JUPITER

JUPITER SYSTEM
DEVELOPMENT PLAN

16 September, 1980

COMPANY CONFIDENTIAL

digital

Jupiter and its four planet-size moons, called the Galilean satellites, were photographed in early March by Voyager 1 and assembled into this collage. They are not to scale but are in their relative positions. Reddish Io (upper left) is nearest Jupiter; then Europa (center); Ganymede and Callisto (lower right). Nine other much smaller satellites circle Jupiter, one inside Io's orbit and the other millions of miles from the planet. Not visible is Jupiter's faint ring of particles, seen for the first time by Voyager 1. The Voyager project is managed for NASA's Office of Space Science by Jet Propulsion Laboratory, California Institute of Technology.

JUPITER SYSTEM DEVELOPMENT PLAN

16 September, 1980

Issued by: Donald A. Lewine MR1-2/E85 DTN 231-6430

Major revision: 2
Edit: 13

Engineering Group Manager
Bill McBride ....

Program Manager
Donald Lewine ....

Product Manager
Rich Fiorentino ....

CPU Engineering
Nat Kerllenevich ....

Technology Engineering
Sultan Zia ....

Diagnostic Engineering
Dick Maliska/Jack Rosen ....

TOPS-10
Peter Hurley ....

TOPS-20
Peter Hurley/Sumner Blount ....

Layered Products
Norma Abel ....

Customer Service
Walter Manter/Max Weinfuss ....

Manufacturing
Charlie Bradshaw/Bill Martel ....

Qualification Engineering
Ron Setera ....

Operations
Roy Rezac/Nick Cappello ....

Communications
Gary Passon ....

GALAXY & DBMS
Dave Braithwate ....

CHAPTER 1

THE JUPITER PROGRAM

1.1 TABLE OF CONTENTS

CHAPTER 1 THE JUPITER PROGRAM

CHAPTER 2 SYSTEM DESCRIPTION

CHAPTER 3 DEVELOPMENT

CHAPTER 4 COST

CHAPTER 5 SYSTEM REALIZATION STRATEGIES

APPENDIX A RELATED DOCUMENTS

1.2 INTRODUCTION

1.2.1 PROGRAM PRIORITIES

When making trade-offs the JUPITER project will consider the following list of primary product goals:

1. Earliest possible time to market (most important)

2. Performance should be 5 to 7.5 X KL1ØE (+/- 15 percent). We will assure that performance is at least 5.ØØ X KL1ØE. The compile times for MR1CD1 (a COBOL program) and SP1111 (a FORTRAN program) run under TOPS-2Ø will be used as the benchmark to measure logic performance.

3. FORTRAN performance with the APA will be at least 5 X KL1ØE. This will be measured as the execution time for SP1111. Without the APA FORTRAN execution speed will not be much faster than 2 X KL1ØE and some heavy double precision programs may run slower than the KL1ØE.

4. Lowest possible cost of ownership. This includes the cost of the JUPITER, the cost of field service and the impact of downtime.

5. Provide for coexistence with VAX.

In addition there are the following secondary JUPITER product goals:

1. Provide adequate microcode address space and storage to allow for future functionality or performance improvements.

2. Perform comprehensive error detection at the interfaces between modular units to prevent errors from propagating into the rest of the system without detection.

3. All detected errors will be logged in non-volatile storage.

4. Provide remote diagnostic capability greater than that provided on the KL1Ø or KS1Ø.

1.2.2 PROGRAM HIGHLIGHTS

1.2.2.1 SYSTEM - This plan addresses the MASSBUS based JUPITER systems to be delivered during the first year of product availability. The deliverable configuration is:

Disks:	RP06 RP20
Tapes:	TU72 TU78
Line Printers:	LP05 LP14 LP07
Card Reader:	CD20
Async Interfaces:	DN25 (DZ11-based interfaces)
Intercomputer:	CI links to other JUPITER systems to support a common file system.
Software:	TOPS-20 based multi-terminal multi-language operating system

This plan does not address any post FRS features. These will be covered in a JUPITER STAGE-II system plan to be issued later this year.

1.2.2.2 TECHNOLOGY -

Press pin, multilayer, controlled impedance back- planes (approximately 18 layers)

MULTIWIRE modules will be used where their improved density and quick turnaround improve time to market. 4 Layer Printed Circuit modules will be used where density and schedule permit.

ECL 100K MSI/SSI chips

Air cooled

1K and 4K ECL RAMs

64K MOS memory chips (Memory array module will also accept 16K and 256K parts)

Corporate "switching regulator" modular power supplies

1.2.2.3 SCHEDULE -

MILESTONE	50% DATE	90% DATE
CPU poweron	JAN 81	APR 81
CPU runs	MAR 81	SEP 81
LOGIN on CTY	JUN 81	JAN 82
Breadboard UETP runs	JUL 81	MAR 82
Proto Power on	NOV 81	AUG 82
Start DMT	FEB 82	NOV 82
Field Test	FEB 82	NOV 82
First Ship	AUG 82	JUL 83

More detail as well as the rationale for 50% and 90% dates is contained in section 3.5.

CHAPTER 2

SYSTEM DESCRIPTION

NOTE

The JUPITER system is described in considerable detail in the "2Ø8Ø ENGINEERING FUNCTIONAL SPECIFICATION". This chapter is only a brief summary of the system.

2.1 OVERVIEW

2.1.1 PRODUCT STRATEGY

The JUPITER provides the basic upgrade path for current TOPS-1Ø and TOPS-2Ø customers. The JUPITER also will allow 36-bit customers to coexist with VAX by using the corporate interconnect strategy.

2.1.2 CPU CLUSTER BLOCK DIAGRAM

```
           +-----------+        +------------+         +------------+
           !ARITHMETIC !        !            !         !            !
           !    HOT    !<=======!    EBOX    !<=======>!    IBOX    !
           !    BOX    !        !            !         !            !
           +-----------+        +------------+         +------------+
             !   ^                 !   ^  !                !  ^
             !   !                 !   !  !                !  !
             !   +----------+------!---+-------------------!--+ DATA BUS
             !______________!______!______!________________! ADDRESS BUS
                            !    !        !
                      +-----+-------+     !
CONNECTIONS TO DIAG   !             !     !
LOGIC ON EACH BOARD   !    MBOX     !     !
       ^              !             !     !
       !              +-------------+   +-------+
       !                     !          !  ECL  !
       !                     +----------!  TO   !=======================
 +------------+              !          !  TTL  !    !     !     !     !
 !            !       +-------------+   +-------+    !     !     !     !
 ! DIAGNOSTIC !       !             !       ^        !     !     !     !
 !  CONSOLE   !       !   MEMORY    !       !    +-------+ ! +-------+ !
 !            !       !             !       !    !  ICCS ! ! !  IBM  ! !
 +------------+       +-------------+       !    !  PORT ! ! !  PORT ! !
        !                                   !    +-------+ ! +-------+ !
        +-----------------------------------+              !           !
                                                           !           !
                                                      +----+      +----+
    +-------+  +-------+  +-------+                   !           !
    ! CARD  !  !  LINE !  ! ASYNC !   +-------+  +-------+    +--------+
    !READER !  !PRINTER!  !       !   ! DN20  !  !  UBA  !    !  MBA   !
    +-------+  +-------+  +-------+   !       !--!       !    !        !
       ^           ^          ^       +-------+  +-------+    +--------+
       !-----------!----------!-----------!                       !
                                                                  !
                                                              +-------+
                                                              ! DISK  !
                                                              !       !
                                                              +-------+
```

NOTE

Some details (e.g. packet buffers, port modules) are omitted for clarity

2.1.3 SYSTEM CONFIGURATIONS

2.1.3.1 INITIAL SHIPMENTS - In order to reduce the risk and schedule for the JUPITER system, the configurations are being split into two phases. First is a MASSBUS and UNIBUS based system to support shipments during the first year of production. Second is a CI based system targeted for the majority of the JUPITER product life.

The MASSBUS based system will support the following peripherals:

Disks:	RP06 RP20
Tapes:	TU72 TU78
Line Printers:	LP05 LP14 LP07
Card Reader:	CD20
Sync Interfaces:	DMR11-AA (2400 baud to 19.2 KB) [*] DN21-BA (19.2Kb to 56Kb) DMR11-AE (Speeds up to 1Mb) [*]
Async Interfaces:	DN25 (DZ11-based interfaces)
Intercomputer:	CI links to other JUPITER systems to support a common file system.

2.1.3.2 TARGET SYSTEMS - Here is a picture of a possible dual JUPITER configuration which should give a good idea of the sorts of things that can be done. There are two JUPITER systems hooked together with ICCS. Each JUPITER has an HSC50 which can access a number of disks. Terminals are attached using a MERCURY communications controller on the ICCS bus.

In practice, this system would have more ICCS lines connecting the JUPITER'S to the various I/O controllers.

- - - - - - - - - - - - - - -

[*] Sharon Passon (Distributed System Product Manager for LSG Communications) has indicated that new communications options will be called by their corporate name not DN2x-xx. See LSG COMMUNICATION HARDWARE PRODUCT REQUIREMENTS dated July 31, 1980

```
        +----------+                            +----------+
        !          !---+----------+---------+---!          !
        !          !   !    +---------+     !   !          !
        ! JUPITER  !   !    ! MERCURY !     !   ! JUPITER  !
     +--!          !   !    +---------+     !   !          !
     !  !          !   !     ! ! ! ! !      !   !          !
     !  +----------+   !     ! ! ! ! !      !   +----------+
     !                 !     TERMINALS      !
 +---------+       +-------+            +-------+
 ! MERCURY !       ! HSC5Ø !            ! HSC5Ø !
 !  COMM   !       +-------+            +-------+
 +---------+           !      +------+      ! !      +------+
  ! ! ! ! !            !      !      !      ! !      !      !
  ! ! ! ! !            +------! DISK !------+ +------! TAPE !
  TERMINALS            !      !      !      !        !      !
                       !      +------+      !        +------+
                       !                    !
                       !      +------+      !
                       !      !      !      !
                       +------! DISK !------+
                              !      !
                              +------+
```

NOTE

The plans for the MASSBUS based JUPITER system are quite solid. Plan for CI based systems are still in considerable flux.

2.2 HARDWARE ORGANIZATION

This section provides a summary of the major blocks in the JUPITER CPU cluster. Full details of each block will be found in "2Ø8Ø ENGINEERING FUNCTIONAL SPECIFICATION".

2.2.1 CONSOLE SUMMARY

The console performs the traditional "lights and switches" function on the JUPITER system. It performs three functions for the JUPITER system: bootstrap, operator support, and diagnostics.

2.2.1.1 BOOTSTRAP - This function turns the machine into something that the software bootstrap can use to load the monitor or diagnostics. The following actions need to take place:

1. Reset the JUPITER and verify that nothing is obviously broken.

2. Load the microcode (IBOX, EBOX, MBOX, PORT).

3. Load and start a pre-boot from disk or tape.

2.2.1.2 OPERATOR SUPPORT - This mode provides several simple commands to start and stop the system. It also provides operating system support for the console terminal so the operator can talk to the operating system.

2.2.1.3 DIAGNOSTIC SUPPORT - The console can control the hardware and monitor outputs to enable programs to execute in the console and diagnose the machine.

2.2.2 IBOX SUMMARY

The part of the JUPITER most responsible for its speed, (and complexity) is the IBOX (or instruction unit). The purpose of the IBOX is to pre-fetch instructions and operands, so that the EBOX (or execution unit) will have a steady stream of instructions to execute.

The IBOX attempts to gather up to 16 instructions and operands ahead of the EBOX, so as to smooth the flow between the memory system and the EBOX, both of which have considerable variation in their operation times.

2.2.3 EBOX SUMMARY

The EBOX does all of the "real" computation in the JUPITER. The EBOX has a wide microcode word which directly controls all of the logic in the EBOX.

The EBOX does all of the stores required by the program. This keeps everything in sync so that we can recover from a page failure.

2.2.4 ARITHMETIC PROCESSING ACCELERATOR SUMMARY

The JUPITER APA monitors the IBOX and EBOX and speeds up selected operations. These include single and double precision floating point add, subtract, multiply and divide and single precision integer multiply and divide.

The following classes of instructions will be processed in the APA:

1. Single Precision Floating Point

2. Single Precision Integer

3. Double Precision Floating Point

4. Expanded Range Double Precision Floating Point

5. Conversion Instructions

2.2.5 MBOX SUMMARY

The MBOX is the central block of the JUPITER system. It contains the cache memory and all of the hardware required to do paging and to read and write physical memory.

The MBOX has a microprocessor which handles many of the complex MBOX functions (like a cache refill). Simple virtual memory reads do not require the microprocessor to do anything.

2.2.6 MEMORY SUMMARY

The JUPITER has internal 64K MOS memory. The memory has ECC on every word. The memory always reads 4 words at one time. These four words are latched on the memory array cards and sent serially to the MBOX where they are written into the cache.

Each JUPITER memory backplane has room for 16 memory modules for a total of 4 million words of memory per backplane. The JUPITER is being designed to allow for two memory backplanes for a total of 8 million words, however, the memory bus repeater required to use the second backplane is not being designed. This limits the maximum memory at FRS to four million words.

2.2.7 I/O SUMMARY

The JUPITER uses a microprogrammed IO processor called a "PORT". The port performs several common functions for all JUPITER IO interfaces:

1. Interface to the JUPITER TTL bus.

2. Data repacking. In particular, conversion of 36 bit words into 8 bit byte streams in any one of several formats.

3. Command processing. The port maintains several queues of outstanding commands and schedules data transfers.

4. Processing channel command lists and buffer descriptors.

5. Interrupt processing.

There are up to eight ports in the JUPITER. Each port can control any one of four types of IO interface:

1. ICCS link - connection to other processors, HSC5Ø[*] and MERCURY[*].

2. MASSBUS - Connection to current disks and tapes (e.g. RPØ6, RP2Ø, TU78).

3. UNIBUS (or 1Ø/11 interface) - connection to DN2Ø-style communications options.

4. IBM BLOCK MUX - connection to PCM disks and tapes[*].

[*] HSC5Ø, MERCURY, and the IBM block multiplex channel will not be available for first customer ship.

CHAPTER 3

DEVELOPMENT

3.1 DEVELOPMENT STRATEGY

3.1.1 BREADBOARD

Engineering will build two breadboards that will run at full speed and will allow complete verification of system functionality. It is expected that the functionality of the breadboard will be equivalent to that of the final machine, but revisions will be made as necessary. This stage in the development will enable Diagnostics and Software to begin debug. It is planned that people from Manufacturing, Customer Service, and System Qualification will participate; they will begin learning the system in terms relevant to their own tasks, and will be actively involved in identifying problem areas where Engineering can help them.

The breadboards will use backplanes which are partly etch and partly wirewrap and a mixture of printed circuit and multiwire modules.

3.1.2 PROTOTYPE

Engineering will build prototype machines that will be very similar to the breadboards. Most of the backplane runs will be converted to etch and all design errors will be corrected. These prototypes are allocated as follows:

Proto #	Use
1	Hardware Engineering
2	TOPS-20
3	Diagnostics
4	DVT
5	DEC 102, FCC, etc. [This machine will be expensed.]

6 MEG [If capital is approved]

7 and 8 Manufacturing QV stations

People from Manufacturing, Customer Service, and System Qualification are expected to be deeply involved in these activities. As is the case with the breadboards, the prototypes will be built and driven by Engineering.

At this stage, System Engineering will use the prototypes to verify a limited number of system configurations. Since the initial prototype systems will be built and verified without released documents, we will set up an account for purchasing components and mechanical parts and to track actual costs instead of standardized costs.

3.1.3 PILOT

Manufacturing will build pilots and will then update them to reflect the documentation as it is released. The goal is for these to be functionally identical to the Engineering prototypes. The Manufacturing pilots will be built with Engineering support and maintained by Customer Service.

Since the initial pilot systems will be built and verified without released documents, we will set up an account for purchasing components and mechanical parts and to track actual costs instead of standardized costs.

3.2 SOFTWARE STRATEGY

3.2.1 GOALS

The most basic goal of the JUPITER system is to support the current user base of DECsystem-10 and DECSYSTEM-20 customers by providing cost effective hardware to run their current software and protect their software investment. This goal implies the following:

1. All current KL10 based software will operate correctly on the JUPITER without modification or recompilation or alteration. There are some programs which will not run without modification. These are:

    - Programs which depend on the KL10's speed.

    - Some privileged programs which depend on hardware or software features unique to the KL10's machine organization. These programs include GALAXY and some user mode diagnostics.

- Programs which depend on hardware or software not supported on JUPITER systems (e.g. TU45s or RSX20F line printers).

2. All of DIGITAL's supported layered products will run without modification on JUPITER systems, however, they will have to be modified to take advantage of features unique to the JUPITER (e.g. NCIS, One word global byte pointers).

3. Customers' data is completely transportable between KL10 systems and JUPITER systems.

An additional goal for the JUPITER software is support for all of the hardware RAMP features and error-recovery included as part of the system design.

3.2.2 TIMING

Because machine time on breadboards, prototypes and pilots is expensive and therefore limited, we can not develop both TOPS-10 and TOPS-20 at the same time. TOPS-20 will be released with the initial JUPITER systems and TOPS-10 will come out about one year later than FRS.

3.3 DEVELOPMENT TOOLS

Being developed internally are a number of programs, many very large and complicated, to aid in hardware design. Although many of these tools are available throughout the Corporation, most are being developed or enhanced specifically for the JUPITER Program by the LSG CAD Group. These programs are generally referred to as CAD tools, for "computer aided design".

Basic Circuit Design

The fundamental CAD tool is SUDS, the Stanford University Design System. This is based on a sophisticated graphics editor that aids in the design and checking of logic circuits and drawing of circuit schematics. It also contains programs to create wirelist and plot files. The outputs of SUDS provide the inputs to CALDEC, IDEA, and most of the other software discussed below. The program also has facilities for interacting with the various intermediate products of the board and backplane layout procedures.

SAGE2 Simulator

From information provided by the SUDS wirelist file, SAGE2 simulates the hardware of individual 100K logic components with the ability to inspect the interaction between individual gates in real time (although many times slower than actual gate speeds), to determine whether the logic actually does what it was designed to do.

VOTE Simulator

This program determines the effectiveness of the test patterns, test programs and test microcode generated by Diagnostics and Manufacturing. VOTE effectively runs the tests on a simulation of the hardware with the equivalent of physical fault insertion done entirely in software, and from this determines which of the faults the test was unable to detect.

ISPS/TUMS

This collection of programs is used to allow simulation of microcode. ISPS is a machine description language which allows the simulator writer to describe how the hardware works using a powerful higher-level language. TUMS converts the ISPS description of the JUPITER into a simulator.

The ISPS/TUMS combination allow easy modification of the simulator as the machine design evolves.

IDEA

This program is the successor to CALDEC. Like the earlier procedure, it uses SUDS outputs to lay out circuits, but it is more advanced and handles many more layers.

Delay Calculation

This software package allows the designer to determine the physical delays between individual signal points in a circuit design, either a single board or a set of boards in a backplane. From the SUDS wirelist file and the CALDEC output (eventually the IDEA output), DLY creates a database that represents the physical hardware as it would be built, and from that CAL calculates all signal propagation times taking into account gate delays, wire links, and even stubs. Then with DLYED, the user can determine the delay structure of his design by inspection of propagation times across individual elements in each signal path.

PINCUT

This program helps the designer determine the optimal position of logic on a module.

Wirewrap

From wiring rules and from information about the special character of runs, their type and termination supplied by the user, the wirewrap package generates the pattern for wirewrapping a circuit board or backplane, including assigning twisted pair grounds and generating an NC tape.

Multiwire

This assortment of programs, including parts of SUDS, PINCUT and the wirewrap package, assists the designer in laying out a multiwire board and generates the materials for the multiwire vendor to produce it.

3.4 PHASE REVIEWS

The JUPITER Program will follow the "Product Development Process" used by the Large System Group in Marlboro.

During Phase 2 the Program will accomplish the following major tasks.

Renew the commitment of Engineering, Manufacturing, Marketing and Customer Service to the revised Product Business Plan.

Hold technical reviews of the engineering design, manufacturing process, and service process.

Complete the detailed design, hardware breadboard and prototype test, and software internal tests; at the end of Phase 2, DMT and field testing will be ready to begin.

Run performance benchmarks and publish a performance report.

Make sure announcement criteria have been met.

3.5 MAJOR MILESTONES

The phase review process discussed above defines a number of major stages covering the entire history of a product. But to be able to determine the exact status of the Program at any time, and thus to know whether it is on target and to identify problem areas for taking corrective action, the entire development Program - at several levels - is continually undergoing a tracking procedure.

Throughout the course of the Program, there will be a number of particular events or milestones that must occur at specific times if the overall goals of the Program are to be met. Keeping track of the actual dates of these events as against the target dates given here will provide a very good indication of whether the Program is on schedule and where any trouble may lie.

Two schedules are maintained. One, called the 50% schedule, is a good indication of what can be done if everything goes well and required resources are in place. The other, called the 90% schedule, includes extra time to allow for things to go wrong. Engineering[*] is attempting to meet the 50% schedule and making sure that all tasks on which we depend happen on schedule. All outside groups which depend on engineering, such as product lines, should use 90% dates to make sure that they are not disappointed.

Events which are near are easier to see than those which are far away, therefore, the gap between the 50% schedule and the 90% schedule is

[*] In this context "engineering" is meant in it most general form and includes not only the direct engineering groups but MEG, CSSE and manufacturing start up.

larger for events further into the future. Here are the rules for adding uncertainty:

1. Three months is added to the design and layout phase. This will be deleted at breadboard CPU poweron.

2. Three months is added to CPU/TOPS-20 debug. This will be deleted at "LOGIN on CTY".

3. One month is added for additional CPU/TOPS-20 breadboard debug. This will be deleted at "breadboard UETP runs".

4. One month is added for prototype build. This will be deleted at prototype poweron.

5. Two months are added for DMT. This will be deleted at DMT complete.

MILESTONE	50%	90%	DONENESS CRITERIA
CSL ready	NOV 80	FEB 81	The console module is built and ready to start debug of console and diagnostic software.
MBA ready	DEC 80	MAR 81	The MBA and microcode are ready to be debugged. Debug can not start until the PORT and IOBOX run.
Basic microcode ready	DEC 80	MAR 81	All microcode written to perform all functions found on KL10 except BIS and DP floating point.

PORT ready	JAN 81	APR 81	The PORT is built and ready to plug in.
Backplanes ready	JAN 81	APR 81	CPU, IO and memory backplanes are built and ready.
IOBOX ready	JAN 81	APR 81	IOBOX is built and ready to plug into the machine.
CPU poweron	JAN 81	APR 81	All CPU modules built plugged into the backplane and power applied.
CPU runs	MAR 81	SEP 81	CPU runs all instructions.
MBA runs	MAR 81	SEP 81	MBA runs the diagnostic and is able to seek, read and write an RPØ6 disk.
UBA ready	MAR 81	JUN 81	UBA is built and ready to plug in.
CPU Debugged	APR 81	OCT 81	Microcode for all functions present in the KL1Ø complete and runs all test which exist. This is the point where monitor debug can begin full time.
LOGIN on CTY	JUN 81	JAN 82	First LOGIN under TOPS-2Ø on the CTY.
Breadboard runs UETP	JUL 81	MAR 82	Breadboard hardware and software run the UETP test package (the same one used on KL1Øs and KS1Øs) for 4 hours without error.
Prototype Poweron	NOV 81	AUG 82	All prototype modules available, plugged in and power applied.
Prototype runs UETP	DEC 81	SEP 82	Same as breadboard runs UETP except 24 hour test.
Start DMT	FEB 82	NOV 82	4 machines in chamber being tested.
First Revenue Ship	AUG 82	JUL 83	First shipment to a customer who is expected to pay.

3.6 RISKS AND DEPENDENCIES

3.6.1 RISKS

Following are some of the more critical risks, whose impact on the Program could be severe. In each case whatever actions can be taken to lessen the risk are indicated, and backup strategies are considered wherever appropriate.

Multi-Signal Layer, Controlled Impedance PC Boards and Backplanes

Boards and backplanes with multiple signal layers are absolutely necessary to provide sufficient component interconnectivity to allow the required component density. Such units are in general use, but no vendor can supply them in the quantity needed or at a suitable price.

Schedule

The overall schedule is dependent upon the timely meshing of a multitude of individual tasks, and a delay in any one of them is quite likely to have at least some impact on the Program as a whole. Some major risks are unavailability of 1K or 4K ECL RAMs in the required volume in the timeframe set by our schedule, and any unforeseen contingencies such as illness or accident. Another problem is resource conflicts with the VENUS Program in the areas of site support and Manufacturing. However as explained in the preceding section, we have devised mechanisms for tracking the Program to a very fine degree. Hence should any unforeseen event occur, we at least will know exactly what effect it will have on the various parts of the Program, and we can thus take remedial action.

Transfer Cost

Ninety percent of the transfer cost is in materials alone, and in many cases it is the high risk technology itself that is critical to meeting cost objectives. Manufacturing is naturally pursuing all avenues for arranging the best possible terms for purchase of the needed materials, and is also investigating any savings that would result from bringing processes in-house rather than purchasing from vendors.

FCC Regulations

Current FCC rulings mandate testing, correction and documentation of all computer products relative to new requirements pertaining to radio frequency emissions resulting from the high speed switching in data processing equipment and electronic pollution of power lines resulting in ineffective or improper line filtering. This will require a major program for LSG. (Legal relief is being sought, but at best it would only provide a delay and may not be granted at all.) In preparation for the effective date of these regulations, the Technology Group has investigated the matter and issued a report outlining the kind of program required for both new and old products, its cost, and the cost and availability of test equipment and facilities. The report also lists current products that will have to be tested and corrected, although it is anticipated that many of the older products will fall under some sort of grandfather clause.

Preliminary testing with a KL1Ø indicates that we will have a problem in this area.

Functional Tests

The current budget and schedule do not permit us to write all of the functional tests that are desirable to perform a complete verification of the JUPITER architecture. This increases our risk of an ECO after a large number of JUPITER systems have been built.

3.6.2 DEPENDENCIES

Besides the major risk areas that could have such an adverse effect, the Program and its various parts are dependent for their fulfillment on the performance of many other people and groups throughout the Corporation. Here are some examples.

The design and layout of pc boards and modular backplanes are heavily dependent on the development of very sophisticated software, which in turn is heavily dependent on computer time and personnel to do the job. In addition to schedule risks, there are also risks in the processes themselves. For example, the representations of some complicated designs may be too cumbersome even to handle.

Diagnostics is dependent on the VOTE simulator being available on schedule and having the expected functionality; a failure here could prevent verification of fault coverage.

CHAPTER 4

COST

The major cost categories that must be considered are the cost of developing the product, the cost of building it, and the life cycle

4.1 DEVELOPMENT COST

The development budget given on the next page includes material expenses to cover building two breadboard and five Engineering prototype systems, creating all diagnostic programs, and releasing twenty three L modules and four backpanels to Manufacturing. The budget does not however include funding for building the pilot systems. It is estimated that the Manufacturing prototypes (to become pilots) will cost approximately $8ØK each, and we recommend that X95 manufacturing accounts be opened to capture these costs. Individual user groups must capitalize and depreciate their machines, as the Corporation will charge them off to the various cost centers over five years.

TASK	FY81	FY82	FY83	TOTAL	DESCRIPTION
CAD Tools	119	132	145	396	Develop and maintain a set of CAD tools to allow the JUPITER project to move smoothly into production.
Memory	206	100	25	331	Design the JUPITER memory array, a memory array tester and a process to allow JUPITER memory modules to be built in the far east.
Packaging	353	300	147	800	Design the cabinets and other mechanical parts of the JUPITER system. This includes thermal and acoustical work but does not allow for a redesign of the corporate cabinet to meet FCC rules.
Qualification	52	152	63	267	Confirm that JUPITER lives up to its design specifications.
System design	1646	1811	1700	5157	Design the JUPITER CPU, IO Cluster. Write microcode for everything.
Technology	447	508	475	1430	Bring ECL 100K technology in house. Determine wire rules and define purchase specifications and test procedures for all parts.
Materials	250	600	50	900	This is direct expense related to building breadboards and prototypes. It does not include the cost of operating engineering machines.

TOPS-20	355	390	429	1174	This is the cost of converting TOPS-20 to operate on the JUPITER hardware. This includes the software required to do a common file system using CI hardware.
COMM Software	161	177	195	533	This is the cost of converting the front-end DECnet software to run with the JUPITER UBA as well as adding local Async communications.
TOTAL	3589	4170	3229	10888	An additional 1899K was spent in FY79 and FY80 so the total engineering expense is estimated at 12747K.

These costs do not include the following:

1. The cost of computers and the operations and field service support required to operate them.
2. The cost of MEG and CSSE labor.
3. The cost of manufacturing engineering labor.
4. The cost of manufacturing startup and process specific capital.
5. The cost of training customers, field service and software support.
6. The cost of any GALAXY work which may be required.
7. The cost of software release engineering.

4.2 MANUFACTURING COST

The manufacturing costs were computed on August 13, 1980 using the following assumptions:

1. Cost of 100K chips are based on latest quotes from Fairchild. Usage of 100K parts is based on latest engineering estimate.
2. Cost of ECL 100K chips, and all memory parts are projected in 1982 dollars.

3. MULTIWIRE modules are projected to cost $290 each in 1982.

4. Assembly time is based on hand insertion of components.

5. The cost of battery backup is NOT included.

6. There is no provision for ECOs in any area.

7. No manufacturing and/or field service spares have been considered.

8. These cost do not allow for any change in management philosophy or policy.

9. Module burn-in and thermal cycle is assumed.

10. Diagnostics are assumed to fault isolate to a given module. QV diagnostics can isolate to networks with greater than 95% fault detection.

I Box	5550		
E Box	11577		
M Box	8396		
I/O Box	1077		
Console	856		
CPU subtotal		27456	
512KW Memory	7440		
Backplanes	3184		
Power	4912		
cabinets	3393		
Total Packaging		11489	
CPU, Memory, Packaging		46385	
UBA	563		
2 MBAs	1430		
3 Ports	3738		
2 RL02s	2355		
LA34	600		
VT100	500		
Total I/O		9286	
Assemble and Test		5072	
Total basic cluster			60743

CHAPTER 5

SYSTEM REALIZATION STRATEGIES

It is one thing to define a logical system and even to develop it into a physical entity, but then the result must be made a reality - it must be manufactured as a marketable, maintainable product. Following are the strategies for accomplishing this objective.

5.1 MANUFACTURING STRATEGY

5.1.1 MANUFACTURING GOALS

1. Be a prime contributor to the success of the JUPITER project through a given process strategy in producing a highly reliable and cost effective product.

2. To devise the proper methods and techniques, select the proper capital facilities required to meet the demands of the new technologies.

3. Optimize available management controls for standards, WIP cycles, schedules, and inventory levels in maintaining delivery schedules.

4. Develop an early strategy for producibility and testability at all levels.

5.1.2 MATERIAL PLAN

The material group philosophy for the JUPITER project will not vary drastically from the current techniques and processes of material procurement, storage and disbursement.

There are five major inputs to Material Requirements Planning:

1. The scheduled forecast
2. The build schedule
3. Inventory information
4. Bill of material information
5. Purchasing information

From these inputs one can correctly order and schedule the required material.

5.1.3 MANUFACTURING ASSEMBLY PROCESS

5.1.3.1 STRATEGY - The JUPITER manufacturing strategy is currently planned to enter three separate phases with the objectives of early awareness for training, methods and techniques development in establishing processes and developing standards allowing for efficient and cost effective operation. These three phases are:

1. Prototype build and test.
2. Pilot production build and test.
3. Steady state or Volume production build and test.

The early involvement with engineering will assist manufacturing in developing a nucleus work force which can be utilized in guiding and training new personnel long before the volume production schedule.

5.1.3.2 ENGINEERING RELEASE MECHANISM - An engineering release activities plan was generated by engineering site services, identifying major steps to be taken to affect a complete release.

5.1.3.3 VOLUME PLANT - The JUPITER is currently planned to be produced in the volume segment of the Marlboro plant.

The Marlboro facility, with some new capital adjustments, has the capability and capacity to produce JUPITER systems to the current marketing forecast.

5.1.3.4 PLANNED PROCESS FLOW -

5.1.3.4.1 ASSEMBLY - The following is proposed at this time:

1. Sequential build rather than batch build.

2. Effective use of subassembly build to plug into the main assembly line.

3. Detailed process sheets for each build station.

4. Effective use of ramps, conveyers and power tools.

5.1.3.4.2 MODULES - All modules, except for the ICCS link, will be produced in the volume segment of the Marlboro manufacturing facility.

5.1.3.5 MAJOR CAPITAL EXPENDITURES - The following equipment is required to support the production of equipment as currently conceived by design and technology engineering:

1. Incoming inspection and test of 1ØØK logic will require a J325/S357 tester at a cost of 577K.

2. Component prep, special storage equipment and static eliminater will cost 35K.

3. Upgrade of semi-automatic insertion equipment will cost 189K to cover both MULTIWIRE and etch modules.

4. New module testers to handle ECL 1ØØK and higher density will cost 1ØØ9K.

This equipment has a total cost of about $1.8 million dollars.

5.2 QUALIFICATION

This task takes the steps required to ensure that JUPITER is a qualified product. To be qualified the system must:

1. Adhere to Engineering functional and performance specifications.

2. Achieve product cost goals.

3. Achieve MTBF and MTTR goals.

4. Undergo a formal DMT and PMT to prove that the system lives up to defined RAMP goals.

Qualification approaches the product from a system level to prove that this system can perform the functions previous systems have and to build confidence in areas of risk. A risk area is any as yet unproven functionality or technology. To reduce exposure to risk, vulnerable areas will be identified and provisions made to test them. A heavy emphasis will be placed on ensuring that all system RAMP features are achieved.

Testing procedures fall into two general categories, system qualification and product performance. The schedule for these tasks is determined principally by the schedule for the Program as a whole.

5.2.1 SYSTEM QUALIFICATION TESTS

These are tests to confirm that JUPITER lives up to its design specifications in the areas of sensitivity to various conditions, conformance to DEC standards, functionality of various procedures and features, and characteristics of the overall system. Tests are performed in the following categories.

1. Sensitivity of the system to margins in voltage, temperature, humidity, and clock rate.

2. Sensitivity of environmental and power sensors.

3. Conformance of mechanical characteristics to international regulations and DEC Standards 102 and 119.

4. Conformance of electrical characteristics to international regulations, DEC Standards 102.7 (particularly EMI/RFI), 122 and 123, and FCC requirements.

5. Verification of performance under voltage margins.

6. Verification of all power up/down sequences.

7. Verification of initialization and configuration routines utilized to prepare the system for bootstrapping.

8. Verification of the bootstrap program via all load paths.

9. Verification of software support of the hardware (i.e. functionality of the operating system).

10. Verification of the orderly shutdown of the operating system.

11. Verification of system RAMP features.

12. Determination of the sensitivity of the hardware and software to variations in system configuration.

13. Determination of the reliability of the hardware and software, especially RAMP features, through the design maturity test (to measure MTBF) and a software load test.

5.2.2 PRODUCT PERFORMANCE TESTS

Hardware and software benchmarks will be run to measure the performance of the system against existing Digital computers and those of the competition.

5.3 DIAGNOSTIC STRATEGY

This section describes the Large Systems Diagnostic project for the KC1Ø CPU cluster and peripheral devices. The CPU cluster consists of the KC1Ø CPU, its memory, an optional arithmetic accelerator, a console processor, and all specified peripheral bus adapters. The diagnostic programs for peripheral devices and intelligent subsystems included in this plan are those that already exist and will require modifications to operate with the KC1Ø CPU.

The major effort will be the development of the diagnostic programs needed for the cluster. These will be the repair level diagnostic programs which will be used by manufacturing and field service in the building and maintenance of the system.

In addition to the diagnostic effort, Diagnostic Engineering will provide the console software system [KCCON] and the run time support program [KCRSP]. This provides support for the operating system and an on-line system for isolating most solid and intermittent faults to the defective module.

In order to permit quick verification of repair and to allow rapid testing of modules in manufacturing, we shall supply hardware diagnostic programs for each CPU module as a subset of the programs which verify KC1Ø operation.

Current projections indicate that the required manpower to provide the items listed in this document will grow from its current level of eight engineers (8) to ten (1Ø) during Q2 of FY 81. This manpower level shall be then required for the life of the project.

We shall support hardware and software engineering with console programs, debugging tools, limited hardware operability tests and updated PDP-1Ø functional tests for breadboard power on. Hardware diagnostic test development will follow this activity with preliminary release of these programs to support shipment of the field test systems. The diagnostic programs that will be required to support the I/O devices will be available to meet FCS. These device diagnostic programs will be the same diagnostic programs that are used for KL1Ø's with modifications made for JUPITER.

Diagnostics will be providing six categories of software. These are the (1) Console Operating System Software, (2) Hardware diagnostics, (3) Functional Diagnostics, (4) Exercisers, (5) User-Mode Diagnostics, (6) Utilities.

5.4 CUSTOMER SERVICE STRATEGY

5.4.1 GENERAL

Maintainability Engineering is responsible for insuring the qualities of Reliability and Maintainability are optimized in the design (Hardware and Software) of a product. MEG is also responsible for insuring that users of a product take full advantage of RAMP features and procedures. It is the goal of the JUPITER MEG group to devote its efforts toward delivery of a System by DIGITAL to the Market which is the most successful in reference to Availability.

MEG because of the collective experience of its members has the knowledge of the factors which effect the product in its end environment. With this knowledge, the knowledge of all contributors and our mutual cooperation we will achieve the Products Goals.

5.4.2 MAINTENANCE PLAN

This CPU and supporting Software are designed to affect System repair by Modular replacement. The module will be either a Circuit Board or a Power Supply or Cable. The Diagnosis System which includes the Hardware Error Detection, the Software Error Handling, and the Console Hardware for Error Reporting will result in a Field Replaceable Unit (FRU) being called out for solid problems. When an error threshold is exceeded on intermittents, a FRU will be called out. When there is a class of error that can not be associated with one FRU, two FRU's will be called out.

This CPU will utilize an Instruction Retry Technique. This means that upon detection of an Error the Program Counter will be backed up depending on the Retry Algorithm, and the instruction will be retried (not all classes of instruction can be retried). This is hoped to significantly reduce the number of System Crashes. When a Instruction is recoverable (no Problem the second or third execution) the fact that this event occurred is stored in the error file. In addition a background System will monitor error occurrences. This System will inform the Customer once a Error occurrence threshold is exceeded. The Console Hardware and Software will determine the severity of Malfunction to determine if System Crash is the only alternative.

5.4.2.1 PROPOSED CALL HANDLING SEQUENCE - Field Service is currently faced with a major problems in the area of labor which does not show signs of improving in the future. The problem is that of the unavailablity of a skilled labor force. With this in mind the Service Organization is proposing a change to traditional Call Handling and Maintenance Philosophy. The change is based around a SPECIALIST program due to the long training cycle for the current BRANCH SUPPORT person (someone who knows something about all devices).

CUSTOMER CALLS (800 Number) Gives Console information

SERVICE DISPATCHER

attempts to dispatch SPECIALIST based on Console inform.

if DISPATCHER can not identify the problem to a device he calls the ASSISTANCE CENTER

ASSISTANCE CENTER

tries to determine problem to a device via investigation over comm. line. Informs DISPATCHER to send SPECIALIST or BRANCH SUPPORT engineer.

SPECIALIST

attempts to resolve problem within guidelines (1-2 Hrs) using fixed techniques. If unsuccessful he notifies ASSISTANCE CENTER. If their assistance does not resolve the problem in (1 Hr.) a call is placed for a BRANCH SUPPORT person.

BRANCH SUPPORT

Will perform some of the following procedures to isolate the problem. The order of the procedures will change dependent on symptom, level of persons involved, and type of problem.

Run Diagnostics
Utilize Software resources
Margin Volt./Timing
Isolation Soft/Hard. Fault
Review problem history
Review Spear/Sys. Err. Files
Get Additional Assistance
Follow Crisis Management Procedures
Customer close procedure.

5.4.3 AVAILABILITY

The JUPITER System has an availability profile predicted to be at least 99%. This means that for a given six month period the inherent failure rate yields an average contribution to system unavailablity of no more than 1% when measured over that six month period.

Since there are other detractions from the availability such as misuse, abuse, etc. We shall no longer refer to an availability of 99%, rather a system unavailability of no greater than 1%. This 1% of unavailability in real monthly hours is approximately 7.2 hours. The derivation is Per Cent Unavailability is equal to 1- Mean Time Between System Failure divided by Mean Time Between System Failure plus Period of Unavailability.

5.4.4 SYSTEM PRODUCT RAMP REQUIREMENTS

1. Monthly System Unavailability of no more than 1%

2. Mean Time To Repair (nose to console time) of 3 Hrs.

3. Mean Down Time of 5 Hrs.

4. Diagnostics will provide at least:

    1. Diagnostic resolution for the greater than 99% of the faults that are detectable by the hardware diagnostics will be to two or less boards 98.9% of the time.

    2. Diagnostic resolution for the greater than 99% of the faults that are detectable by the hardware diagnostics will be to a single board 92% of the time.

    3. Intermittent faults will be handled by reporting the error symptoms and continuing the system. The hardware error checkers will be used as much as possible to detect and isolate this class of errors.

    4. For 95% of the remaining 20% of the solid faults that the hardware does not detect the diagnostics will isolate to two or less modules.

    5. For 80% of the remaining 20% of the solid faults that the hardware does not detect the diagnostics will isolate to the failing module.

```
                       DIAGNOSTIC FAULT COVERAGE MAP       % Coverage
============= ****************************************  ==========
< 1%          *                                      *
Errors        *           (Undetectable by)          *
Non-Diag      *         (Hardware Diagnostics)       *
Detect        *                                      *
------------- ****************************************  ----
              *                                      *  > 99%
         1.1% *  Unisolatable by diagnostics         *
              *                                      *
> 99%    ---- ****************************************  ----
Errors        *                                      *   98.9%
Diag     6.9% *  Diagnostic Isolated to Two Boards   *
Detect        *                                      *
         ---- ****************************************  ----
              *                                      *   92%
          92% *  Diagnostic Isolated to One Board    *
              *                                      *
============= ****************************************  ----
        100%
```

NOTE

THE IMPLICIT ASSUMPTION IS THAT 80% OF THE CPU CLUSTER IS ERROR CHECKED BY HARDWARE.

5. One occasion of unscheduled "System" unavailability which results in remedial maintenance per month.

APPENDIX A

RELATED DOCUMENTS

A.1 PHASE 1 DOCUMENTS

1. DOCUMENT: Engineering Functional Specification
   EDITOR: Don Lewine
   ADDITIONAL COPIES AVAILABLE FROM: Laura Chu MR1-2/E85 X6355

2. DOCUMENT: Project Plan For TOPS-2Ø Support of the 2Ø8Ø
   EDITOR: Sumner Blount
   ADDITIONAL COPIES AVAILABLE FROM: Sumner Blount MR1-2/E37 X6328

3. DOCUMENT: JUPITER Diagnostic Project Plan
   EDITOR: Jack Rosen
   ADDITIONAL COPIES AVAILABLE FROM: Jack Rosen MR1-2/E68 X6191

4. DOCUMENT: JUPITER Maintainability Engineering Project Plan
   EDITOR: Max Weinfuss
   ADDITIONAL COPIES AVAILABLE FROM: Max Weinfuss MR1-1/S35 X6875

5. DOCUMENT: Project JUPITER Manufacturing Plan
   EDITOR: Bill Martel
   ADDITIONAL COPIES AVAILABLE FROM: Bill Martel MR1-3/P74 X6467

6. DOCUMENT: Business Plan
   EDITOR: Rich Fiorentino
   ADDITIONAL COPIES AVAILABLE FROM: Nancy Parnell MR1-2/E78 X4251

7. DOCUMENT: Qualification Plan
   EDITOR: Tran Phando
   ADDITIONAL COPIES AVAILABLE FROM: Ron Setera MR1-2/E18 X6213

8. DOCUMENT: Hardware Development Plan
   EDITOR: Nat Kerllenevich
   ADDITIONAL COPIES AVAILABLE FROM: Laura Chu MR1-2/E85 X6355

A.2 OTHER DOCUMENTS

A.2.1 DIAGNOSTICS

1. DOCUMENT: Massbus Adapter Diagnostic Functional Spec.
   EDITOR: Mike Augeri

2. DOCUMENT: KC1Ø MBOX Diagnostic Functional Spec.
   EDITOR: David Tongel

3. DOCUMENT: KC1Ø Console Subsystem Software (KCCON) Function Spec.
   EDITOR: John Kirchoff

4. DOCUMENT: KC1Ø Console Diagnostic Support Package (KCDSP) Functional Spec.
   EDITOR: Skip Gaede

5. DOCUMENT: KC1Ø Console Trace/Debug Package (KCTRAK) Functional Spec.
   EDITOR: Jean Basmaji/Jim Jones

6. DOCUMENT: Console-based KC1Ø I/O Port Diagnostic Functional Spec.
   EDITOR: Rich Muratori

A.2.2 PRODUCT MANAGEMENT

1. DOCUMENT: 2Ø8Ø Product Requirements
   EDITOR: Richard Fiorentino

2. DOCUMENT: LSG Communication Hardware Product Requirements
   EDITOR: Sharon Passon

A.2.3 TECHNOLOGY

1. DOCUMENT: 2Ø8Ø Technology PERT
   EDITOR: Warren Peluso

APPENDIX B

PRODUCT REQUIREMENTS -- ENGINEERING RESPONSE

B.1 GENERAL PRODUCT GOALS

1. TIME TO MARKET: Target date is July 1982. 90% commitment date is July 1983.

2. TRANSFER COST: 46K for CPU and 512K memory.

3. PERFORMANCE: Commitment to 5.00 X KL10 using standard logic mix. Performance may be significantly higher.

4. System will have significantly higher availability and significantly lower cost to service than a comparable 2060 system.

5. The JUPITER system will be able to support multisystem configurations using the CI bus. All CI nodes must run TOPS-20.

6. DECNET phase III with terminal concentration will be fully supported on the JUPITER at FRS.

7. IBM communications is not funded.

B.2 IO REQUIREMENTS

1. MASSBUS - being done.

2. UNIBUS - being done.

3. IBM BLOCK MUX - will be available at FRS + 1 Year

4. CI - being done.

5. NI - will be available at FRS + 1 Year if it is a corporate product at that time.

B.3 RAMP

All goals should be met. Instead of measuring monthly mainframe availability, we intend to measure monthly system unavailability of 1% or less. This amounts to 7.2 hours per month of unavailability and includes all scheduled and unscheduled corrective and preventive maintenance.

B.4 MAINTAINABILITY

See section 5.4.4

B.5 CPU REQUIREMENTS

All requirements will be met.

B.6 I/O AND MEMORY CAPACITY REQUIREMENTS

All requirements will be met.

B.7 BACK-UP/RESTORE REQUIREMENTS

All requirements will be met.

B.8 COMMUNICATIONS REQUIREMENTS

All requirements will be met.

B.9 SOFTWARE REQUIREMENTS

Will be addressed separately.

B.1Ø PACKAGING REQUIREMENTS

All requirements will be met. We will use dual RLØ2 disk drives instead of RXØ2 disk described in the requirements.

B.11 ENVIRONMENTAL REQUIREMENTS

All requirements will be met. The cost to upgrade office space to computer room environment is $11.5Ø per square foot. Thus a $25K fit-up requirement translates into over 2ØØØ square feet of machine room. That is large enough to house a large JUPITER system.

B.12 INSTALLATION REQUIREMENTS

All requirements will be met.

B.13 SERVICE COST REQUIREMENTS

All requirements will be met.

APPENDIX C

PROGRAM ORGANIZATION

APPENDIX D

GLOSSARY

ACS Accumulators -- General Purpose registers in PDP-1Ø architecture

ADDR Address

ALU Arithmetic and Logical Unit -- Type of IC used to do computation

APA Arithmetic Processing Accelerator -- Hot box used to speed up floating point, integer and byte instructions.

BACKPLANE A piece co circuit board into which modules are inserted. Contains pins which are wirewrapped or connected by etch.

BIS Business Instruction Set. Used by COBOL on the KL1Ø. See NCIS.

BISYNC Name of protocol used by IBM.

BMC Basic Monthly Charge -- Charge for minimum level of field service coverage. All other field service charges are derived by multiplying BMC by a premium rate.

BPI Bits Per Inch

BREADBOARD A machine built by engineering to test out the design concepts used in the JUPITER system. May have differences from the final machine.

CAD Computer Aided Design.

CALDEC A PDP-15 based printed circuit board layout system.

CCW Channel Command Word

CI Computer Interconnect -- New name for the ICCS bus. This is the high speed bus used to build multi-computer networks.

COMM Communications.

CROSSTALK Transfer of a signal from where is is supposed to be to some other wire by electromagnetic radiation (Radio).

CSL Console

CSSE Customer Services Software Engineering?

CTY Console Terminal. The terminal directly connected to the JUPITER CPU cluster.

DATAPATH That part of the CPU used to transfer or operate on data. As opposed to control logic.

DDCMP Protocol used to control physical links in DECNET.

DECNET Name of Digital Equipment's network product family.

DIAG Diagnostic

DMA Direct Memory Access -- The ability of an IO device to read and write memory without interrupting the CPU.

DMT Design Maturity Testing. This process runs several machines and looks at failures. From this test it is possible to predict field operation.

DOUBLEWORD Two words; 72 Bits

DP Double Precision

DVT Design Verification Testing -- The process of verifying that the JUPITER meets its design specifications.

EACALC Effective Address Calculation. The process of evaluating an instruction's address.

EBOX Execution Box -- The place where the instructions are performed.

ECC Error Correcting Code -- A method of storing data and checkbits such that if any bit changes one can detect exactly which bit changed and correct it.

ECL Emitter Coupled Logic -- A high-speed logic family used in the JUPITER CPU.

ECO Engineering Change Order -- Bug fix.

EMI Electro-magnetic interference

FAILSOFT The ability to tolerate a failure without stopping the entire system.

FCC Federal Communications Commission. Administers the communications act of 1934 as amended. Part of the Congress of the United States its rules have the effect of laws.

FCS First Customer Ship -- The first time a customer (Paying or not) gets a product.

FPA Floating Point Accelerator -- see APA

FRS First Revenue Ship -- The first shipment from FA & T to a customer who is expected to pay for a JUPITER.

FRU Field Replaceable Unit -- The smallest thing a field service person is allowed to replace.

FY Fiscal Year. Runs from July to June.

ICCS Inter Computer Communications Switch. See CI.

IBOX Instruction Box -- Does instruction fetch, effective address calculation and operand fetch for most instructions.

INTERCOMPUTER From one major source of computation to another.

IOBOX Input output BOX -- Used to interface the high speed ECL memory system to the TTL PORTs.

IOBUS A collection of wires used to do input and output.

IPA ICCS Port Adapter -- Path to the CI.

JRST A PDP-1Ø instruction used as a unconditional jump.

KCCON The main subroutine package used by all console software. Runs in the console.

KCRSP KC1Ø Runtime Support Package. Runs in the console PDP-11 during normal operation.

KILOBAUD 1ØØØ bits per second.

KLINIK Name of remote diagnosis product used on the KL1Ø.

LCS Loosely Coupled Systems -- Multiprocessor system using the CI bus.

LRU Least Recently Used

LSG Large Systems Group

MASSBUS A DEC bus used to connect disk and tapes to CPUs.

MBA MASSBUS Adapter

MBOX Memory Box -- That part of a JUPITER that holds the Cache, Paging and main memory.

MEG Maintainability Engineering Group.

MICROCYCLE The time required to execute one microinstruction.

MICROINSTRUCTION A collection of control bits used to manipulate the JUPITER datapath.

MICROORDER A single opcode in one of the opcode fields of a microinstruction.

MIPS Millions of Instructions Per Second

MLP Maynard List Price -- This is the amount in the price book. The customer pays this amount plus shipping and installation less discount.

MOS Metal-Oxide Silicon. A type of integrated circuit.

MSI Medium Scale Integration. Refers to the amount of logic on a single chip.

MTBF Mean Time Between Failure

MTTR Mean Time To Repair

MULTIWIRE Type of circuit module which uses magnet wire in epoxy instead of etched copper.

NC Numericly Controlled. A type of automatic manufacturing machine.

NCIS New Commercial Instruction Set -- A group of instructions designed to help COBOL performance.

PARTITIONING The process of breaking the design up into modules.

PBOX The IBOX + EBOX

PCM Plug Compatible Manufacture. Someone who makes disks and tapes that attach to IBM computers.

PIPELINED Machine organization where several instructions are in execution at the same time.

PLI Port-Link-Interface.

PMT Process Maturity Test. A test to make sure that machine built by manufacturing in volume are as reliable as those tested during DMT.

POWERON Applying power for the first time.

PROTO Prototype.

QV Quick Verify. Used by Manufacturing to see if a module works.

RAMS Random Access MemorieS -- High speed read/write memories.

RFI Radio Frequency Interference.

STANDALONE Used by only one person at a time.

SSI Small Scale Integration. A type of IC with only a few gates on it.

TBD To Be Defined -- We don't know yet.

TTL Transistor-Transistor Logic. A logic family which is slower, cheaper and uses less power than ECL.

UBA UNIBUS ADAPTER

UETP User Environment Test Package -- some test programs that run under the monitor in normal timesharing.

UNIBUS A PDP-11 Bus

VAX A 32-bit processor family manufactured by DEC.

VMA Virtual Memory Address -- Also the name of register which holds the address.

WIRERULES The set of restrictions that make sure one signal does not interfere with another.

WRITEBACK A type of cache where written data is only stored in main memory when space is needed in the cache.