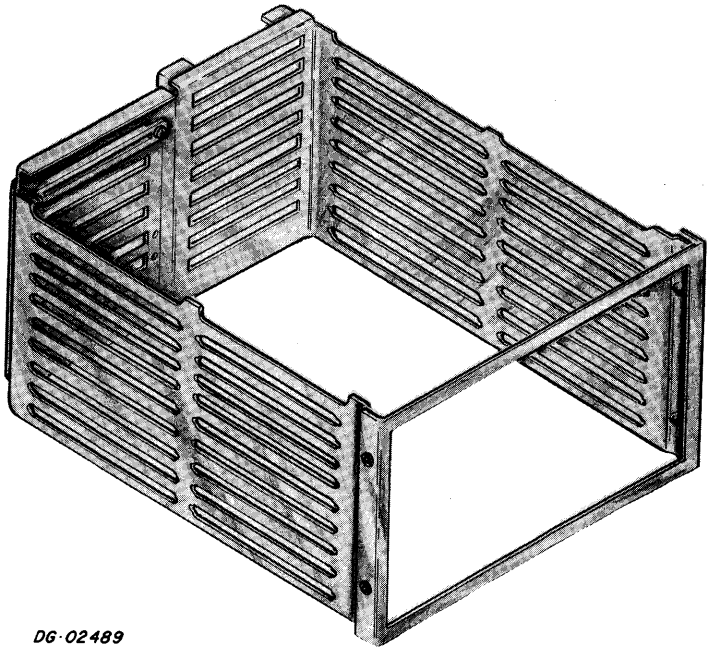
18-SLOT MAIN CHASSIS



DG 02953

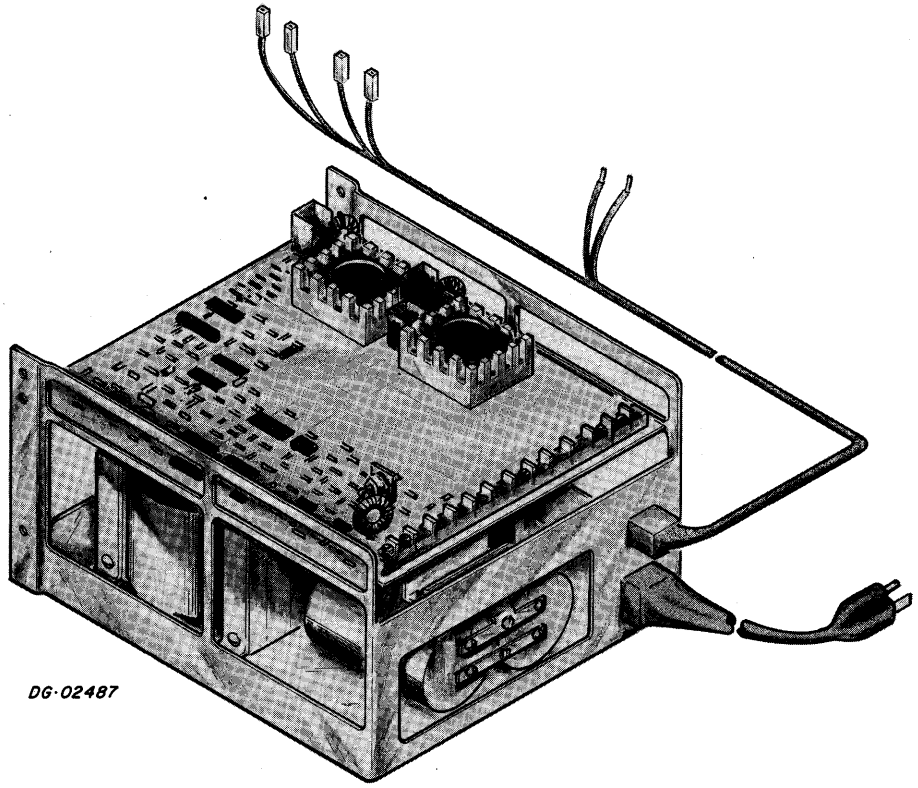
9-SLOT MAIN CHASSIS

microNOVA CARDFRAME



DG·02489

microNOVA POWER SUPPLY



DG·02487

SECTION XI
microNOVA CHASSIS

Data General uses two distinct chassis to package microNOVA printed circuit boards. The chassis are rack-mountable, stamped metal enclosures which interconnect and supply power to the microNOVA line of printed circuit boards. Each includes cardframes, a power supply with optional battery backup capability, cooling fans, and a front panel. A printed circuit backpanel carries the memory bus, I/O bus, and necessary power from board to board. Major chassis components may be ordered separately for special applications, excluding front panel and metal chassis.

OVERVIEW AND INTRODUCTION

Data General offers two microNOVA computers as standard products for use with the microNOVA line of printed circuit boards. Two distinct chassis, especially designed to accept microNOVA computer boards, are used for these two models. The chassis differ in board capacity, overall size, and application. Models 8560 and 8561 are complete microNOVA computers with open slots available for additional memory and/or I/O control boards from the microNOVA product line. Each of these models contains a CPU board (microNOVA CPU and 4K words of MOS memory) installed in the chassis, and therefore comprises a basic microNOVA computer. The chassis used with these models contain one or two 9-slot cardframes for additional boards, a front panel with operator controls, a power supply and internal cabling, and are available with optional battery backup units. These chassis are denoted as 9- or 18-slot main chassis. All models mount in a standard 19" equipment cabinet.

The cardframe and power supply may be ordered separately, for those customers who desire to install a microNOVA computer in custom equipment. The Model 4212 Cardframe is designed to accept and interconnect up to 9 printed circuit boards from the microNOVA product line. The Models 4213 and 4214 Power Supplies are designed to provide all necessary power for up to two fully occupied cardframes. Overall dimensions and electrical specifications are identical for both power supplies. Either supply may be installed in a 9- or 18-slot microNOVA chassis. The Model 4214 includes a battery unit, which supplies operating power during loss of primary line voltage, and charging and switchover circuits to support the battery service.

The remainder of this section describes, in more detail, the characteristics of each model.

CHASSIS & CHARACTERISTICS

8560 Chassis	17 available slots (8 memory and/or I/O and 9 I/O), includes CPU and 4K MOS; battery optional
8561 Chassis	8 available slots (memory and/or I/O), includes CPU and 4K MOS; battery optional
4212 Cardframe	Accepts up to nine microNOVA printed circuit boards. Includes backpanel and stiffening frame.
4213, 4214 Power Supply	Provides power for up to 18 microNOVA printed circuit boards. Includes power cord, fuses, and internal cable for dc power distribution.
	The Model 8560 main chassis contains two Model 4212 cardframes with a Model 8563 CPU board,a power supply, and a front panel with power/reset/run switch, indicator lights, and program load/continue switch.
	The Model 8561 main chassis is identical to the Model 8560, but contains only one Model 4212 cardframe.

CHASSIS

The microNOVA chassis are sheet metal enclosures containing a power supply, one or two 9-slot cardframes, an optional battery backup unit, a front panel, and interconnecting cables. Ventilation fans on the left side of the enclosure provide air flow to cool the printed circuit boards and power supply. The fans draw air in from the left, across the power components, and out the right side of the enclosure. Incoming air is roughly filtered by wire mesh grids.

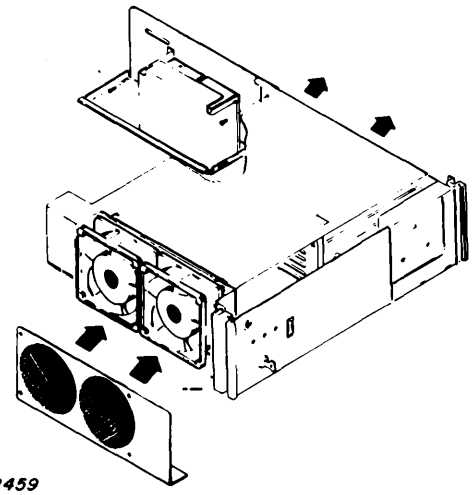
The chassis mounts in a standard 19in. equipment cabinet, and occupies 5 1/4in. of rack space. The front face of the sheet metal enclosure is open, and is covered by a hinged front panel as described below. Cardframes mount inside the enclosure, facing forward to give front access to remove and install microNOVA boards. The 18-slot main chassis contains two cardframes, mounted side-by-side inside the enclosure. The 9-slot chassis contain one cardframe, mounted on the left side of the enclosure. The physical size of the enclosure varies, depending on the number of cardframes in the chassis. The main component is an open frame which bolts to the forward cabinet rails, and is the same unit for one or two cardframe applications. The enclosure's rear support member varies according to the number of cardframes installed. The single cardframe in the 9-slot chassis is located on the left next to the cooling fans, and the power supply is located in the space on the right side. In this case, sheet metal extensions are attached to the rear of the enclosure and bolt to the rear cabinet rails. The 18-slot chassis has an additional sheet metal housing, fastened to the rear of the main enclosure frame, which houses the power supply in addition to providing rear support to the enclosure. An additional ventilation fan is included on the left side of this housing to cool the power supply. This housing increases the overall depth of the enclosure; overall dimensions are shown in figures at the end of this section.

A strain relief bracket fastened to the rear of the enclosure behind the cardframes provides mechanical support for I/O and device cables and prevents inadvertent damage to the printed circuit boards. The bracket accepts from 1 to 20 ribbon cables.

A hinged front panel, containing certain operator controls, covers the front of the enclosure and provides convenient access to the microNOVA printed circuit boards. The panel hinge is on the left side, and allows the panel to swing open fully. A latch on the right side holds the panel securely closed. The front panel used with the 9- or 18- slot main chassis contains a four-position power/run/reset keylock switch, three lights to indicate ac power, dc backup power, and CPU run condition, and a two-position program load/continue rocker switch. Additionally, a recess designed to store the handheld console is molded into this panel.

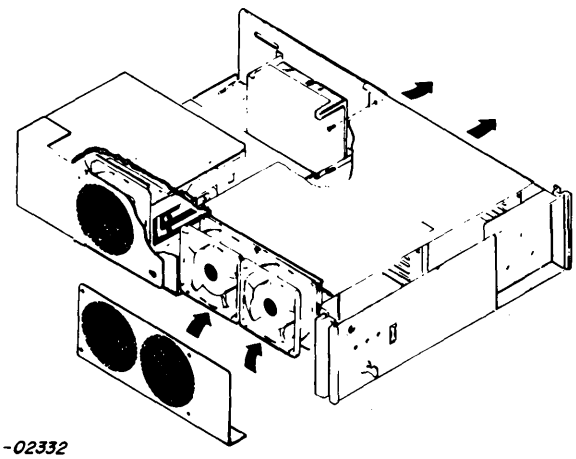
Electrical signals from the front panel controls are carried to their destinations within the enclosure by cables. All connections to the power supply and microNOVA boards are made with plugs, and the cables are tied to the rear of the front panel to provide unobstructed access when the panel is open.

9-SLOT CHASSIS



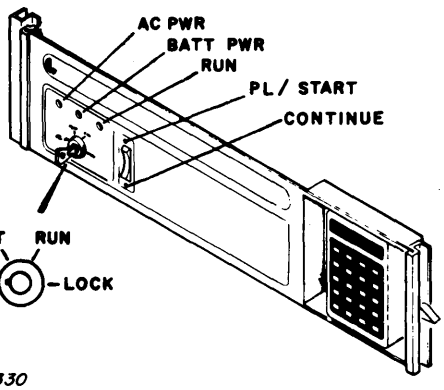
DG-02459

18-SLOT CHASSIS



DG-02332

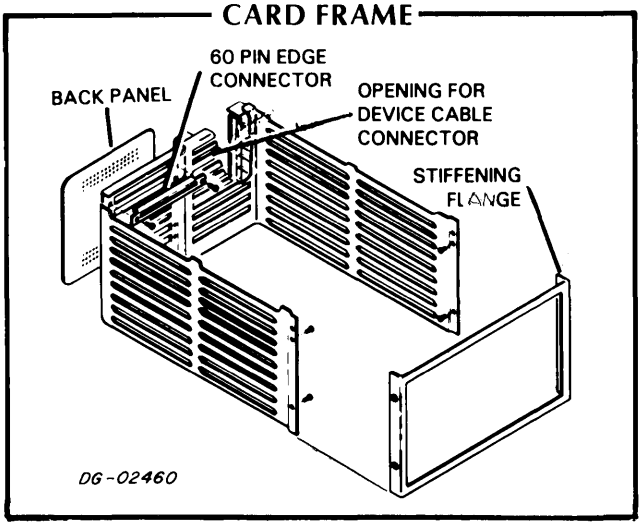
CHASSIS FRONT PANEL



DG-02330

CARDFRAME

The Model 4212 cardframe is the basic building block of the microNOVA chassis; it is a stamped steel enclosure with slots for nine microNOVA printed circuit boards, a printed circuit backpanel with nine 60 pin female edge connectors, nine adjacent openings in the rear for device cable connectors, and openings for cooling air flow. A printed circuit board is installed in one of the slots from the front of the cardframe. The 60 pin edge connector on the board mates with and plugs into the 60 pin socket on the backpanel; the 50 pin edge connector on the board is then aligned with the corresponding opening for a device cable connector. Spring clips on the rear of the cardframe hold the cable connectors securely in position.



The printed circuit backpanel carries memory and I/O signals among the boards installed in the cardframe, and delivers the necessary power to the boards from the power supply. Etch on the backpanel passes all signals in parallel from one 60 pin edge connector to the next. Every contact on all edge connectors corresponds to a wire-wrap pin protruding from the rear of the backpanel; the pins are spaced so that standard plug connectors may be used to extend the I/O bus, communicate with certain peripherals, etc. The pins are numbered as shown in the diagram -- viewed from the pin side, the pins are numbered from left to right, 1 to 60, with odd numbers on the top and even numbers on the bottom. The slots are numbered 1 through 9, from bottom to top.

Two jumper wires may need to be installed on the backpanel. They are used to continue the data channel and program interrupt priority chains, which pass from slot to slot via etch on the backpanel. These priority chains are terminated at a slot when that slot does not contain a board. In order to include any boards above the empty slot in either of these priority networks, they must be continued across the empty slot with jumper wires wrapped to the appropriate backpanel pins. These pins are denoted as INTP and DCHP, and may be located by reference to the DGC engineering drawing of the backpanel layout.

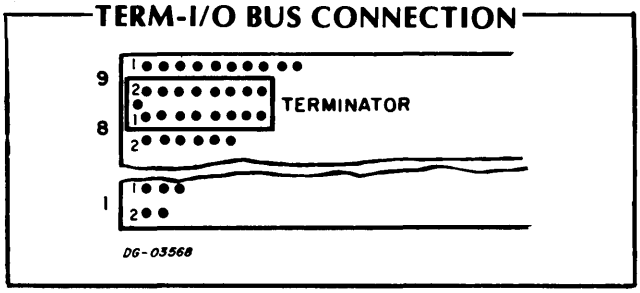
Any board in the microNOVA series will fit in any slot; however, when a cardframe contains the microNOVA CPU board, that board must be installed in slot number 1. The remaining slots may contain a mixture of memory and I/O boards, but all memory boards in the entire system must be installed in this cardframe, which is called the primary cardframe. The slot assignment diagram shows this arrangement. A microNOVA computer system may contain up to 3 fully occupied card frames. The expansion cardframes may contain any I/O board, but may not contain memory boards or an additional CPU board. When a system is expanded to include more than one cardframe, the I/O bus is extended from one frame to the next with a daisy-chain I/O cable, whether the cardframes cages are in the same enclosure or if they are in separate enclosures. An I/O terminator must be installed on the last cardframe in the system.

When a cardframe is ordered separately from the assembled chassis, a stiffening flange is included to reinforce the front. This flange does not interfere with installation or removal of the microNOVA printed circuit boards. The stiffening flange is not necessary when the cardframe is installed in a microNOVA chassis enclosure.

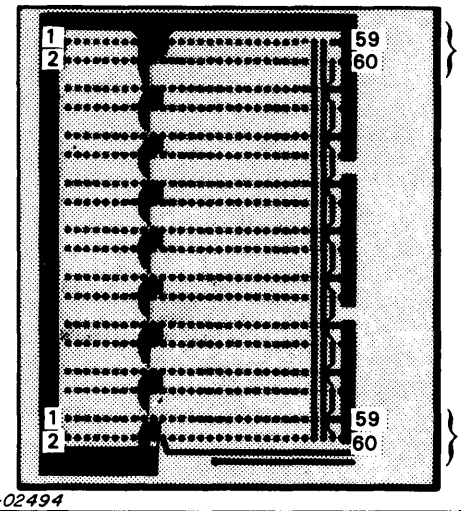
CARDFRAME PIN ASSIGNMENTS

	1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37	39	41	43	45	47	49	51	53	55	57	59
DDO	MCLOCK	GND	BIO1	INTP OUT	BDCHRT	GND	BIO2	BOCLOCK		INTPOUT	DCHPOUT	BDATA15	BSP1	BDATA14	DATAXFER	BDATA13	BDATA12		BDATA11		BDATA10		BDATA9	BSAE	BDATA8	BOTEN	GND	+15V	VCC	VCC
MDI	MCLOCK	BIO1	CLEAR	BEXTINT	LOCK	BIO2	GND	BOCLOCK	VCC	INTPIN	DCHPIN	BDATA7	BP	BDATA6	FDCHR	BDATA5	BDATA4	GND	BDATA3		BDATA2	BSP2	BDATA1	BWE	BDATA0	PHL	GND	+15V	5V	VCC
	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48	50	52	54	56	58	60

The differential I/O lines must be terminated at the end of the I/O bus. The I/O connector, or terminator, must be inserted between slots 8 and 9 in the 9 slot CPU. To extend the I/O bus to 18 slots, an internal I/O cable must be inserted between slots 8 and 9 of the 9 slot cardframe to slots 1 and 2 of the added 9 slot cardframe. If the I/O bus ends within the added cardframe, a terminator must be inserted between slots 8 and 9. To extend the I/O bus beyond the CPU chassis, an extended I/O bus cable must be inserted in place of the terminator located between slots 8 and 9, whether 9 or 18 slot CPU, and the terminator moved to the end of the bus. The connector, or terminator, must be aligned across pin 1 (slot 8 or 1) and pin 2 (slot 9 or 2) as shown in the drawing.

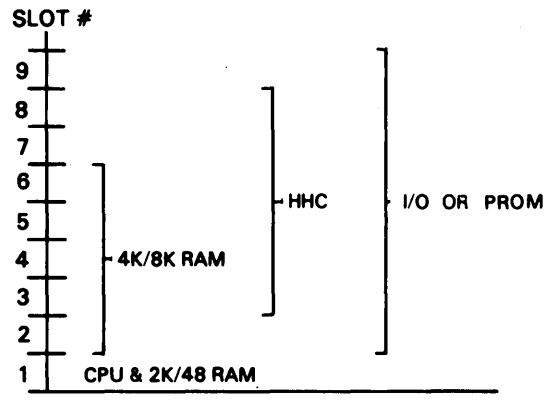


BACKPANEL PIN NUMBERING



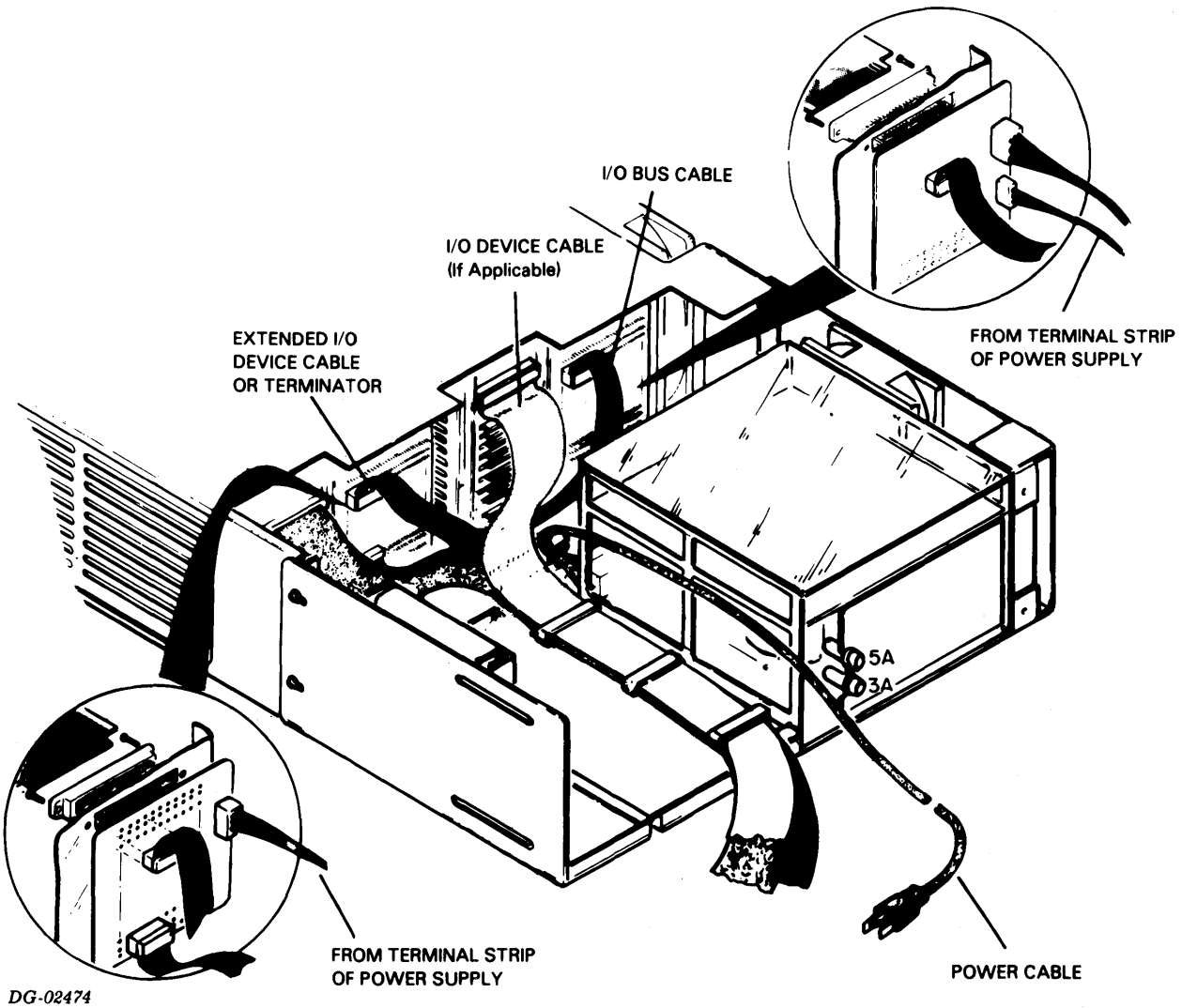
DG-02494

SLOT ASSIGNMENT



DG-03407

EXTERNAL CABLE CONNECTIONS



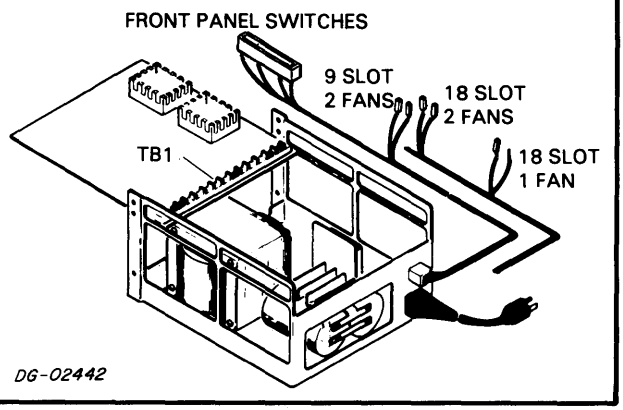
DG-02474

POWER SUPPLY

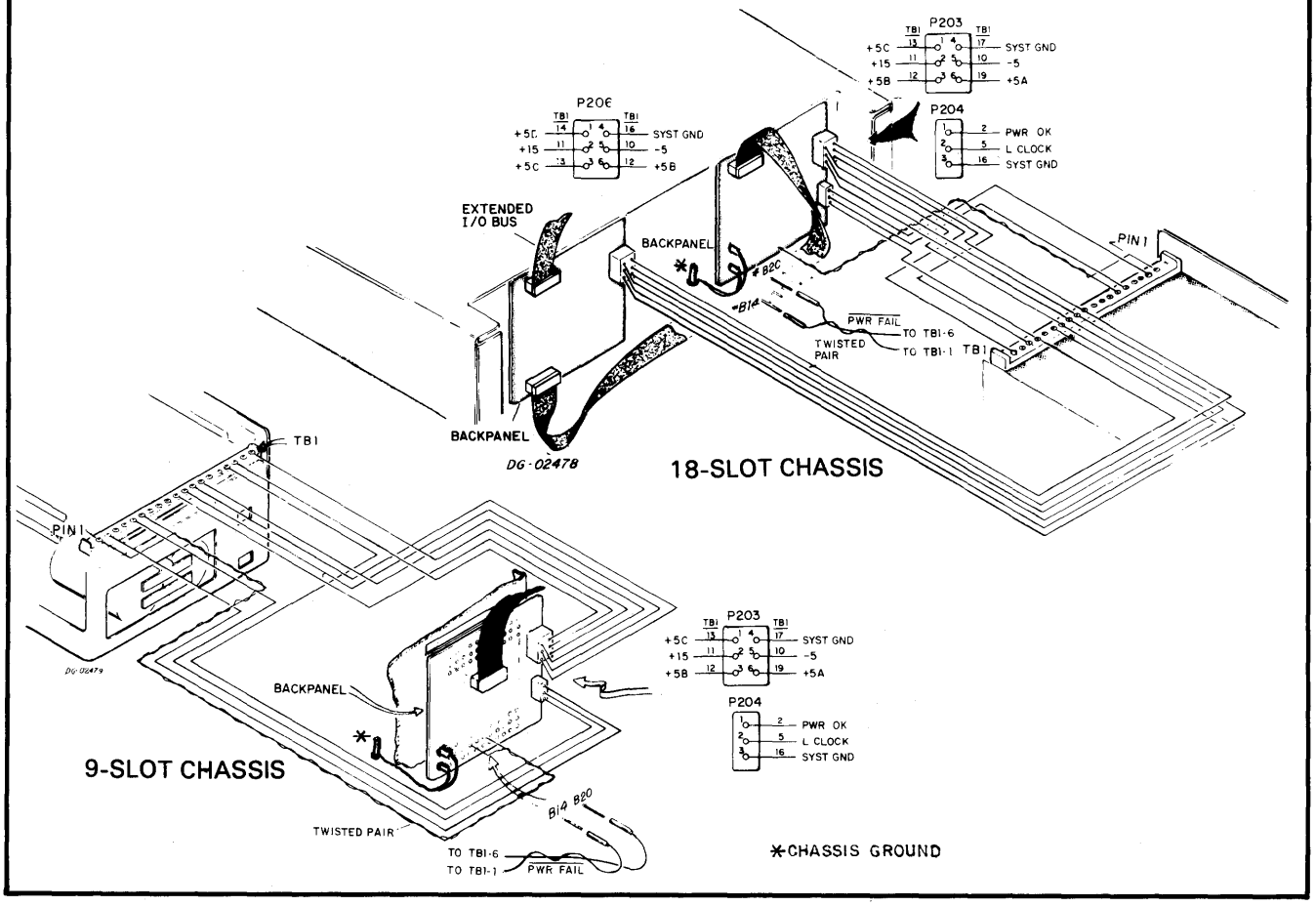
A Model 4213 or 4214 power supply provides the necessary power for one or two fully occupied cardframes, and is contained within a sheet metal box which has the same outside dimensions as a cardframe. The power supply box contains large circuit components, and a printed circuit board is fastened to the top of the box. A 19-lug terminal strip is fastened across one end of the board, and supplies power to the cardframe backpanel(s) through cables. These cables fasten to the terminal strip with standard lugs, and plug into the backpanel(s) with molded connectors. A three-conductor cable, and one or two six-conductor cables, distribute power from the supply to the cardframes. The three-conductor cable and one six-conductor cable are used to provide power to one cardframe; when two cardframes are employed, each receives power from a six-conductor cable, and the three-conductor cable is plugged into the first (CPU + memory) cardframe. The cabling diagram shows these connections for both cases.

The power supply is placed in two different locations within the chassis, depending on the number of cardframes in the chassis. The power supply is on the right-hand side of the 9-slot chassis, next to the single cardframe. The terminal strip is toward the rear and the fuses are accessible from the front. On the 18-slot chassis, which contains two cardframes, the power supply is fastened inside an additional housing, behind the left cardframe. In this case, the terminal strip is toward the front and the fuses are accessible from the rear. In both cases, the line supply power cord is routed out the rear of the enclosure.

POWER SUPPLY



TERMINAL STRIP CONNECTIONS

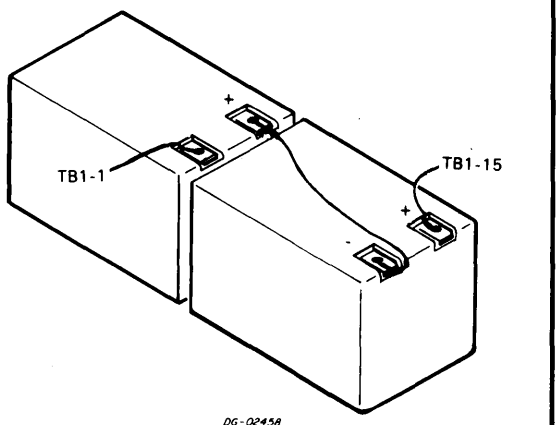


Power Supply with Battery Backup

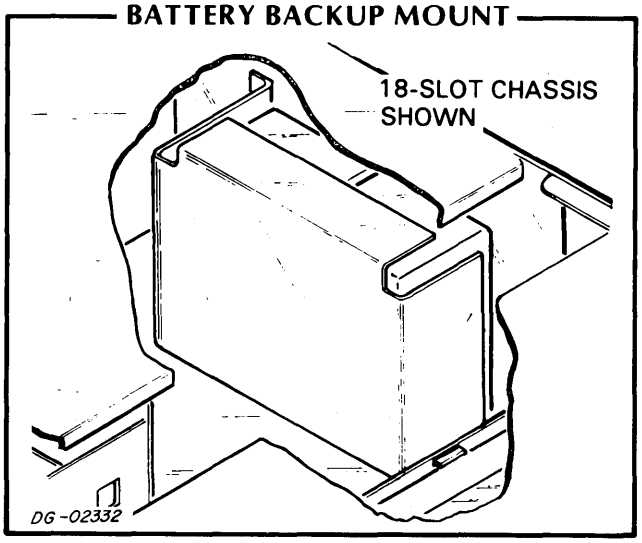
Both power supply models have identical dimensions and power ratings. Model 4214 has the required circuitry to support a battery backup unit, and this battery unit is included with the power supply. The battery backup unit consists of two sealed plastic cases containing lead-acid cells with sufficient charge to power the microNOVA CPU and MOS Memory in the event of primary power failure. A shelf, stamped into the right rear support member of the enclosure, supports the two cases and a sheet metal plate clamps them securely in place. The cases are wired in series, and two wires carry battery power to the power supply terminal strip, using faston connectors on the battery end and terminal strip lugs on the power supply end.

Battery power is used only to maintain data in the microNOVA memory when primary power is lost. The customer may choose battery backup service for up to 6 slots of the primary cardframe, or any of three subsets of slots of that cardframe. Slots 1-3 always receive battery backup service; slots 4-6 or 7-9 may be selected for battery backup service by altering the terminal strip connections on the power supply. The illustration shows the slots which are protected for various terminal strip connections. It is good practice to protect only those slots containing MOS memory boards and the microNOVA CPU, to minimize current drain and overheating (the cooling fans do not operate during battery operation).

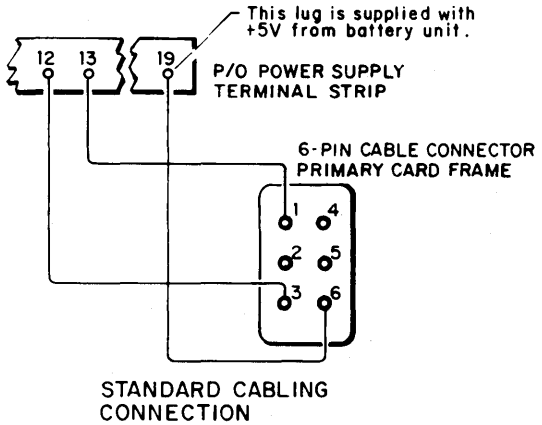
BATTERY BACKUP UNIT

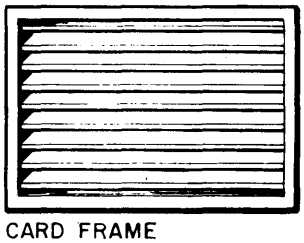


BATTERY BACKUP MOUNT



CABLE CONNECTIONS FOR ADDITIONAL BATTERY PROTECTION





SLOTS 7-9;
PIN 1 MOVED FROM
LUG 13 TO LUG 19

SLOTS 4-6;
PIN 3 MOVED FROM
LUG 12 TO LUG 19

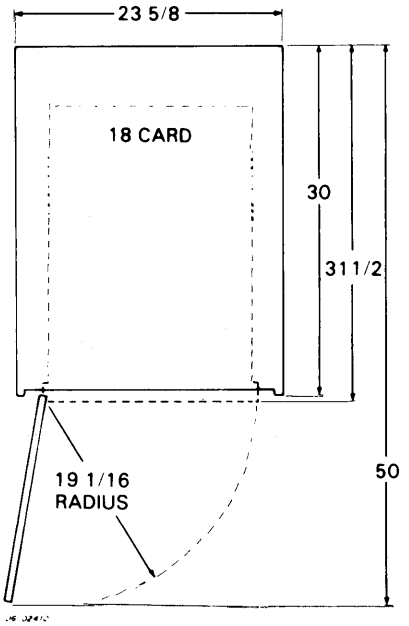
SLOTS 1-3; PIN 6
ALWAYS CONNECTED
TO LUG 19

CARD FRAME
SLOTS BACKED-UP

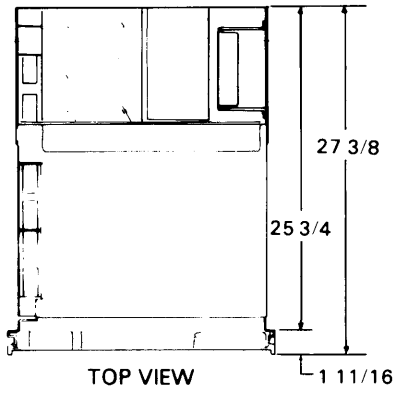
NOTE: MAXIMUM CURRENT
AT TB1-PIN 19 IS 6A.

CHASSIS DIMENSIONS

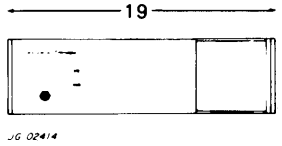
18-SLOT CHASSIS



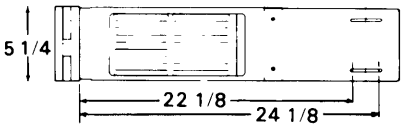
MINIMUM SERVICE CLEARANCE DIMENSIONS



TOP VIEW

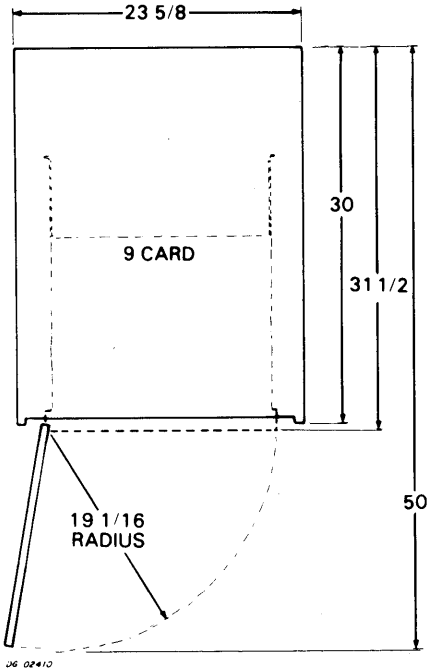


FRONT VIEW

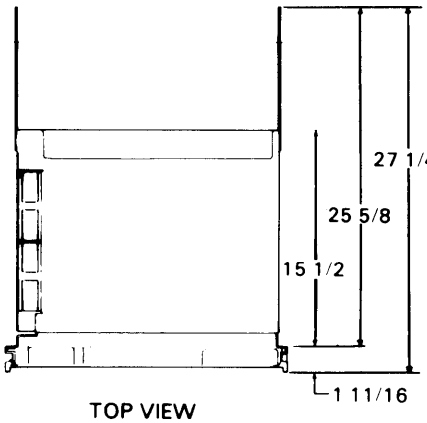


SIDE VIEW

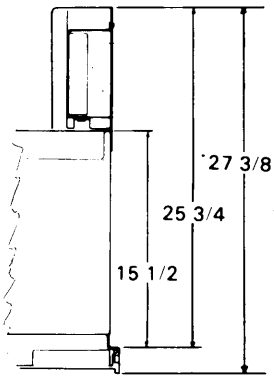
9-SLOT CHASSIS



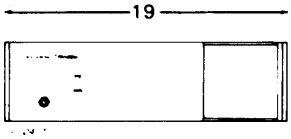
MINIMUM SERVICE CLEARANCE DIMENSIONS



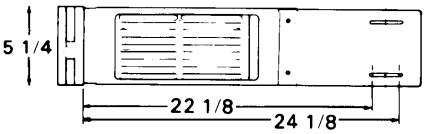
TOP VIEW



TOP VIEW
(WITH BATTERY
BACK-UP OPTION)

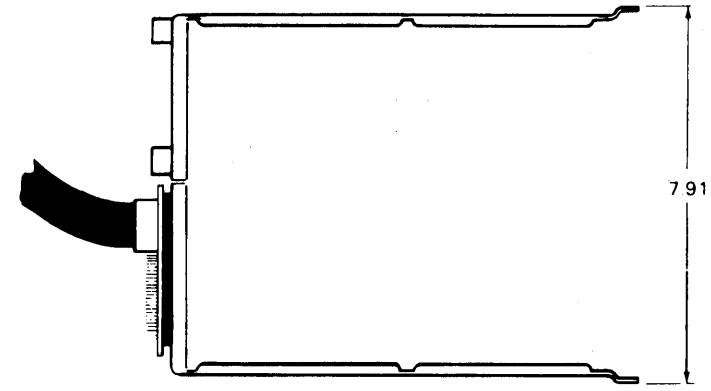


FRONT VIEW

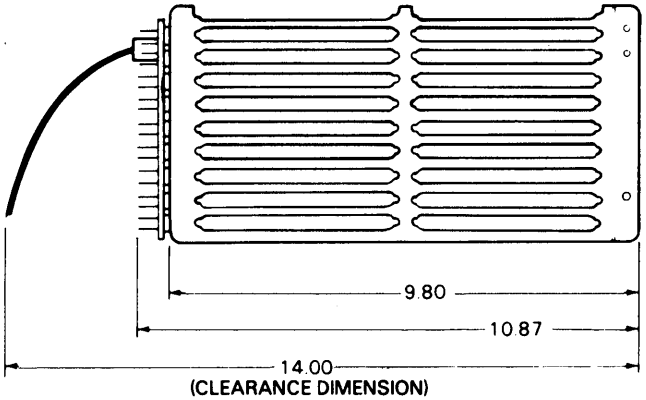


SIDE VIEW

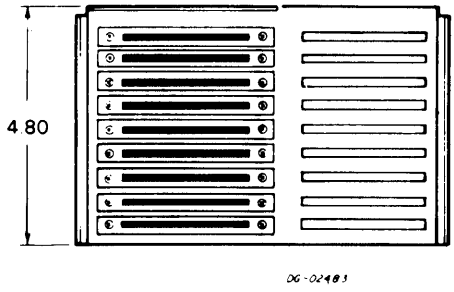
CARDFRAME



TOP VIEW

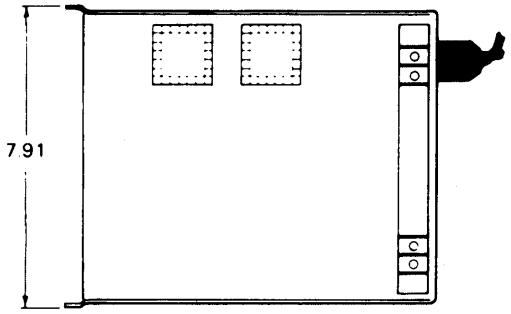


SIDE VIEW

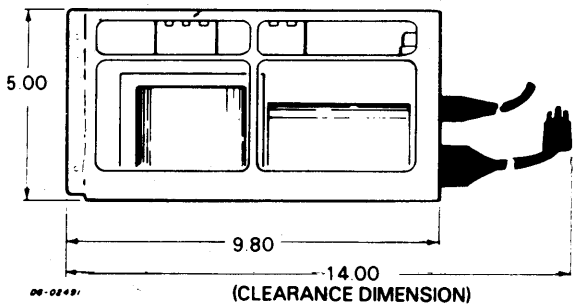


FRONT VIEW

POWER SUPPLY

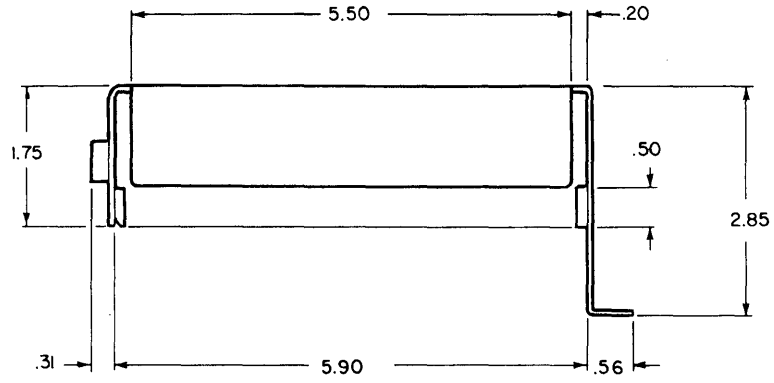


TOP VIEW

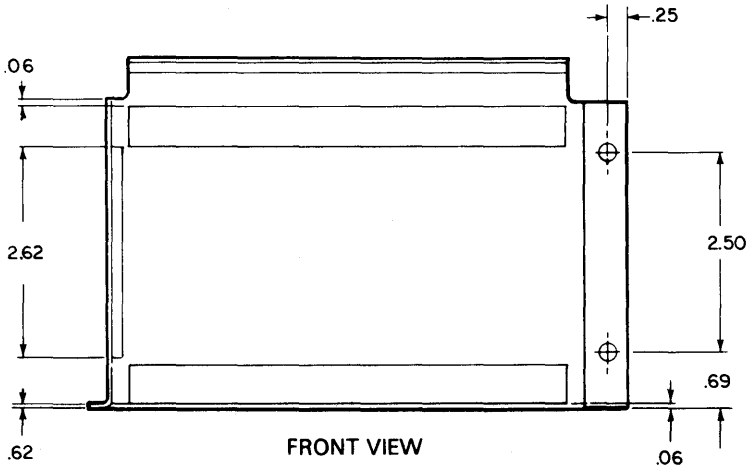


FRONT VIEW

BATTERY BRACKET

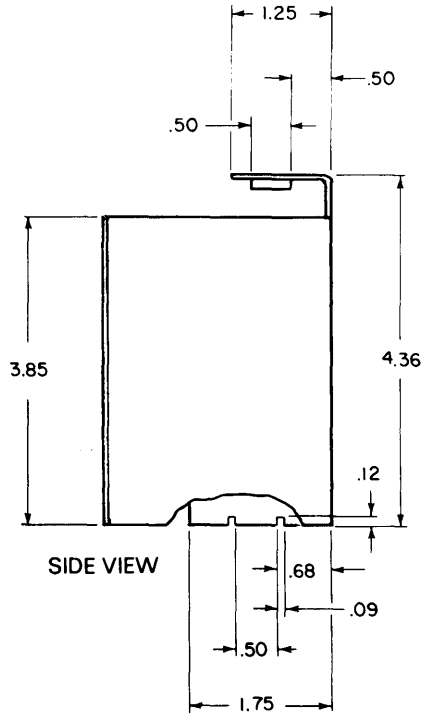


TOP VIEW

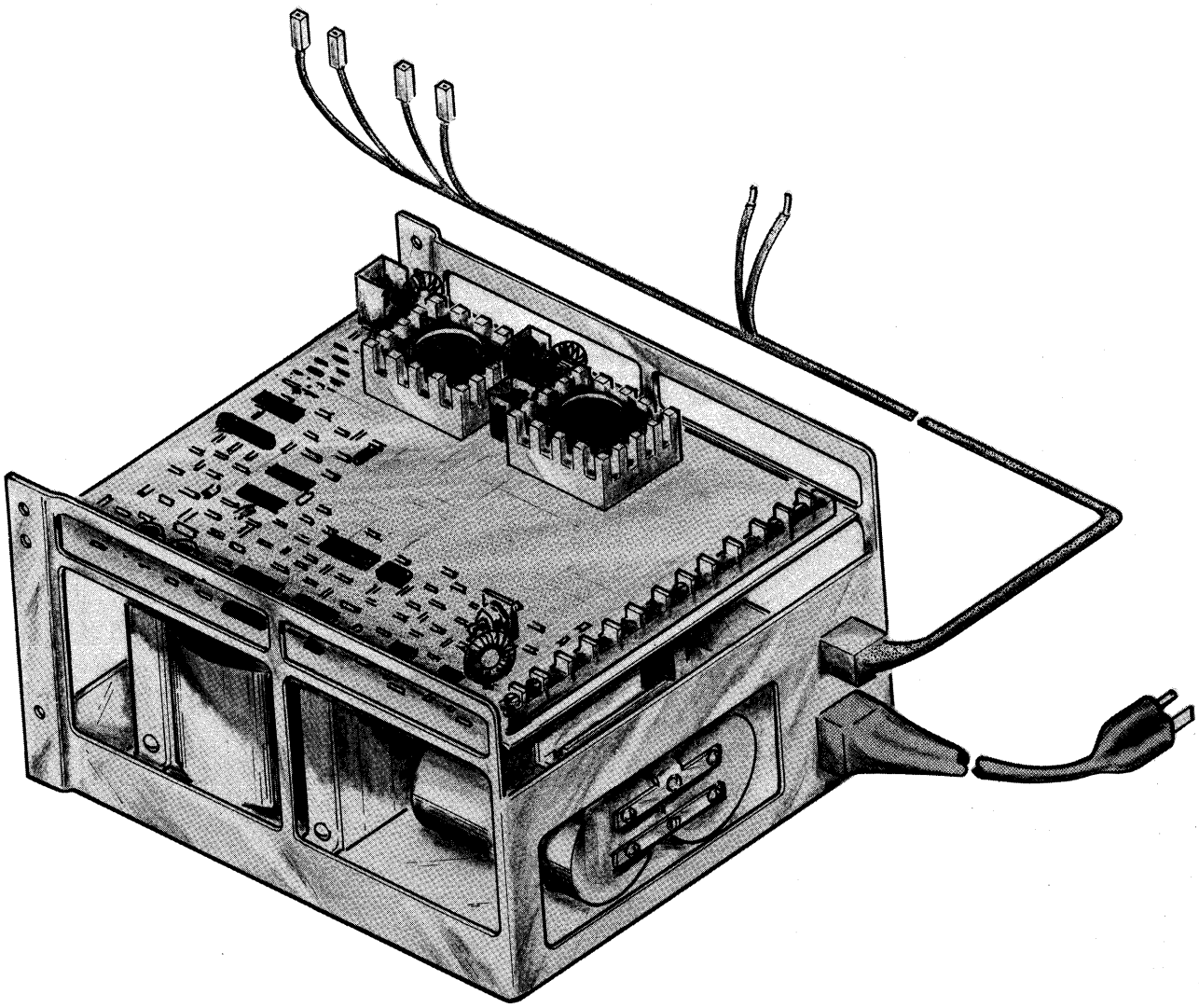


FRONT VIEW

DG-02514



SIDE VIEW



SECTION XII
POWER SUPPLY ASSEMBLIES

The model 4213 power supply provides 18 amps @+5Vdc (± 4.1%), 2.5 amps @15 Vdc (± 1.6%), and 1.8 amps @-5Vdc (± 2.8%). Low voltage switching regulators control all three; both the +5V and the +15V portions by independant pulse width modulation, the -5V portion by variable pulse rate generation. Line voltages of 100, 120, 220, and 240 volts, in the frequency range of 47-63 Hz, are accommodated using transformer taps. Line filtering, overvoltage sensing, independent overcurrent sensing for each output voltage, a power-up sequence, and a 4-way power-fail monitor provide system protection. The model 4214 power supply provides the same outputs but includes a battery charger and a backup battery pack. The battery pack contains two 12V lead-acid batteries connected in series having a capacity of 2.5 amp-hours. An ac line cord and the wiring harness for the microNOVA computer chassis are supplied.

OVERVIEW AND INTRODUCTION

Power supplies in microprocessor systems must be able to maintain regulation under rapidly varying loads, adequately protect their loads from overcurrents and overvoltages, supply a number of dc voltages, typically +5Vdc, -5Vdc, and +15Vdc, and work from the many different ac lines commonly found around the world.

One of the more accurate and efficient ways of regulating a dc voltage in the face of line and load variations is accomplished with a controlled switching regulator. The switching regulators used for two of the voltages (+5V and +15V) in both power supplies are controlled using the pulse width modulating technique (PWM). Regulation is accomplished by changing the duty cycle (on-time divided by the period x 100%) of a fixed period switched dc voltage to compensate for changes in either the input voltage or the output load. The output voltage is maintained constant by varying the duty cycle; increasing the duty cycle increases the output current; decreasing the duty cycle decreases the output current while maintaining the desired output voltage.

The switching regulator used for the third power supply voltage (-5V) is controlled using a variable pulse rate generator. Regulation is accomplished by changing the repetition rate of a pulse of constant width pulsed dc voltage to compensate for changes in either the input voltage or the output load. The output voltage is proportional to the repetition rate; increasing the repetition rate increases (makes more negative) the output voltage; decreasing the repetition rate decreases (makes more positive) the output voltage.

Since the dynamic random access memories (RAMS) used in the microNOVA system lose their contents when deprived of power, the model 4214 power supply is offered with battery backup. The batteries are charged through a controlled rate charging circuit built as an integral part of the power supply.

SUMMARY OF CHARACTERISTICS

AC INPUT:	85-110/102-132/187-242/204-264Vac 47-63Hz
+5Vdc OUTPUT:	Max Load - 18A Line Reg- < 5% Load Reg- <3.2% from .25 to 18A Ripple - <240mV p-p to full load
+15Vdc OUTPUT:	Max Load- 2.5A (Models #4213,4214) Line Reg - <.2% Load Reg - <2% from 0 to 2.0A Ripple - <100mV p-p to full load
-5Vdc OUTPUT	Max Load- 1.8A Line Reg- <2% Load Reg- <3% change from .1A to 1.8A load Ripple- <260mv p-p to full load
SHORT CIRCUIT PROTECTION:	Fuses on +5v Current limiting on all voltages
OVERVOLTAGE PROTECTION:	Trip Point (+5Vdc)-6.65 ± .25 Trip Point (+15Vdc)-16.5 ± .2 Crowbar Blows VNR Fuse On Either Trip Point
UNDERVOLTAGE PROTECTION:	+15Vdc Disabled When -5Vdc Becomes More Positive Than -4±.3Vdc
BATTERY BACKUP:	Switch Over Points Power Down / Power Up } 85% (or lower) of nominal line voltage Switching Time ≤40 msec from last half cycle received Support Time (machine halted) 30 minutes at 25°C for CPU + 32K (8K Memories) Recharge Time 16-20 Hours from Full Discharge
MAX OPR TEMPERATURE:	131 F/55 C
STORAGE TEMPERATURE:	-40° →+185° F/-40° →+85C
WEIGHT:	16lbs/7.3Kg without Battery
DIMENSIONS	5.0"/12.7cm-h 7.94"/20.1cm-w 10.5"/26.7cm-d

BLOCK DIAGRAM

The block diagram on the opposite page provides a simplified representation of the operation of the power supply. The supply can be divided into seven major areas for purposes of discussion. These areas are: ac to unregulated dc conversion, reference voltage generation, +5V regulation, +15V regulation, -5V regulation, power-fail detection, and battery backup implementation. A general discussion of the operation of each of these areas follows.

AC to DC Conversion

Straightforward ac to dc conversion is accomplished using a step down transformer, a full-wave bridge rectifier, and a large smoothing capacitance. An ac line filter and 2 ac fuses are provided between the line cord and the transformer. Both ac fuses (3A, 5A) are used on 220/240 volt systems. The 3A fuse is used to determine maximum input current. The 3A fuse is bypassed on 100/120 volt systems. The line cord selects the various taps on the transformer, determined by the type of line cord, allowing the use of a wide variety of main power sources. The output from this section of the power supply is a +30V unregulated power bus. Excessive voltage on either the +5Vdc or the +15Vdc outputs is detected by a protection circuit which removes the unregulated dc voltage from the rest of the supply. Power removal is effected by the triggering of a silicon controlled rectifier which blows the fuse for the +30V power bus.

Reference Voltage Generation

Reference voltages for controlling the three regulators and the power-fail detection circuitry are generated from the unregulated power bus using various zener diodes. This same network is also used to supply the voltages needed by the various logic elements controlling the regulators.

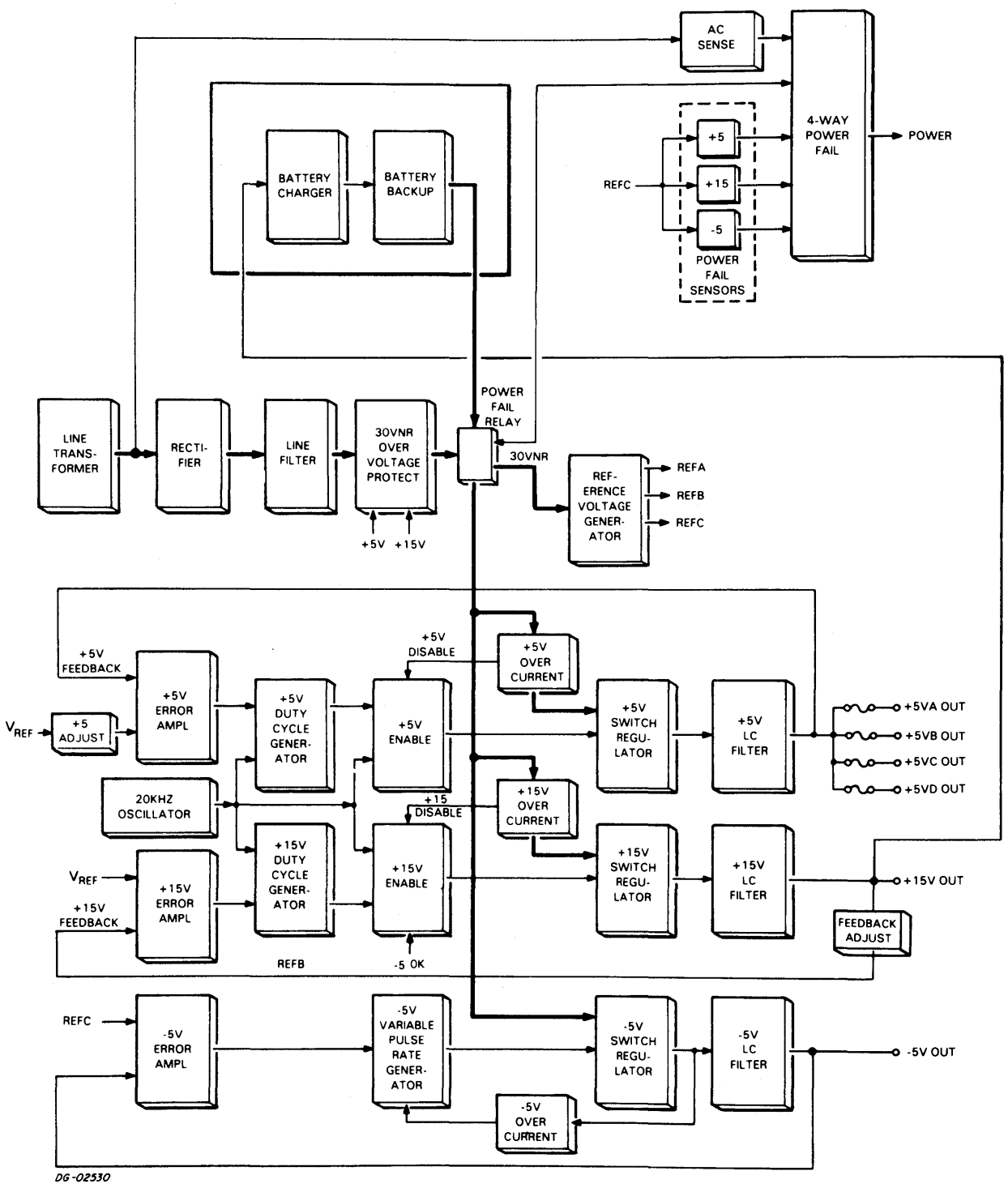
+5V Regulation

The +5V portion of the power supply contains a switching regulator which is controlled by the pulse width modulation technique. This technique is advantageous since the frequency of current switching will be constant, resulting in more predictable response characteristics for the feedback loop.

The regulation is obtained as follows: the unregulated dc power bus is switched through a power transistor to an LC output filter. This filtered output is fed back to an error amplifier which compares the output voltage with a reference voltage. The amplifier, in conjunction with a 20kHz oscillator, determines the duty cycle of the power switching transistor. Since the output voltage is proportional to the duty cycle, the control must increase the duty cycle when the output voltage is too low, and decrease the duty cycle when the output voltage is too high. The waveform of the 20kHz oscillator determines the duty cycle range of the regulator. The range is normally between 12 and 25%. The 20kHz waveform, together with the error amplifier signal, forces the power transistor to assume the necessary on-time during the 20kHz period for proper regulation.

An overcurrent protection circuit senses the peak current from the unregulated power bus flowing into the power transistor. If the current is excessive, a disable signal is generated, turning off the regulator's power transistor. This use of a peak current detector during every cycle provides a nearly constant current, decreasing voltage curve as, seen by the load.

microNOVA POWER SUPPLY



+15V Regulation

The regulation of the +15V portion of the power supply is also accomplished with a pulse width modulated switching regulator. This regulator operates in a similar manner to that on the +5V portion of the supply, and, in fact, shares the 20kHz oscillator with that supply. Overcurrent protection is accomplished as in the +5V supply.

Since damage could occur to some of the integrated circuits used in the microNOVA system if the +15V supply is applied before the -5V supply, the +15V output is disabled whenever the -5V output is more positive than -4V.

-5V Regulation

The -5V portion of the power supply contains a switching regulator which is controlled using the constant width, variable rate pulse technique.

The -5V is obtained from the unregulated power bus by using a coil in the "flyback" mode; i.e., when the coil is charged with a positive voltage and that voltage is removed, the coil reverses polarity (flies back) and a negative voltage is produced.

The regulation is obtained as follows: the unregulated power bus is switched through a power transistor to the flyback coil and output filter. The filtered output is fed back to an error amplifier which compares the output voltage to a reference voltage. The output of this error amplifier controls the constant width pulse generator. This constant width pulse is used to control the switching of the power transistor. If the output voltage is found to be too low (in this case, too negative) compared to the reference voltage, the pulse repetition rate is decreased. If the output voltage is too high compared to the reference voltage, the repetition rate is increased.

An overcurrent protection circuit senses the current flow from the supply. If the current is excessive, the pulse generator is disabled.

Power-fail Detection

The power-fail detection network samples 4 voltages in the power supply: ac line voltage, +5V, +15V, and -5V. If any of these voltages drop below their specified values, a power-fail signal is transmitted to the rest of the microNOVA system. In the case of the ac line voltage dropping, the battery backup portion of the 4214 power supply is cut into the system.

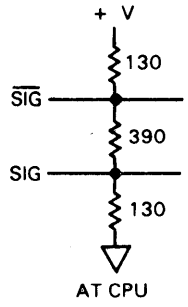
Battery Backup Implementation

The battery backup package in the model 4214 power supply is charged through one portion of the power supply. A variable rate, constant width pulse generator switches the +15V output through a step up transformer to the 24V battery pack. As long as the ac line voltage is present, the battery pack is being charged. If the line voltage drops, the power-fail monitor energizes a relay which switches the battery voltage to the unregulated power bus. The battery will supply power to the power supply until its voltage drops below 20V. This drop in voltage will occur after 2.3 ampere-hours of energy has been drawn from the battery. Battery recharge, for a totally depleted battery pack, requires 16 hours.

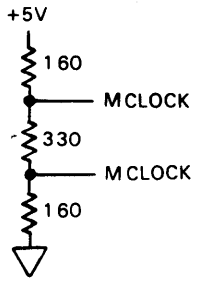
APPENDIX A
I/O BUS CONSIDERATIONS

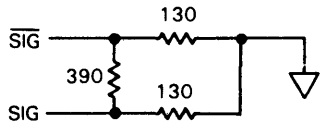
1. The differential I/O lines must be terminated at the CPU (MN629) and at the end of the I/O bus. The CPU terminations are on the CPU board. The termination for the far end of the I/O bus is provided by the terminator module. These terminators are:

All signals except MCLOCK.



For MCLOCK:





END OF BUS

2. The clear line should have a 0.047μF capacitor to ground at the CPU and each I/O device even if it is not used.

3. Gates driving the following signals should be open collector and capable of sinking the designated current: $\overline{\text{BEXTINT}}$, $\overline{\text{BDCINT}}$ (60 ma), $\overline{\text{DCHP OUT}}$, $\overline{\text{INTP OUT}}$ (16 ma), and $\overline{\text{CLEAR}}$ (300 ma).

4. Input priority signals $\overline{\text{INTP IN}}$ and $\overline{\text{DCHP IN}}$ should have a 470-ohm pullup to +5V.

5. The interrupt request lines to the CPU ($\overline{\text{BEXTINT}}$, $\overline{\text{BDCINT}}$) have a 100-ohm resistor to +5 volts at the CPU board.

6. The beginning of the priority chain at the CPU board ($\overline{\text{INTP OUT}}$, $\overline{\text{DCHP OUT}}$) is simply a 100-ohm resistor to +5 volts. Either of these signals can be overridden by an open collector gate (pulling to ground). $\overline{\text{INTP OUT}}$ also has a gate which disables priority during a power fail.

7. At the CPU, the $\overline{\text{CLEAR}}$ line has a 100-ohm resistor to +5 volts and a 0.1 μF capacitor to ground.

This page intentionally left blank.

APPENDIX B
microNOVA - NOVA CONSIDERATIONS

A few differences exist between the microNOVA and the NOVA line computers because of the additional capabilities of the CPU and the different I/O bus.

1. The number of indirect addressing levels is limited to 7 on the microNOVA because of memory refresh timing and not limited on the NOVA/NOVA 3. If this level is exceeded on the microNOVA, the CPU will halt.

2. Device code 1 is reserved for **MUL/DIV** and the stack on the microNOVA and NOVA 3. It is not available to the system designer (along with device codes 0 and 77_8).

3. Locations 46_8 and 47_8 are reserved for the trap instruction in the microNOVA and NOVA 3. If the trap instruction is used, these locations cannot be used for .ZREL code.

4. In the microNOVA and NOVA 3, location 3 is reserved for the stack interrupt vector. On the microNOVA, location 2 is also reserved for the internal real time clock interrupt vector.

5. All arithmetic/logical instructions that specify no load, no skip (bits 12 - 14 = 10_8), are treated as trap instructions in the microNOVA and NOVA 3.

6. The binary equivalent for an **IORST** instruction is different in the microNOVA from the NOVA/NOVA 3. The I/O reset instruction in the NOVA/NOVA 3 is DICC 0, CPU. Because of the I/O structure of the microNOVA, The I/O reset instruction is DOAC 0, CPU. There is a jumper that can be installed on the NOVA 3 to make it compatible with the microNOVA.

7. If the hand-held console is part of the microNOVA system, one of the 16 RAM locations (77577_8) is reserved for the traditional front panel switches. This word can be set/modified using the hand-held console and can be accessed with an I/O instruction (DIA ac, HHC) which has the same function as the NOVA/NOVA 3 **READS**.

8. The microNOVA will honor interrupts when the CPU is halted and the NOVA/NOVA 3 will not.

9. The microNOVA and NOVA 3 will honor data channel requests while the CPU is halted. The NOVAs will not.

10. The real time clock that is internal to the microNOVA has a fixed frequency of 542.5Hz (every 1.8432msec) and the only program control is interrupt enable/disable. The only way the RTC can be processed is by interrupt (location 2.vector), no programmed I/O is supported as in the NOVA/NOVA 3 except to turn RTC on and off.

11. The power fail detection on the microNOVA is different than the NOVA/NOVA 3. On the NOVA/NOVA 3 the detection is accomplished with a skip instruction. On the microNOVA the detection is done by checking the returned value from an INTA instruction. If the specified accumulator is -1 (all ones), then the interrupt was caused by power fail. The skip instructions used by the NOVA/NOVA 3 will be treated as follows:

    SKPDN CPU = NEVER SKIP
    SKPDZ CPU = ALWAYS SKIP

In addition, the remaining skip instructions detect the state of the system interrupt in the microNOVA and NOVA as follows:

SKPBN CPU = SKIP IF INTERRUPT ON
SKPBZ CPU = SKIP IF INTERRUPT OFF

12. On the microNOVA, if a device is not connected to the I/O bus (board out) then any programmed input from that device will return all ones (-1). This means that DIA, DIB, and DIC will receive all ones and that the busy and done flags for that device will always be clear. **SKPDN** and **SKPBN** will never skip and **SKPDZ** and **SKPBZ** will always skip. On the NOVA/NOVA 3 the input data will be zeros. An **INTA**, issued when no device is requesting an interrupt will also load the specified AC with all ones on the microNOVA.

13. On the microNOVA and NOVA 3, only input/output data channel transfers are allowed. There is no increment or add-to-memory options on the data channel as in the NOVA.

14. The interrupt and data channel priority chain on the microNOVA has the opposite sense from the NOVA/NOVA 3. On the NOVA/NOVA 3 a low or 0V priority signal passed to a device meant the device had priority. On the microNOVA, this priority determination is high or +5V. If a board is removed from a NOVA or microNOVA without jumpering the priority chain across the open slot, and a device on either side of the open slot requests interrupt service, the response to the **INTA** instruction is indeterminate since two I/O controllers will try to transmit their device codes. This situation should be avoided.

15. On the microNOVA I/O instructions which affect the CPU Interrupt Request or Data Channel Request lines through an action at the peripheral device controller do not take effect until after the instruction following the I/O instruction causing the action.

    For example, assume a TTY output controller is done and not busy, but its interrupt is masked off due to interrupt mask bit 15 being set. If an MSKO instruction is issued which clears TTY output mask bit, the interrupt will occur after the instruction following the MSKO instruction. In a NOVA/NOVA 3, the interrupt will occur following the MSKO instruction.

    As another example, assume a TTY output controller is done and not busy, but interrupts are disabled in the CPU. If an INTEN instruction is issued followed by an NIOC TTO, the teletype will interrupt the processor following the NIOC instruction. In a NOVA/NOVA 3 the clear pulse will take effect during the NIOC instruction and no interrupt will occur.

16. In the microNOVA, the stack pointer and the frame pointer are always read with bit 0 equal to 0, independent of what is written in those bits. This is not true in the NOVA 3.

CUT ALONG DOTTED LINE

READERS COMMENT FORM

DOCUMENT TITLE: .....

Your comments, accompanied by answers to the following questions, help us improve the quality and usefulness of our publications. If your answer to a question is "no" or requires qualification, please explain.

How did you use this publication?

( ) As an introduction to the subject.
( ) As an aid for advanced knowledge.
( ) For information about operating procedures.
( ) To instruct in a class.
( ) As a student in a class.
( ) As a reference manual.
( ) Other.....

Did you find the material:

- Useful..... YES ( ) NO ( )
- Complete..... YES ( ) NO ( )
- Accurate..... YES ( ) NO ( )
- Well organized..... YES ( ) NO ( )
- Well written..... YES ( ) NO ( )
- Well illustrated..... YES ( ) NO ( )
- Well indexed..... YES ( ) NO ( )
- Easy to read..... YES ( ) NO ( )
- Easy to understand..... YES ( ) NO ( )

We would appreciate any other comments; please label each comment as an addition, deletion, change, or error and reference page numbers where applicable.

COMMENTS

PAGE	COL	PARA	LINE	FROM	TO

From:

NAME..... TITLE.....
FIRM..... DIV. .....
ADDRESS.....
CITY..... STATE..... ZIP.....
TELEPHONE..... DATE.....

Data General Corporation
ENGINEERING PUBLICATIONS
COMMENT FORM
DG-00935

FOLD DOWN FIRST FOLD DOWN

FIRST CLASS
PERMIT NO. 26
SOUTHBORO
MASS. 01772

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL

Postage will be paid by:

DataGeneral
Southboro, Massachusetts 01772

ATTENTION: Engineering Publications

FOLD UP SECOND FOLD UP

STAPLE