DEC 06 1982

Ron Setera

CONVERSION STEPS

1. GENERATE A LOGIC EQUIVALENCE LIBRARY FOR MSI-TO-MCA CELLS.

2. CONVERT MSI MODULES DATABASE TO MCA CELLS "INTERMEDIATE" DATABASE.

3. CONVERT MSI MODULES DATABASE TO RAMS AND 10KH" INTERMEDIATE" DATABASE.

4. CONVERT MCA CELLS AND RAMS AND 10KH LOGIC ON TO "L" SERIES BOARD BOUNDARIES.

5. RUN PLACEMENT TOOL TO DO TOTAL MODULE PLACEMENT FROM BACKPANEL IN, TO FORCE OPTIMIZING WITHIN MODULE BOUNDARIES GENERATING NEW MODEL "C" DATABASES.

6. RUN COMPARISON BETWEEN OLD MODEL "B" DATABASE AND NEW MODEL "C" DATABASE.

7. RUN SAGE ON NEW DESIGN.

8. SUBMIT NEW MCA'S AND "L" SERIES MODULES.

Gordon Bell
from
Ron Setera

Gordon Bell
from
Ron Setera

Ron Satara

CONVERSION STEPS

1. GENERATE A LOGIC EQUIVALENCE LIBRARY FOR MSI-TO-MCA CELLS.

2. CONVERT MSI MODULES DATABASE TO MCA CELLS "INTERMEDIATE" DATABASE.

3. CONVERT MSI MODULES DATABASE TO RAMS AND 10KH" INTERMEDIATE" DATABASE.

4. CONVERT MCA CELLS AND RAMS AND 10KH LOGIC ON TO "L" SERIES BOARD BOUNDARIES.

5. RUN PLACEMENT TOOL TO DO TOTAL MODULE PLACEMENT FROM BACKPANEL IN, TO FORCE OPTIMIZING WITHIN MODULE BOUNDARIES GENERATING NEW MODEL "C" DATABASES.

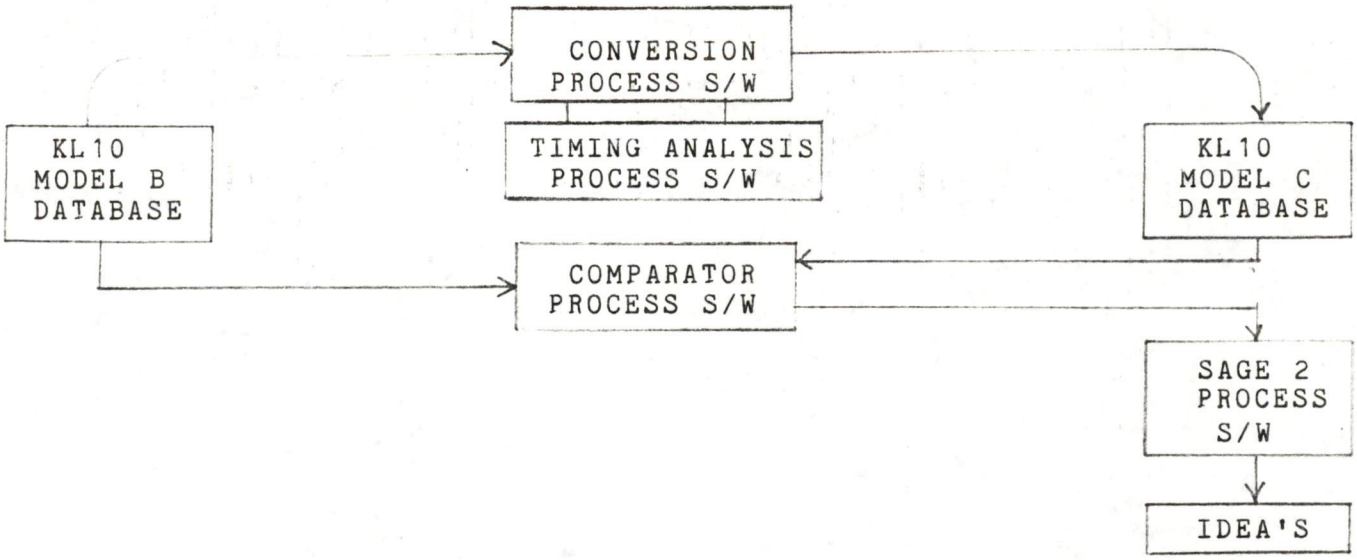
6. RUN COMPARISON BETWEEN OLD MODEL "B" DATABASE AND NEW MODEL "C" DATABASE.

7. RUN SAGE ON NEW DESIGN.

8. SUBMIT NEW MCA'S AND "L" SERIES MODULES.

Ron Setera

DETAILS OF CONVERSION PROCESS



- KL10 MODEL B DATABASE - USING THE "NET LISTS" THE CONVERSION S/W WILL RECONSTRUCT THE DESIGN IN MCA'S FROM PREVIOUSLY IMPLEMENTED IN MSI ECL.

- CONVERSION PROCESS S/W - THIS NEW S/W WILL ACCESS THE EXISTING MSI 10K ECL SUDS DATABASE. USING THE "NET LISTS" TO RECONSTRUCT THE DESIGN INTO A PRESELECTED GROUP OF MCA TYPES LIBRARY.

- TIMING ANALYSIS - THIS S/W WILL ACCESS THE NOW "INTERMEDIATE NET LISTS" DATABASE AND WILL PROVIDE TIMING INFORMATION FOR THAT MODULE LEVEL DESIGN.

- PARTITIONING S/W - THE NEW LOGIC PARTITIONING S/W WILL EQUIP DESIGNERS WITH THE TOOLS NECESSARY TO QUICKLY REORGANIZE THE LOGIC NETS INTO NEW LOGIC CHUNKS. THAT IN TURN WILL BE GROUPED INTO MODULE LEVEL DESIGNS, STILL MAINTAINING AND PRESERVING THE ORIGINAL LOGIC NETWORK OR FUNCTION.

- KL10 MODEL C DATABASE - THIS KL10 MODEL C DATABASE WILL CONSIST OF ALL NECESSARY FILES, I.E., DRAWER, TLE, WLS, FILES, ETC.

- COMPARATOR PROCESS S/W - THIS NEW S/W PROCESS COMPARES THE TWO DESIGN DATABASES. THE PURPOSE IS TO ENSURE THAT THE LOGIC NETS/FUNCTIONS ARE EQUAL.

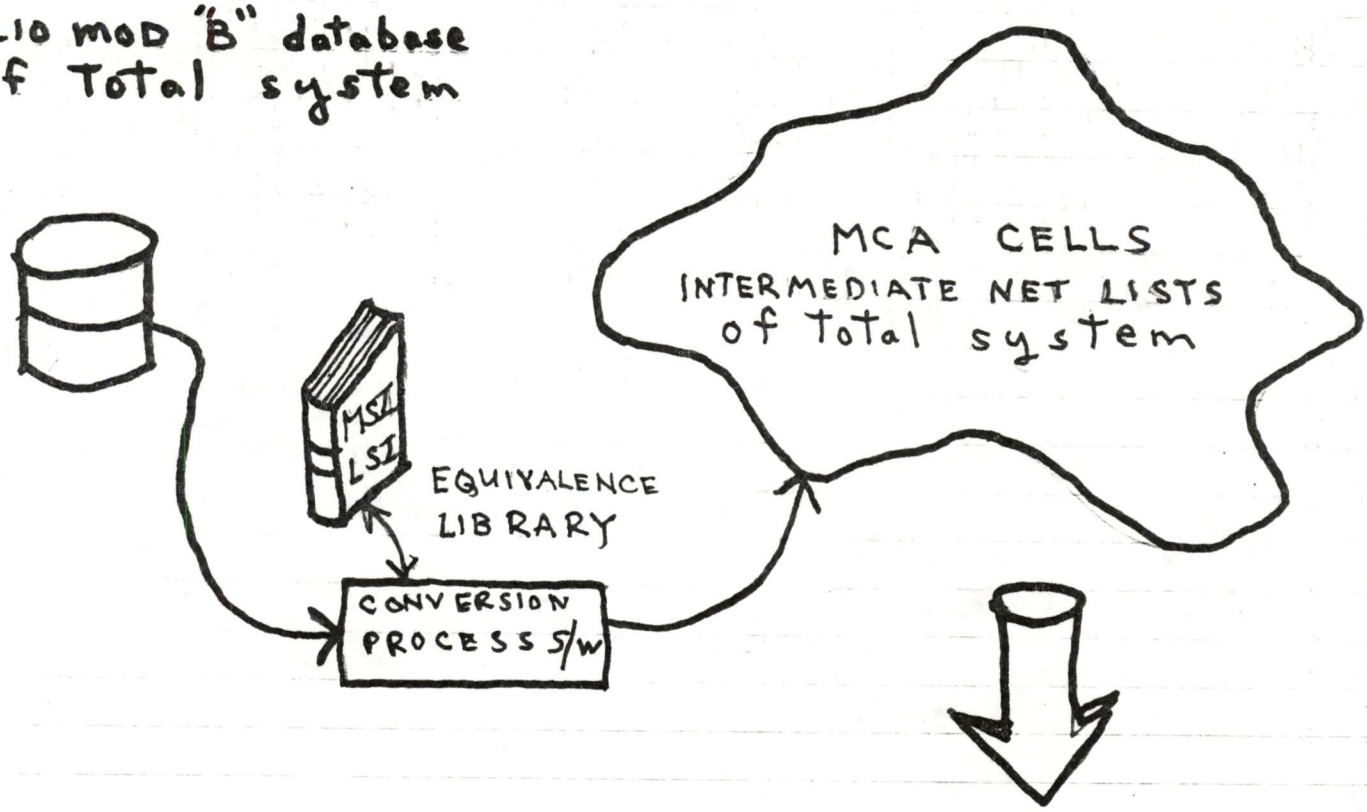
- SAGE 2 - IN THIS DESIGN PROCESS FOR THE KL10 MODEL C PRODUCT, THE AMOUNT OF FUNCTIONAL/ARCHITECTURAL CHANGE IS SMALL. IN THOSE AREAS OF CHANGE SAGE 2 SIMULATION MUST BE DONE TO GUARANTEE ERROR FREE DESIGN.

Database Generation

Ron Setera

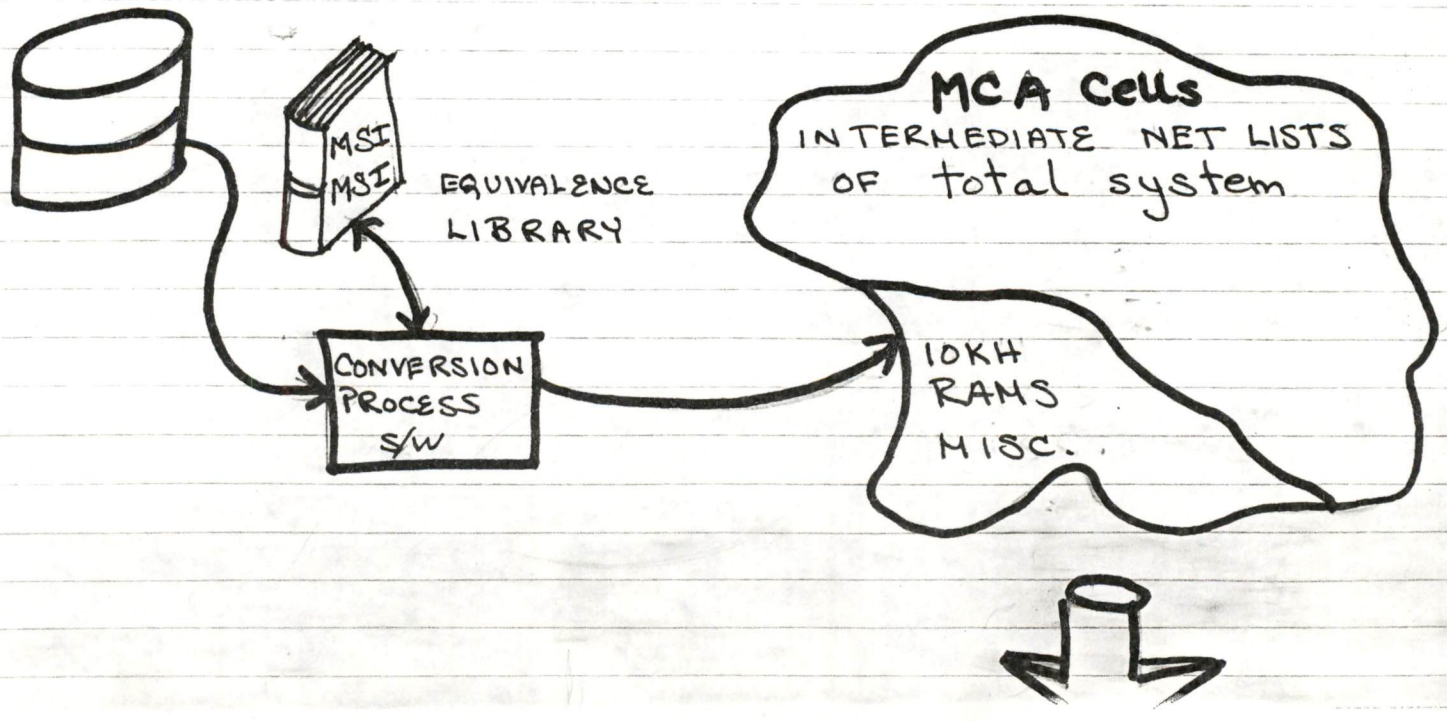
STep 1: (AUTOMATIC 100%)

KLIO MOD "B" database of Total system



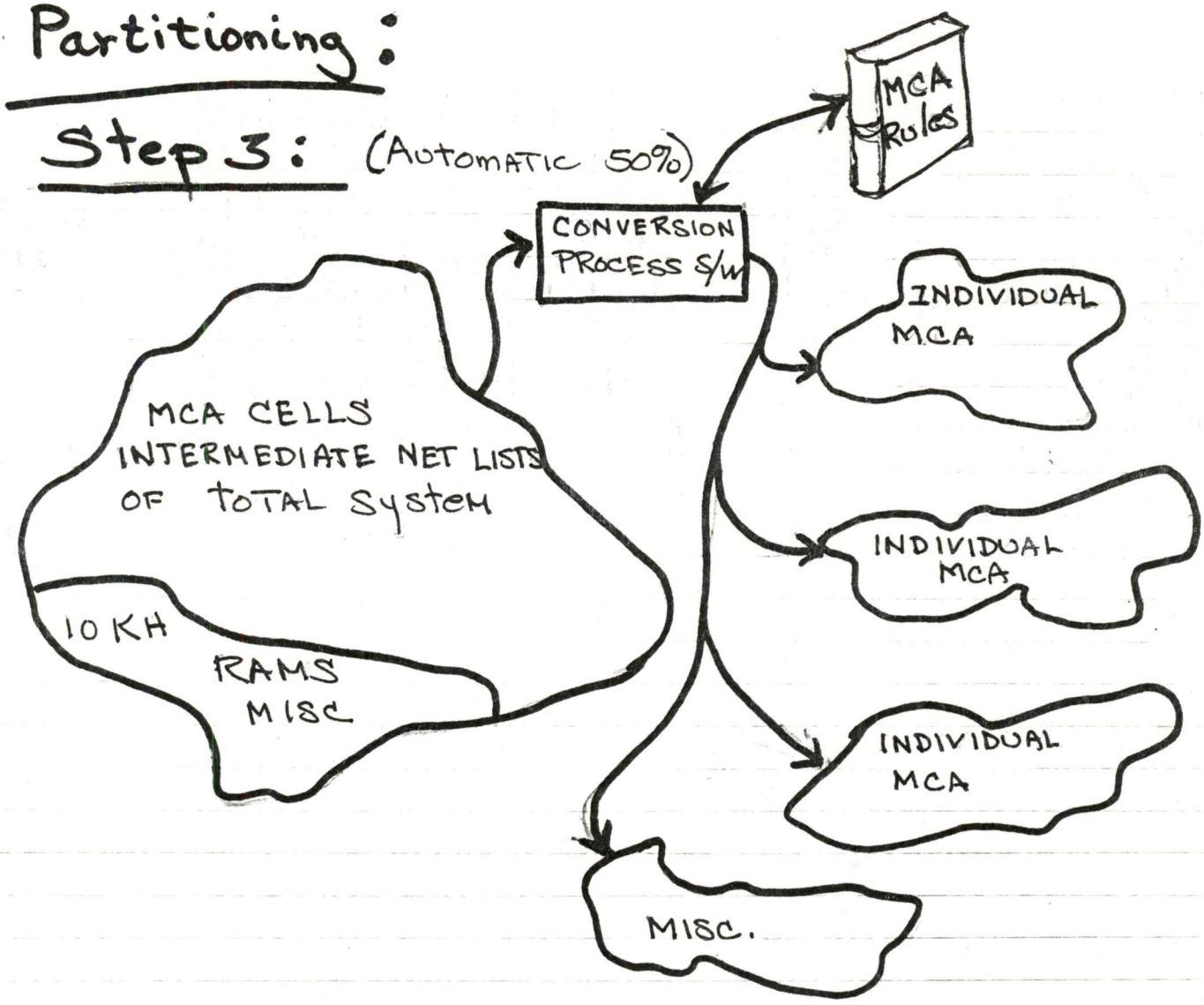
Step 2: (AUTOMATIC 80%)

KLIO MOD "B" database

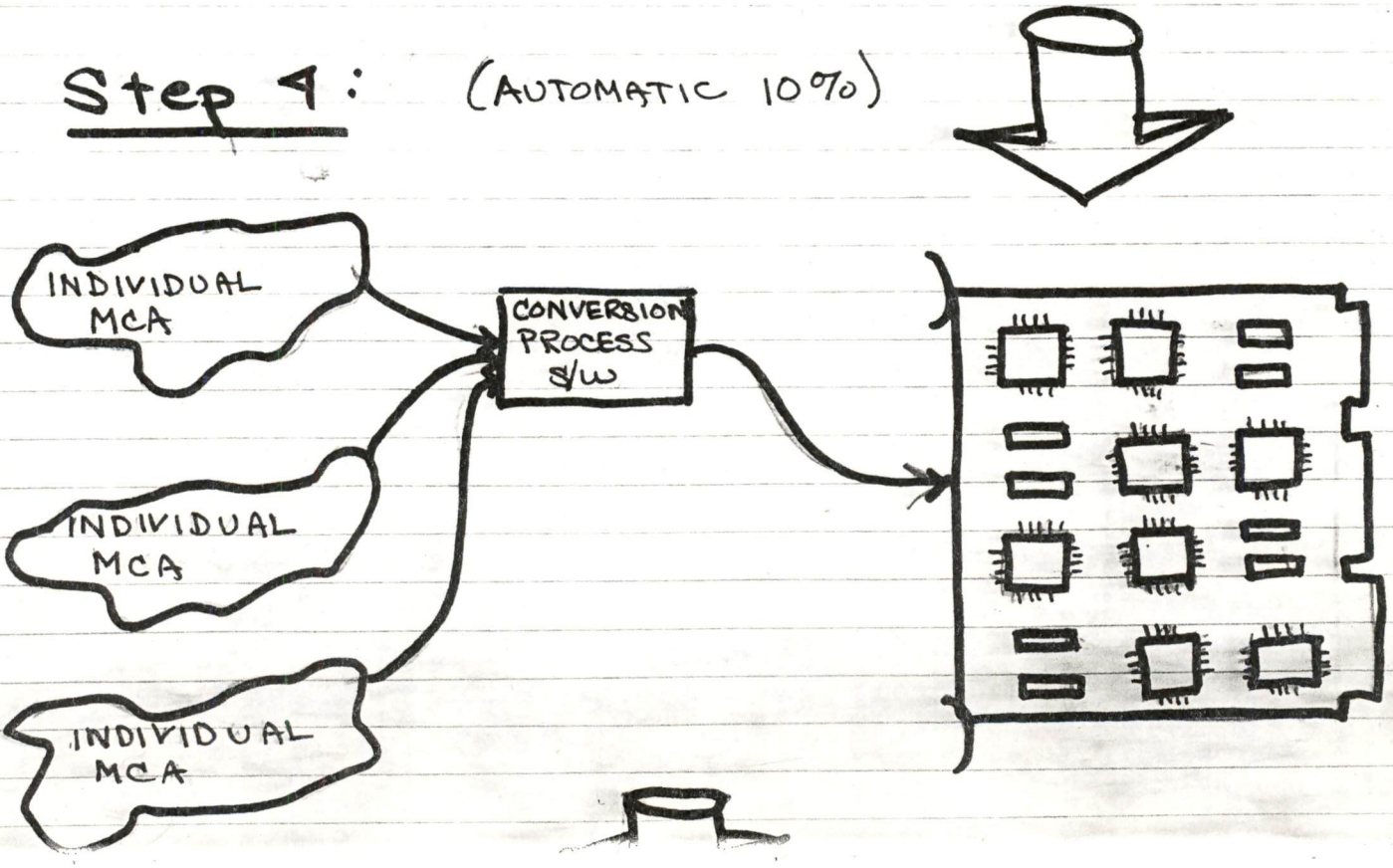


Partitioning:

Step 3: (Automatic 50%)

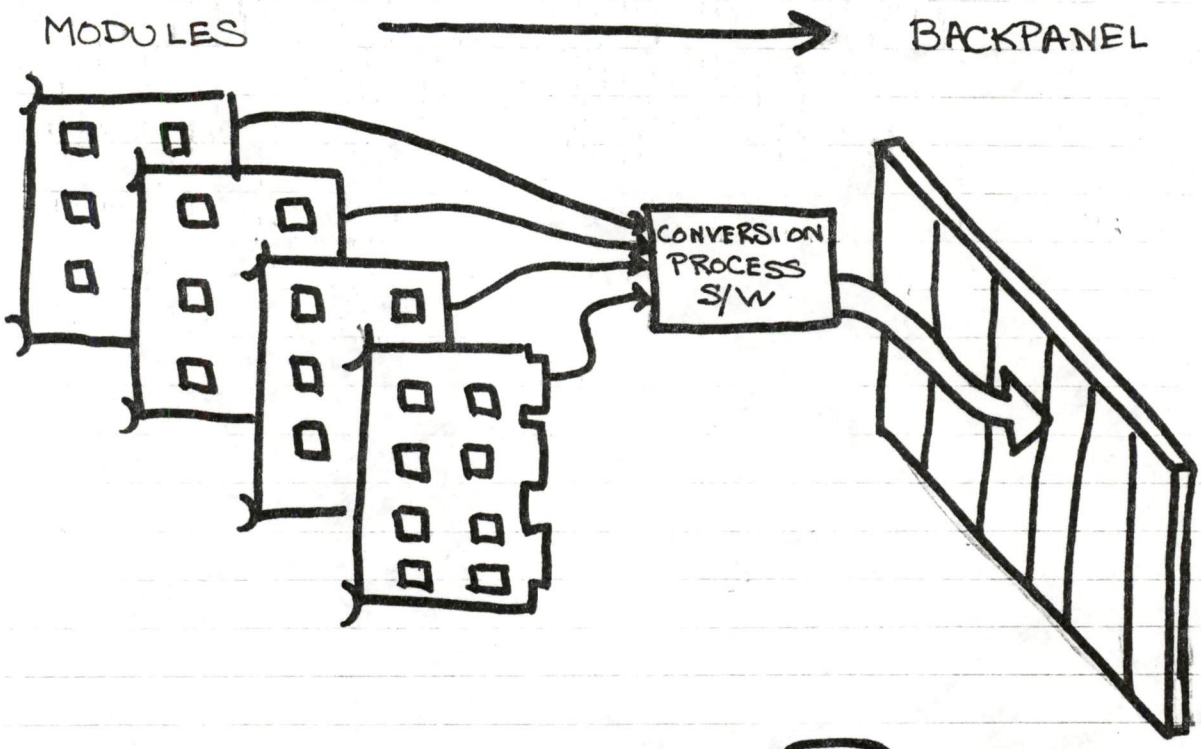


Step 4: (Automatic 10%)

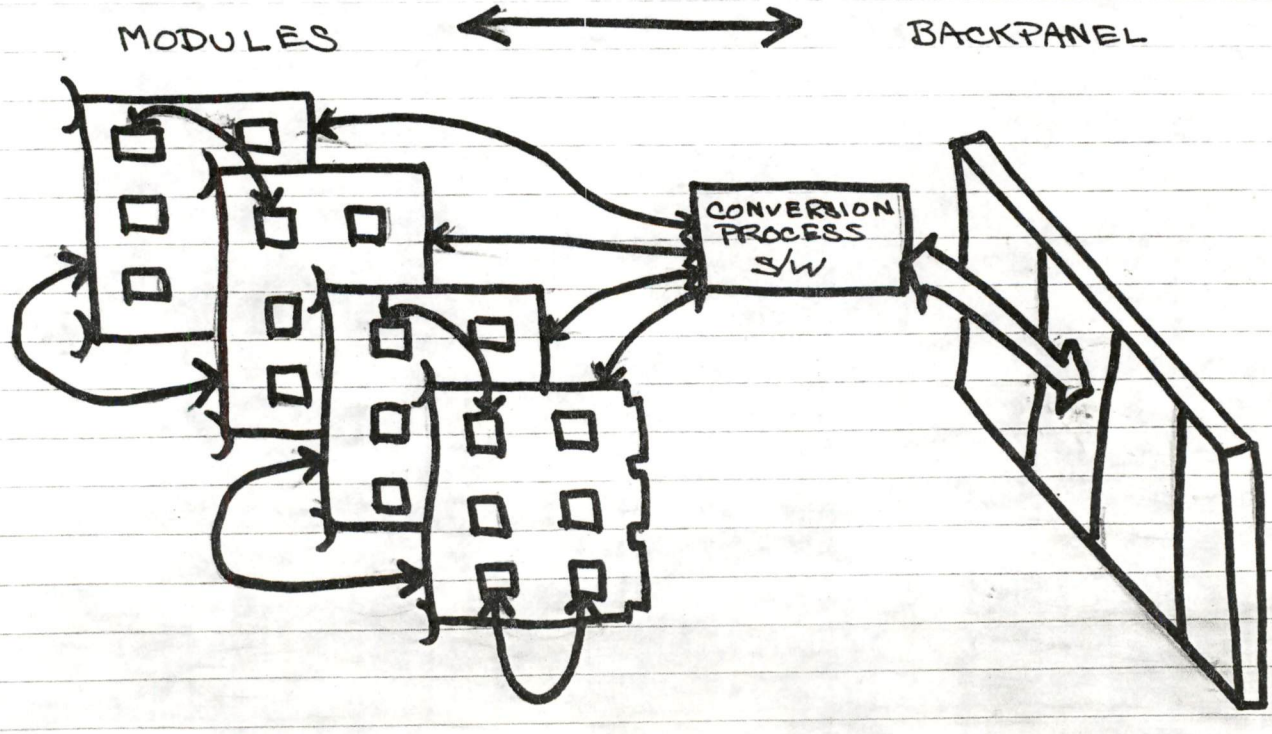


Partitioning - Optimizing:

Step 5: (AUTOMATIC 90%)



Step 6: (AUTOMATIC 90%)

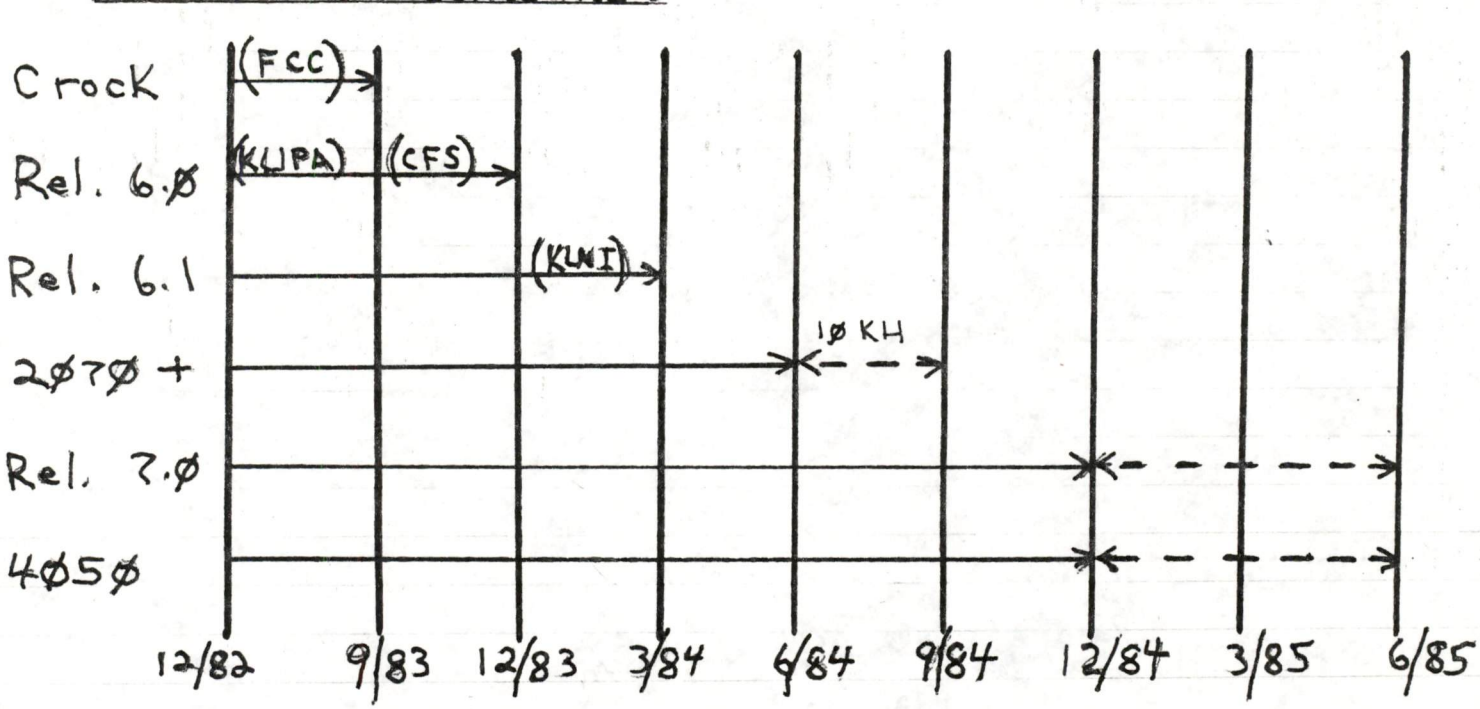


Tops ∅∅ KL/KC10

1976 - KL10 "B" - (SINGLE)	1.7 MIPS
1983 - KL10 "B" - CFS - KL10 "B"	3.4 MIPS
1984 - KL10 "C" - (SINGLE)	2.48 MIPS
1984 - KL10 "C" - CFS - KL10 "C"	4.96 MIPS
1985 - KC10 "B" - (SINGLE)	6.8 MIPS
1985 - KC10 "B" - CFS - KC10 "B"	13.6 MIPS

Ron Setera

Overview of schedules:



NOTE: The assumptions here is that the 2070+ will be done with 10KH not MCA'S. The estimate for a MCA'ed KL10 is add one more year.

KL10 MSI → MCA's

KL10 CPU = 3600 IC's

cache Rams = 360 IC's

3960 = 4K IC's

MCA → MSI Ratio is 30:1

8 "L" series modules assuming
15 MCA's per + rams + 10KH

2 "L" series modules for the caches

2 slots for cables etc. etc.

Total of Twelve (12) slot backpanel

15000 gates in KL10

* * * * * * * * *
* d i g i t a l *
* * * * * * * * *

INTEROFFICE MEMORANDUM

TO: List

DATE: 12 November 82
FROM: Ron Setera
DEPT: CPS/Peripherals
EXT: 231-6213
LOC/MAIL STOP: MRO1-2/E18

SUBJ: KL10E Product Extension Efforts Proposal
(Create a Model "C")

In the interest of extending the KL10E product life, the following tasks are considered essential to improve the KL10E price/performance picture, its manufacturibility and the customer's lifecycle costs. The intent of this proposal is not to do any significant system redesign, but to do some system fine tuning in areas where we would get the biggest bang for our buck.

1. Due all 2070 proposal tasks as defined in the proposal by that name. -20K.
(See Attachment 1)

2. Speed-up the CPU clock reducing the clock period from 67 nsec. to 47 nsec. This 30% CPU improvement will be achievable through the use of ECL 10KH technology in much of the CPU to solve critical timing areas. The following are some interesting facts about ECL 10KH.

MECL 10KH vs. MECL 10K

- MECL 10KH is voltage compensated logic which allows guaranteed DC and switching parameters over a +/- 5% power supply range.
- Noise margins for MECL 10KH are 75% better than MECL 10K.
- MECL 10KH is compatible with MECL 10K to allow system designers to enhance existing systems by increasing their product's speed in critical timing areas.
- Many MECL 10KH devices are pinout function duplications of MECL 10K.
- MECL 10KH features 100% improvement in propagation delay and clock speed while maintaining power supply currents equal to MECL 10K devices.

o Typical family characteristics:

	10K	10KH
Propagation delay (ns)	2.0	1.0
Power (mw)	25	25
Power speed product (pj)	50	25
Rise/fall times (ns) (20-80%)	2.0	1.5
Temperature range (C degrees)	-30/+85	0/+75
Voltage regulated	no	yes
Technology	junction isolated	oxide isolated

3. Increasing the micro-code-storage size from 2K to 3K words will allow improvement in three areas:

   a. Improving instruction performance
   b. Adding new instruction
   c. Micro-code ECO space

   Example:

   o Move conversion info for OWGBP into scratchpad. Increases IBP, DPB, IDPB, ADJBP, LDB, ILDB and all string instruction speeds when using OWGBP's. For simple byte instructions this is about 15%.

   o Rewrite all Move String and Compare String instructions by caching data and not re-evaluating byte pointers. This reduces the instruction execution time by 50%. This would cause a 4% increase in user throughput in a heavy COBOL environment.

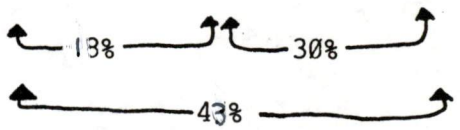
   o Rewrite CNVDBT to cache data and byte pointers. This saves about 20%. This would cause a 15% increase in user throughput in a heavy COBOL environment.

   o Rewrite CNVBDT to not use general divide routine and to cache data and byte pointers. This reduces the instruction execution time by 50%. This would cause a 15% increase in user throughput in a heavy COBOL environment.

4. Use a larger RAM for the AC Blocks. This will still allow 8 sets of AC blocks but also give working space/scotch-pad space for microcode to speed up long instructions that now require multiple memory references.

5. Increasing the cache size from 2K words to 4K words, will improve the cache hit ratio for uni-processors. If in some cases we go from 90% hit to 80% hit, this corresponds to a 30% reduction in processor compute performance. (See Attachment 2) Also, doubling the cache size usually halves the cache miss ratio; therefore, giving us an improvement of around 8%.

KL10 Model	"B" @ 30 MHZ	"C" @ 30 MHZ	"C" @ 39 MHZ
Cache Hits	90%	95%	95%
Memory Type	MF20	MF20	MF20
Clock Freq	30	30	39
Mem Ref Per Instr	1.7	1.7	1.7
EBOX Grind Time/Instr in Clk Ticks	7.0	7.0	7.0
Single CPU W/O I/O MIPS	1.70	1.91	2.48
Single CPU with I/O MIPS	1.60	1.85	2.43
Chan I/O Pages/Sec	600	600	600



6. MF20 memory array modules should be redesigned to use the 64K chip allowing a four to one memory space improvement. This memory enhancement will allow us to increase internal memory from 1.4 megawords to 5.6 megawords and at the same time speed up the memory bus due to shortening from 20 feet to 10 feet.

7. Double the paging RAM size to allow for the separation of user page info and executive page (monitor use) info. This paging overhead had been measured to be approximately 9% of the system overhead on a Rel. 3 monitor. We expect that with the latest monitor now doing extended addressing and languages as well, plus users running larger programs, that this 9% number is now on the low side. Making this improvement will definitely help in this area, possibly reducing it to 5 - 6%. (See Attachment 3)

In summary, it appears that we can comfortably achieve a performance gain of 1.6X the KL10 Model B while reducing the Kernel mainframe cost $20K and overall system level improvement by 1.8X the 2060 system. The simple algorithm below shows our assumption:

Performance Increase for Model C

		10KH	MCA
KL10E		1.00	2.00
CACHE		.08	+
PAGER		.04	+
NO DTE		.20	+
MICROCODE		.10	+
(COBOL)		1.42	2.97
CLOCK SPEED UP	X	1.30	—
		1.85	2.97

MODEL C = 1.85 X MODEL B

VEND PART NO	DEC PART NO	10H PLANNED	SAMPLES	PRODUCTION
10101	19-11398-00	YES	YES	NO
10103	19-11400-00	YES	YES	YES
10104	19-11401-00	YES	YES	YES
10105	19-11402-00	YES	YES	YES
10107	19-11404-00	YES	YES	YES
10109	19-11405-00	YES	YES	YES
10110	19-11406-00	NO USE 10H2/0		
10113	19-11408-00	YES	1Q83	
10115	19-11409-00	YES	2Q83	
10117	19-11410-00	YES	YES	YES
10118	19-11411-00	YES	YES	YES
10121	19-11413-00	YES	YES	YES
10124	19-11414-00	YES	DEC/JAN	
10125	19-11415-00	YES	DEC/JAN	
10131	19-11416-00	YES	YES	YES
10136	19-11500-00	YES	YES	YES
10141	19-11501-00	YES	YES	YES
10144	19-11502-00	NO		
10145	19-11503-00	YES	YES	YES
10147	19-11626-00	NO		
10158	19-11510-00	YES	YES	YES
10160	19-11505-00	YES	YES	YES
10161	19-11417-00	YES	YES	YES
10162	19-11418-00	YES	YES	YES
10164	19-11419-00	YES	YES	YES
10165	19-11530-00	YES	DEC/JAN	
10173	19-11506-00	YES	YES	YES
10174	19-11420-00	YES	YES	YES
10176	19-11507-00	YES	YES	YES
10179	19-11421-00	YES	YES	NOV
10181	19-11511-00	YES	DEC/JAN	
10210	19-11512-00	YES	YES	NOV/DEC
10415A	19-13098-00	NO		

MEMORY DEVICES-

10422-10	256X4	TAA=10NS	AVAILABLE NOW, QUALIFIED IN DEC
10422- 7	256X4	TAA= 7NS	AVAILABLE NOW,
10470-15	4KX1	TAA=15NS	AVAILABLE NOW, QUALIFIED IN DEC
10474-25	1KX4	TAA=25NS	AVAILABLE NOW, QUALIFIED IN DEC
10474-15	1KX4	TAA=15NS	AVAILABLE NOW,

Subject: 2070 Proposal

It is my opinion that the current strategy being pursued for KL/FCC program is wrong. I believe that what we will need will be a 2070. The definition of a 2070 is a new FCC cabineted Model B CPU installed into two double width cabinets instead of three presently. The I/O backpanel would have one unibus interface (DTE), 2 CI ports (KLIPA's), 2 NI ports (KLNI's). This kind of system interconnect would allow the 2070 enough CI/NI raw throughput and mass storage capability that would far surpass the raw m.i.p.s. of the KLIA CPU itself. The DTE would allow an 11/24 front-end to perform all necessary KLINIK and console booting without RSX20F. All terminal and synchronous line communication would take place over the NI through the use of Plutos. The overall performance of this system would porobably realize about a 10% increase due to queued CI and NI protocols.

Attached you will find my drawings of what I see as possible to do within the cabinets, on the drawings that show the use of the 2080 style design.

Transfer cost of this system shown here would be approximately $50K.

Pro's for making some of the suggested changes.

1. Power Supplies - Currently the 760/761 series pass technology supplies are 38% - 42% efficient or about 2/3's of the power consumed by 760/761 goes up directly into heat.

    a. The MF20/MPS supplies are 85% efficient. This efficiency difference will have significant effect on internal heat inside the cab's/reliability of components and customer air conditioning/electrical bill or lifecycle costs.

    b. As an alternative we can use the 2080 MPS regulator design. The power efficiency is the same as above.

2. I/O - I would like to get rid of the old style for the following reasons:

    a. Massbus devices are going away and those that remain will be cost effective for a very short period of time. Due to the FCC compliance reasons, it will be difficult to pass FCC with old devices, not to mention the cost of those efforts.

b. Using NI/CI controllers and devices we should get a free ride on using corporate mainstream developments that will be guaranteed to pass FCC not to mention cost/megabytes and cost per line etc.

3. No more front-end - This has a large affect on product cost. My suggestion is to use an 11/24 for CTY and booting and KLINIK functions. No more communication on F-E. No more RSX20F.

Con's

Higher cost to develop than KL/FCC program now includes.

Attachment 1

2070 MAJOR CHANGES FROM 2060 FCC PRODUCT

1. ELIMINATION OF FRONT-ENDS:

   PRO'S
   - o ELIMINATION OF NEW RELEASE OF RSX20F AND MCB
   - o REDUCE SYSTEM COST BY $12K
   - o REDUCE SYSTEM FOOT PRINT TO SIZE OF 2080
   - o ELIMINATE FURTHER MANUFACTURING OF OUT-DATED PDP11/40's
   - o IMPROVE SYSTEM MAINTAINABILITY AND RELIABILITY

   CON'S
   - o REQUIRES DIAGNOSTIC ENGINEERING DEVELOPMENT ACTIVITY TO MAKE MODIFICATIONS TO KLDCP

2. POWER SUBSYSTEM DEVELOPMENT:

   PRO'S
   - o IMPROVEMENT IN LIFE CYCLE COSTS TO CUSTOMERS DUE TO EFFICIENCY ON NEW POWER SUPPLIES
   - o SOME POWER SUBSYSTEM COST REDUCTION
   - o IMPROVEMENT IN POWER SYSTEM MAINTAINABILITY

   CON'S
   - o REQUIRES POWER SUPPLY ENGINEERING DEVELOPMENT ACTIVITY
   - o RISK TO SCHEDULE DUE TO NEW HARDWARE UNKNOWN

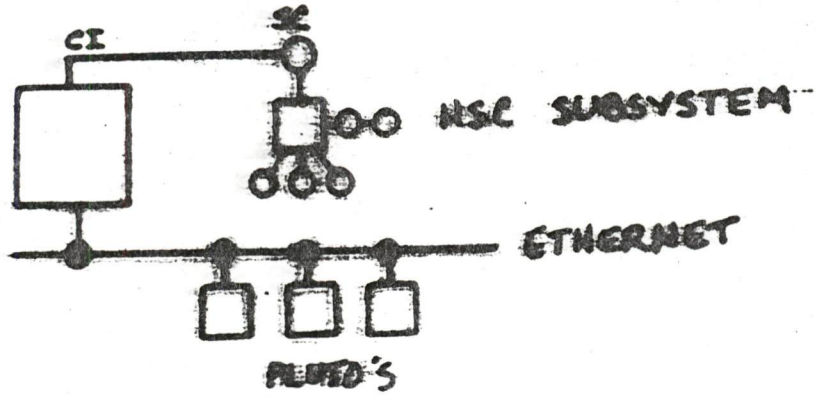
3. SYSTEM CAPACITY ENHANCEMENTS

PRO'S
- o ADDITION OF 2 CI PORTS AND 2 NI PORTS ALLOWING USE OF NEW I/O INTERCONNECTS
- o 2070 WILL USE PLUTO/VOYAGER COMMUNICATION
- o 2070 WILL USE CI CONTROLLERS AND SI DEVICES ALLOWING COST EFFECTIVE ENTRY LEVEL SYSTEMS AND THE ABILITY TO CONFIGURE LARGE SYSTEMS
- o 2070 I/O WILL BETTER PROVIDE GROWTH PATH FOR BRIDGING TO 2080 DUE TO COMMON I/O STRUCTURE, MEDIA TRANSPORTABILITY AND DUAL PORTING
- o 2070 I/O WILL PROVIDE A BALANCED SYSTEM THROUGH-PUT AND PROVIDE ROOM FOR EXPANSION
- o PROVIDES A PRODUCT THAT CAN BE SOLD AS A LOW-END PRODUCT OFFERING TO THE 2080 AND CAN BE STRETCHED TO COVER THE 2080 LIFE CYCLE
- o SYSTEM COST SHOULD BE SIGNIFICANTLY REDUCED DUE TO THE USE OF NEW I/O DEVICES

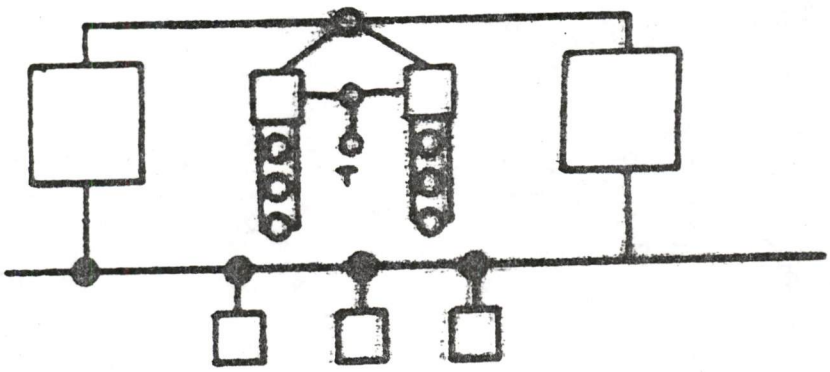
CON'S
- o REQUIRES MICROCODE DEVELOPMENT FOR NI
- o REQUIRES NEW I/O BACKPANEL
- o REQUIRES NEW TOPS-20 DEVELOPMENT
- o REQUIRES MORE DOCUMENTATION EFFORTS
- o MASSBUS DEVICES CANNOT BE ATTACHED DIRECTLY TO 2070

2070 RECOMMENDED CONFIGURATIONS

2071

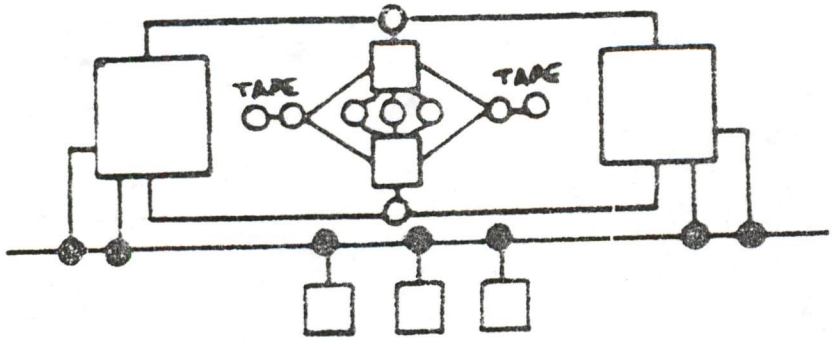


2072
(2073, 2074)



- COMMON FILE SYSTEM

2077



- COMMON FILE SYSTEM
- FULLY REDUNDANT MASS STORAGE SYSTEM
- REDUNDANT ETHER. PORTS
- MULTI-PATH SOFTWARE

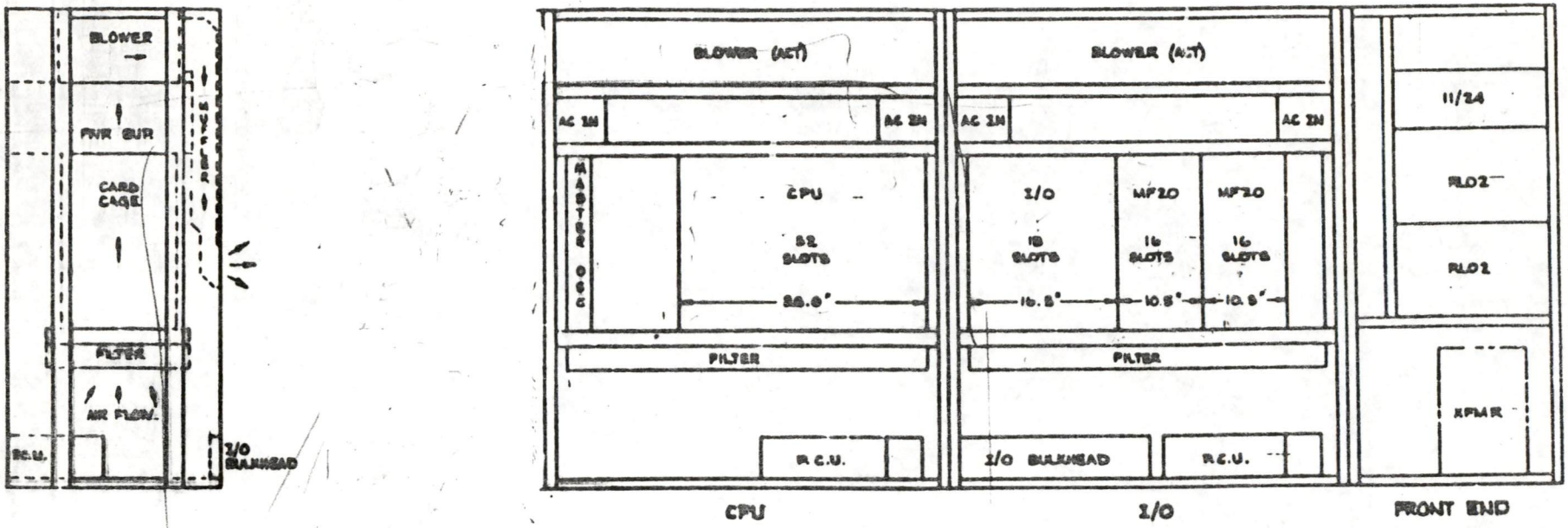
ADVANTAGES AND DISADVANTAGES

ADVANTAGES

1. Cost reduced product.
2. Assembly process that would allow systems to be assembled on the Venus/Jupiter assembly line
3. Minimizes the number of items that must be stocked.
4. Enhances serviceability.
5. This product would meet all present DEC standards.
6. Vastly improved cooling design should improve system reliability.
7. Less labor intensive assembly and harness techniques.
8. More efficient power system should reduce cost of ownership.
9. Frees up manufacturing floor space by phasing out H760, H761, 863 and power supply harness area.
10. Project now becomes the start up focus for Venus / Jupiter assembly line and the MRS volume manufacturing for regulators and AC input modules.

DISADVANTAGES

1. Long time to market.
2. Large development cost.
3. More potential for impact on Venus/Jupiter projects
4. Requires effort from power supply group where resources are a problem.
5. Project now becomes dependent on Venus/Jupiter technology developments and is exposed to risks from problems that may still be hidden.
6. Project now becomes the start up focus for Venus / Jupiter assembly line and the MRS volume manufacturing for regulators and AC input modules.


BLOWER
PWR SUP
CARD CAGE
FILTER
AIR FLOW
B.C.U.
I/O BULKHEAD
BLOWER (A.C.)
AC IN
AC IN
MASTER OSC
CPU
52 SLOTS
20.0"
FILTER
B.C.U.
BLOWER (A.C.)
AC IN
AC IN
I/O
18 SLOTS
MF20
16 SLOTS
MF20
16 SLOTS
16.5"
16.5"
16.5"
FILTER
I/O BULKHEAD
B.C.U.
11/24
RL02
RL02
XFMR
CPU
I/O
FRONT END

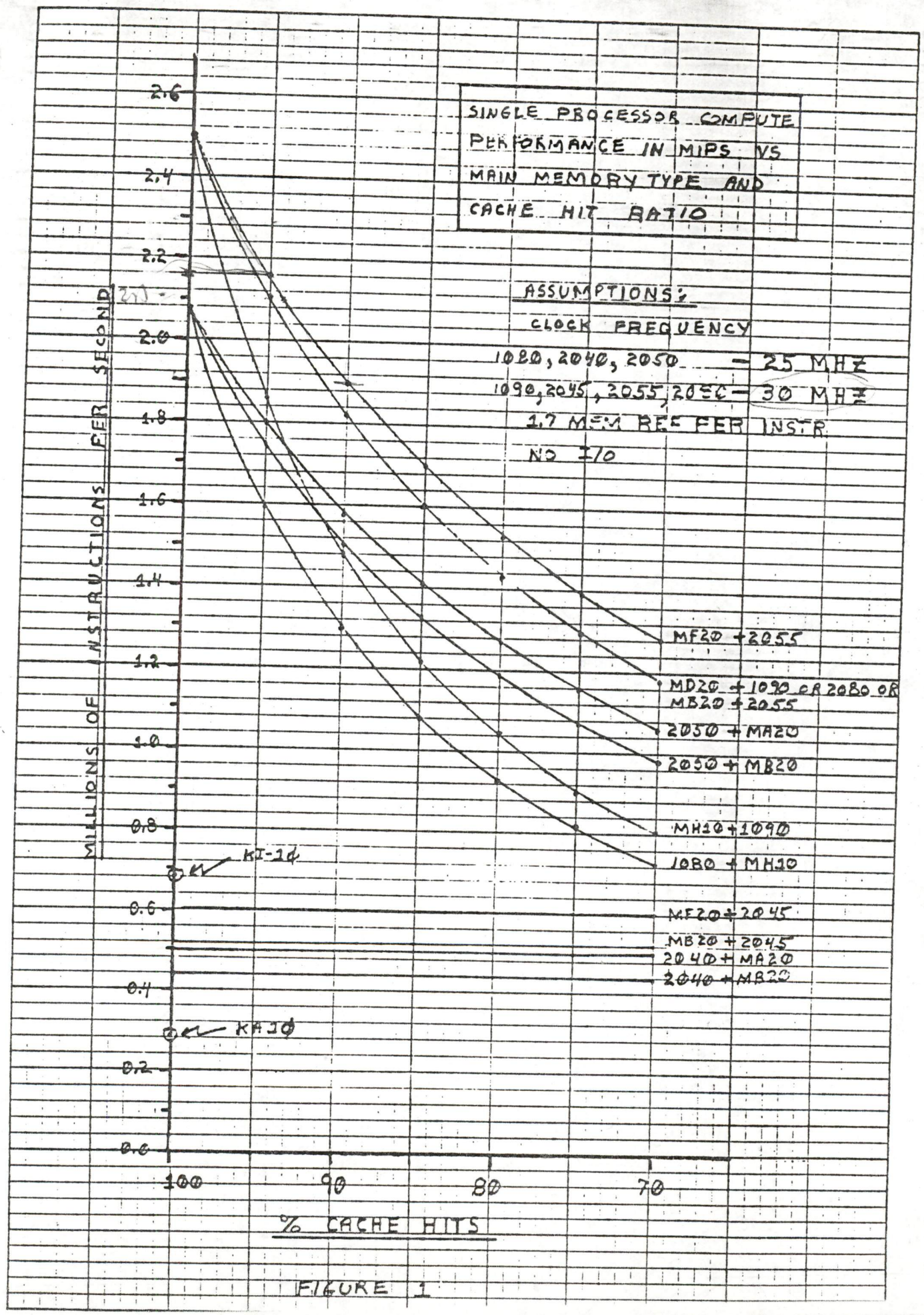


FIGURE 1

compansion of the single processor systems in this section of the memo.

The third group of curves start at the 2.50 MIPS, 100% cache hits point and represents the MODEL B (30 MHZ) group of machines. In all cases the MODEL B machines exceed the performance of their MODEL A counterpart because of the increased clock frequency. The performance of these machines at 90% cache hit is tabulated below.

SYSTEM	MEMORY	MIPS	RELATIVE PERFORMANCE	CLOCK FREQ
1090/1099	MH10	1.48	0.94	30 MHZ
2055	MB20	1.82	1.15	30
→ 2060	MF20	1.90	1.20	30
2080	MD20	1.82	1.15	30

KL10 MODEL B CPU PERFORMANC AT 90% CACHE HITS

The systems with the highest performance in this group are the ones with the MF20 single port MOS Memory and MD20 multiport MOS Memory. Their performance is higher because their memory access time is shorter than that of the MH10 core memories.

Multiprocessor Systems

The compute performance of processors in a multiprocessor may be significantly less than that of a uniprocessor due to reduced cache hit ratio and memory contention.

The cache hit ratio will be lower because the monitor data base must be uncached. This means that references to the data base will go directly to main memory rather than to the cache. As a result the effective cache hit ratio is reduced. On TOPS-10 systems the cache hit ratio for the Master processor of a Master/Slave multiprocessor has been observed to in the 75% to 85% range. This corresponds to a 30% reduction in processor compute performance relative to that of a single processor.

Processors in symetical multiprocessors (SMP) should get higher hit ratio's than the Master processor of a Master/Slave multiprocessor because they should spend more time making memory references to the user address space. Hit ratio's for user mode have been observed to be in the 85% to 95% range on single processor on TOPS-10 systems. SMP processors should get hit ratio's that are worse than single processor systems because the monitor data base will be uncached. As a result SMP processors will most likely get cache hit ratio's in the

80% to 90% range.

Memory contention will cause a reduction in the compute performance of a processor in a multiprocessor system. The magnitude of this reduction is dependent on how fast the memory subsystem can give cycles to satisfy requests from the processors. If memory requests are assumed to be scattered uniformly across the memory address space, then more cycles can be gotten out of the memory subsystem either by increasing the number of memory controllers so more controllers can cycle in parallel or by decreasing the cycle time of the memories.

Current multiprocessor systems use MH10 memories so the only way to get the extra cycles to support requests from additional processors is to increase the number of memory controllers. Future systems will use the MD20 multiport MOS memory. In this case both options are open for consideration in the design of the memory.

To assess the effects of the number of memory controllers and memory cycle time on the performance of processors in a multiprocessor environment a mathematical queuing model for the processor memory subsystem has been developed. The results predicted by this model have been compared with those produced by a very detailed MBOX/MEMORY simulator that was developed for the 1090 Performance Analysis Study. The mathematical model gives results which agree within a few percent with those produced by the simulator for both 1 and 4 processor systems.

The queuing model assumes that all memories and processors are identical. Each processor is assumed to compute for an interval of time, then make a request to memory, wait for the memory access time and any queuing delays, and then obtain the requested data and start the compute cycle again. This basic loop of operations is assumed to repeat forever and memory requests are assumed to be uniformly distributed between the memory controllers on the system.

The model accounts for two queuing delays which exist in the EBOX to memory path. The first is caused by contention between the EBOX and Internal I/O channels for memory cycles. The second is caused by contention between the several processors of a multiprocessor at the memories for memory cycles.

This queuing model has been applied to many of the possible configurations of systems with MODEL A and MODEL B KL10 processors and the results show that as long as the I/O rate per CPU is less than 400 pages per second I/O has no significant effect on the queuing delay experienced at the memories of a multiprocessor. This means that the effects of contention due to I/O and contention at the memories may be treated independently. This is fortunate because it means that CPU compute performance may be treated as being a function of an I/O factor and a memory factor. This significantly reduces the amount of data that must be presented to determine the compute performance of a processor.

Affect of Internal Channel I/O on Compute Performance

The compute performance of a KL10 processors is reduced when data is transfered to and from memory via the KL10's internal channels. How much the compute performance is reduced when I/O transfers are taking place is only a function of the memory type, and the cache hit ratio as long as the I/O rate per CPU remains below 400 pages/second. The amount of performance reduction is a linear function of the I/O data rate presented to the processors data channels.

The percent reduction per 100 pages/second of I/O per CPU is tabulated in Table 2(A) for non cache KL10's and in Table 2(B) for KL10's with a cache. All systems are assumed to be using the RH20 internal data channels.

SYSTEM	CLOCK FREQ	% LOSS IN COMPUTE PERFORMANCE
2040 + MA20	25 MHZ	5.9 %
2040 + MB20	25	3.6
2045 + MF20	30	3.6

(A) NON CACHE KL10 SYSTEMS

		CACHE HIT RATIO		
SYSTEM	CLOCK FREQ	90%	85%	80%
1080 + MH10	25 MHZ	2.1%	2,6%	3.1%
1090 + MH10	30	2.3	3.1	3.6
2050 + MB20	25	1.5	2.1	2.8
2055 + MF20	30	0.8	1.0	1.5
1090 OR 2080 + MD20	30	1.0	1.3	1.8

(B) KL10'S WITH CACHE

Percent reduction in CPU compute peformance per 100 pages per second of I/O per CPU.

TABLE 2

As can be seen systems with no cache suffer the largest performance loss. Systems with cache suffer larger losses when the clock frequency is increased, 1080 vs 1090, and memory access time remains constant, and suffer smaller performance losses as memory access time is reduced.

The percent reduction in compute performance for amounts of I/O which are different from the table can be found by multiplying the value in Table 2 by the actual number of pages of I/O per second divided by 100.

For example if the percent performance loss on a 2040 with MA20 memory due to 250 pages per second of I/O is desired it would be computed to be a

$$\frac{250 \text{ pages/second}}{100} \times (5.9\%) = 14.75\%$$

performance loss.

Note that this performance loss is independent of whether the processor is used in a single or multiple CPU system as long as the aggregate channel I/O data rate remains below 400 pages/second per CPU.

Affect of Memory Contention on Processor Compute Performance

Memory contention reduces the compute performance of a KL10 processor in a multiprocessor system by increasing memory access time. The amount of delay experienced by a request arriving at a memory is a function of the average time a processor computes between successive requests, memory access time, memory cycle time, uniformity of the distribution of request across the address space and the priority algorithm the memory uses for deciding which processor will get the next memory cycle.

The amount of time a processor computes between successive memory cycles was determined by assuming that the average instruction executed by a processor is a 1.7 memory reference fixed point ADD instruction. The equation for this instruction was then used to compute the time between successive memory requests as a function of cache hit ratio (see Appendix).

Memory access and cycle time are a function of the memory type used on the system and are given in Table 1 (b).

****
* d i g i t a l *
****

TO *GORDON BELL

cc: BOB PIERCE @PITB

DATE: TUE 30 NOV 1982 8 09 AM EST
FROM: CHUCK EICHENLAUB
DEPT PITTSBURG GROUP SALES
EXT: 422 2200
LOC/MAIL STOP PT/

MESSAGE ID: 5183307006

SUBJECT: THURSDAY MEETING WITH UNIV OF PGH

By now you should have received Bob Pierce's Briefing EMS for Thursday's meeting. Are you comfortable with your role of convincing them that we are committed to the Jupiter product as well as to continued follow-on 36 bit products?

30 NOV 82 9:37:03 S 01793 DCEM
DCEM MESSAGE ID 5183349722

****
* d i g i t a l *
****

TO *GORDON BELL
BERNIE LACROUTE
cc: see "CC" DISTRIBUTION

DATE: MON 29 NOV 1982 5:15 PM EST
FROM: DEANE CURRAN
DEPT: TECH SALES
EXT: 422 2250
LOC/MAIL STOP PT/422 2250

MESSAGE ID: 5183306706

SUBJECT: UNIVERSITY OF PITTSBURGH 12-2-82 VISIT

***** THIS MESSAGE IS FROM CHUCK EICHENLAUB AND BOB PIERCE *****
The following is provided as background briefing information for the upcoming University of Pittsburgh Visit:

Account History

Since February 1980 Pitt has been working with Digitl to develop and implement a long range computing strategy and has depended heavily on our plans for 36 bit system development and availability in this planning process.

Then and today Pitt's goal is to increase computing capacity 30% per year on the average.

During 1980 Pitt was repeatedly told that Jupiter would be available in 1982. At that time they stated that they would be out of capacity by September 1981.

In May 1980 Pitt began a seven month evaluation of alternative solutions because Digital lost credibility with Pitt in our commitment to Jupiter and 36 bit products.

Weighing the alternatives, in 1981 Pitt chose to add two KL10 processors to their existing dual processors as an interim solution, still believing that Jupiter was on schedule

In April 1982 Pitt visited Marlboro to receive an update on Jupiter. At that time they were told that the system would be announced at DECUS and first customer ships would begin a year later. No mention was made that TOPS 10 would be available 12 months after the first revenue ship of Jupiter.

In August 1982 Pitt was informed that TOPS 10 would not be available until December 1984.

In November 1982 Pitt was told that the announce date for Jupiter would probably slip until spring DECUS. They were also told that this would not impact first revenue ship. I don't believe that Pitt believes us!

The net effect is that we have been promising a Jupiter product to Pitt that won't be available for N +1 2 years.

What Pitt needs to get from their visit

* A strong commitment from Dieter Huttenberger to take a look at Pitt's research work and investigation of where Digital and Pitt might work closely together, i.e., 36 bit, 32 bit, personal computers, and networking software development

* A strong commitment from Gordon Bell that Digital is and will continue to support 36 bit architecture in future systems development.

* If the 36 bit commitment is there; a strong commitment from LCG that we will provide them with a means to handle their growth until Jupiter. I.E., a free 2060, a loaner 2060, a joint development effort in TOPS 10 to 20 conversion.

* A strong message that Digital is committed to the educational marketplace in providing large universities like Pitt the application software needed to handle administrative applications and the hardware to support Digital as Pitt's vendor for the 80's.

* The facts concerning Jupiter as we know them today, announce date, availability.

Pitt attendees and their feelings toward Digital at present

Dr. Paul Stieman, Director University Computer Center
Chairman, Joint Committee for Long Range Planning

- Pro Digital but leaning toward other possible solutions. Paul is very angry and very sensitive in that he feels that he has been betrayed!, by us in his past planning process. Paul is finding it very difficult to support Digital at present. What Paul needs is tangibles that he can sell!

Dr. Neil Timm, Director Institutitonal and Policy Studies,
Office of Planning and Budget
Chairman, Advisory Committe for Administrative Systems Development

- Anti Digital - Very unhappy with constraints of TOPS 10 and TOPS 20 on the administrative application side. He has not stated what might be better, but has expressed his feeling that anything but Digital would.

Dr. J. Norman Bardsley, Professor, Dept. of Physics and Astronomy
Chairman, Executive Committee for Academic Computing

- Pro Digital - Feels VAX provides more alternatives for scientific computing than TOPS 10 or TOPS 20 and supports the idea of distributed processing to handle various academic applications.

Mr. Warren Fugate, University Computer Center
Assistant Director for Systems and Operations

- Pro DEC - Recognized as the universities exper in systems hardware and software. Reviews all computer acquisitions in excess of 15K as a technical consultant to Paul Stieman who must approve these acquisitions.

- Warren is frustrated by the fact that Jupiter has been coming for N + 2 years for the past five years. He needs technical specifics as to what Jupiter s problems are and how long we expect it to take to fix them before he can support future Digital purchases.

Mr. Chuck Heins, University Computer Center
Assistant Director for Administrative Systems Development

- Pro IBM - Chuck has been with the University for only six months and is from an IBM shop at US Air. Chuck is frustrated by the limited amount of administrative software available for TOPS 10/20 when compared to IBM.

UNIVERSITY OF PITTSBURGH REVISED AGENDA
December 2, 1982

10:01	Arrive Logan Airport	
11:00	Arrive Marlboro	
11:15-12:00	Welcome and Introduction	Rose Ann Giordano Ward Davidson
12:00 1:00	Lunch	Rich Whitman Bob Todisco
1:00-2:00	VAX futures	
2:00-3:00	Meeting with Gordon Bell	
3:00-4:00	Meeting with Bernie LaCroute	
4:00-5:00	Round table discussion	
5:00 5:30	Wrap-Up	Ward Davidson
5:30	Limo Depart for Logan	
7:30	Return flight to Pittsburgh	

30-NOV-82 7:45:25 S 01286 DCEM

DCEM MESSAGE ID 5183249139

"CC" DISTRIBUTION

DEANE CURRAN	WARD DAVIDSON	JIM GEORGE
ROSE ANN GIORDANO	RON HEVEY	PER HJERPPE
ANDY KNOWLES	GEORGE LANGER	BEVERLY A STREETER
BOB TODISCO	RICH WHITMAN	RAY WOOD

****
* digital *
****

12/2

TO see "TO" DISTRIBUTION

DATE: THU 11 NOV 1982 3:04 PM EDT
FROM: JIM GEORGE
DEPT: LCG EDUCATION
EXT 231-5597
LOC/MAIL STOP MR2 2/8D2

MESSAGE ID: 5181481311

SUBJECT: DIGITAL'S LONG RANGE PLANS

In a recent customer visit, the University of Pittsburgh represented by Dr. Paul Stieman-Computer Center Director and his assistant Warren Fugate met with Andy Knowles, Ward Davidson and Rose Ann Giordano. The substance of this meeting was the Jupiter-4050 its estimated delivery and performance. Pittsburgh being a Tops 10 site (7 KL10's, 2 Tri Processors, and 1 single) was somewhat unhappy with the answers they received.

Andy realizing the importance of this customer and the investment that Pittsburgh has made in Digital Equipment stated that he would have the following people available to Stieman to discuss Digital's Long Range Plan:

Gordon Bell
Bill Johnson
Bernie LaCroute

Dr. Stieman has requested this meeting, he would be interested in what DEC plans on doing in the PC arena the high end VAX and follow ons to the High-End VAX and the 4050. He is also very interested in alternative approaches to his current situation with current DEC products to include costs and commitments. It is also important to follow up on Andy's second commitment to discuss possible joint ventures in the area of PC's and networking.

The dates that Paul has requested are:

November 19th
December 1st, 2nd or 3rd

Please advise A.S.A.P. of availability, my DTN is 231-5597

Thank you in advance for your cooperation

11-NOV-82 20:19:42 S 04805 MLCG
MLCG MESSAGE ID: 5181474505

"TO" DISTRIBUTION:

JOHN ANDREWS
CARL CROCKFORD
ROSE ANN GIORDANO
BERNIE LACROUTE
RICH WHITMAN
*GORDON BELL
DEANE CURRAN
BILL JOHNSON
RON SPINEK
BOB PIERCE @PITB
WARD DAVIDSON
ANDY KNOWLES
BOB TODISCO

****
* d i g i t a l *
****

TO *GORDON BELL

cc: JOE DINUCCI
STEVE RUVOLO

DATE: WED 1 DEC 1982 4:40 PM EDT
FROM: STEVE RUVOLO
DEPT: 'LARGE ACCOUNT GROUP'
EXT '521 2229'
LOC/MAIL STOP: WR/D

MESSAGE ID 5183410203

SUBJECT: INTEL JUPITER SALE

****

THIS MESSAGE IS FROM ANDREW BARNES INTEL ACCOUNT MANAGER

****

For some time now, we have been promoting the Jupiter to Intel as a mid-level computer. Despite assurances to the contrary, an attitude exists within Intel management that the Jupiter is the last product in the 36-bit line and that we are not committed to follow-on systems. The source of this information is difficult to pin down, but it may even be coming from Intel's contacts within DEC. We need to change the impression of Intel s management that the Jupiter is in fact a whole family of products that incorporate a new architecture and that will be around for a number of years. It would be very helpful to get this message across at the highest levels within Intel.

We understand that you and Gordon Moore will be attending the "Conference on Computers in Science" at the Conrad Hilton Hotel in Chicago, December 6 to December 9. We would like Dr. Moore to leave the conference with the impression of the Jupiter family that we've indicated above. Would you approach Dr Moore and give him this message.

We'll be contacting you after you return from the conference. Thanks in advance for your assistance.

1-DEC-82 20:44:54 S 06141 HZEM
HZEM MESSAGE ID: 5183415071

d i s i t a l

Interoffice Memo

TO: ANDY KNOWLES

cc: see "CC" DISTRIBUTION

DATE: SUN 2 JAN 1983 6:48 PM EST
FROM: GORDON BELL
DEPT: ENG STAFF
EXT: 223-2236
LOC/MAIL STOP: ML12-1/A51

MESSAGE ID: 5186654473

SUBJECT: SUPPORTING THE 10/20 BASED ON HISTORICAL (IBM/AMDAHL) CASE

Learning From History and Amdahl
As a fellow student of history, I know you'll be delighted to learn that Gene Amdahl is giving a talk on his pioneering computers including the 360 at the Museum on March 10 and 11. Please hold the dates. He's also going to spend time with the engineers working on large machines. In addition to being incredibly bright, pleasant and charming, he is an excellent lecturer and story teller. I'm looking forward to the visit.

As a student of history and computer biographer, what I remember is:

1. IBM had a mixed line of computers at the time of introducing the 360 (7040/7094, 7080, 1620, 1401, 1410, 7070 as shown on page 43 of Bell and Newell) crossing scientific and business computing.

2. Gene was on a post Stretch team to build a follow on scientific computer to the successful 7090.

3. B.O. Evans put together the 360 team with Amdahl, Brooks and Blaauw to make a new range of machines for both business and scientific use that would cover everything down to the small computers (the 1130 and 1800 were introduced as a result). SDS did the range bit first, but the control store technology permitted IBM to do a whopping range. Basically, the machine was designed right because they provided a 32-bit address field even though the use in the 360 was just awful!

4. There were pressures to continue every successful line that IBM had, but IBM bet it all not unlike the Fortune article stated.

5. Univac plugged along, evolving the 1103 (1108... 1190), and continued to sell YUKy military machines and evolve the Univac I (now the III), adding some other business and minis. Later they bought the Spectra series, but that's another story. The Univac Series has grown at about the same rate as the 10, with more in the field, and the users are relatively fanatical about it. They don't want Univac to ever stop making it.

6. GE/Honeywell evolved their warmed over 7090 (the 635). They also marketed the H200 that was perceived to be compatible with the 1401. It wasn't but it was marketed well and it got them in the door for a few years, but the machine wasn't really good enough to last even though Honeywell evolved it endlessly till its death. Honeywell's only real computer business is with the old 600 series. The Multics

version is owned by a few zealots, but the other users like the system, and are relatively fanatical about it. They want the series continued forever.

7. Meanwhile at IBM the memory management problem finally started being solved in 1972 with the 370 by providing a 24 bit VA, but because the basic 360 architecture was good enough, the design was capable of being evolved. The 32 bit VA was finally added, and everyone acknowledges the costly mistake.

8. Amdahl may tell the wonderful story of the poor management decision post Watson that forced him to found Amdahl Corp. to build large 360's. By now, he headed the San Jose Lab and had redirected a subsequent machine there to become a 360 and wanted to take it to market.

9. Amdahl Corp. started up to build a 360, IBM fixed the address problem (the 370), and virtually wiped the company out because he had designed the wrong machine (a 360). He took a couple of years for the redesign, but he found friends at Fujitsu who helped him out financially in return for rights to the gate array and packaging technology, the keys to fast computers.

10. Lots of companies build 370 compatible products because it basically does the job, and users regard it as a commodity and standard. This is also the situation in the microprocessor business today... but it's unclear that the semicomputer companies (Motorola and Inte)l make much at it.

11. The basic 360 was a classic design because it could be extended once in the form of the 370. It should be made in its current form for probably another 20 years. I expect it to have a major specialized grafting as IBM tries to build high speed for scientific processing either in the form of a new attached processor or extensions to the 370 along the lines that the Japanese introduced. Here, I'm anxious to talk with Gene about this.

12. The pre-360's designs at Univac and Honeywell were bad, and evolutionary designs haven't gotten any better. They grow the base a bit, but basically it's a replacement market.

13. I have a talk that brings in other examples, but the one recurring lesson (to me): A company that listens to its customer base too long will miss major markets and get wiped out of its current position if it simply follows evolution. There are NO exceptions except the 370 (and PDP-10). The could both be evolved because of good basic design.

Given this memory, here's what I get that's relevant to our situation:

1. The 360/370 are quite similar to PDP-11 and VAX; the PDP-10 is like the 7090 (even has the 36 bit word and fond memories). We used a compatibility mode to deal with the backward evolution. The range goals are different, since we are trying to have a range of sizes down to Personal computers when we finally achieve MicroVAX. Also, we want to use the two machines (the 11 and VAX) together over time to cover our range instead of being religious about having a single architecture. We are much more flexible in architecture to go after

different needs! Thus, we have a much more conservative and evoutionary view.

2. In both the 11 and VAX cases, we adopted a much more evolutionary view by first buildins a machine 11/20, followed by the 05-40-45, and the 780, followed by 730-750-790 asap. This means that we keep on buildins whatever we were buildins before too (the PDP-10 and PDP-8), with the exception of the PDP-15 (I'm still amazed at your ability to set it to stop) because it's too risky to stop because we misht lose some piece of a market. Our attitudes about chanse aren't too different from Univac which is also very ensineerins oriented. The only difference was that we had a much better product base and were in an expandins rather than stable market sesment.

3. One difference between us and IBM is that within IBM there seems to be a stronser seperation between business and ensineerins. I believe ensineers runnins a company are more conservative. As such, IBM senerally soes for the risht solution (which I think is VAX based on product soodness) versus evolution where one bets on everythins. We could never make a decision like the 360 knowinsly. They have siven up lines (the 1130/1800, System 7, 9, 34 etc.) when they have needed to or when they have a better alternative (System 38, Series 1, and PC).

4. Notice IBM always stop poor, deadend machines and replaces them with what should be the most competitive base product in that price band. They also try to minimize the band/market overlap even they often use some crazy sesments (es. the 8100 as a comm machine) instead of sp machines (the Series 1... which is finally cleanins up in this area) due to havins many ensineerins sroups.

Bottom Line (Based on History)
We will not stop a computer line until our customers tell us to...
which is probably not a bad and too costly philosophy.

"CC" DISTRIBUTION:

ROSE ANN GIORDANO
KEN OLSEN
WIN HINDLE
JACK SHIELDS
BILL JOHNSON
JACK SMITH

d i s i t a l

Interoffice Memo

TO: ANDY KNOWLES

cc: see "CC" DISTRIBUTION

DATE: SUN 2 JAN 1983 6:55 PM EST
FROM: GORDON BELL
DEPT: ENG STAFF
EXT: 223-2236
LOC/MAIL STOP: ML12-1/A51

MESSAGE ID: 5186654476

SUBJECT: QUESTIONS YOISE ON MY FULL HEARTED SUPPORT OF THE 10/20

In addition to learning from history, you raised other issues:

1. On VAX conversion, I agree, we're likely to lose a customer UNLESS we approach the problem the right way. (I have one case study, the University of Pittsburgh that looks to be a counter example!) Note that I have always agreed: left alone, 10/20 customers who don't know our full direction (including VAX) will switch to IBM unless we sell. Unfortunately, it requires carefuly selling and work to not lose.

2. Positioning 10's with VAX has to be done. I have a seperate memo on this that's coming around. It fundamentally says:
780 is faster than a KL on Cobol by 30%
780 is faster than a KL on DBMS
KL is faster on record I/O Cobol, unless data is sequental
KL is faster than a 780 by 20% on Fortran, unless double precision
KL can be a factor of 2 faster than a 780 on strictly integer problems until one plays with the VAX data structures
KL's will handle maybe 1.5 times the timesharing load of the same VAX confisuration, but VAX with lots of memory may increase it

2A. I therefore make a 780 at about 2/3- 3/4 a KL, but a 782 should do more work than a KL. I believe a 780F (the 1.5 x 780) now in test should beat a KL on every benchmark and load, especially with lots of the new memory based on 64K chips. Furthermore, the cost of the system should be better by at least by a factor of 2.

2B. Future machines:
Jupiter at 2.3 x KL I hope on the first go around.
Venus should be >4 x 780, or about 3 x KL
Nautilus is too early to tell precisely, but because it's a dual processor from the ground up should do well on both cost and performance.

3. My business acumen. Since I simplistically rank product goodness as the key to all current and future business, I could understand why you believe I lack it, but believe you're wrong.

4. Smoke damage is costly, and it really hurts me to be accused of this. Here, I think we'd better have a talk about what you want me to do to change, because I sure don't want to do anything but help keep customers and build a follow-on 10/20. Communication is strictly amongst us, so I can't understand a cost. Would like to get input from this group too regarding my

behavior.

5. Heros... who they? It would be nice to be one or find one in here maybe next to the pony that we should soon discover somewhere near the Jupiter project.

As ever, I'm dedicated to preserving our customers and setting us a follow-on 10/20.

Gordon

"CC" DISTRIBUTION:

ROSE ANN GIORDANO	WIN HINDLE	BILL JOHNSON
KEN OLSEN	JACK SHIELDS	JACK SMITH

****
* d i g i t a l *
****

TO: *GORDON BELL

cc: see "CC" DISTRIBUTION

DATE: TUE 14 DEC 1982 9:48 AM EDT
FROM: ANDY KNOWLES
DEPT: SMALL SYSTEMS GROUP
EXT: 278-4567
LOC/MAIL STOP: UP2-4/UP2-4

MESSAGE ID: 5184730222

SUBJECT: DECSYSTEM 10,20

I think I understand, more than anybody, your hangups vis-a-vis the 10/20, its address limitations, etc., etc., etc. As a pure computer scientis you are completely justified, as a marketer and protector of our customer base you are completely wrong.

I had dinner with Gene Amdahl last Wednesday night in Monterey, California. You remember him. He was the architect of the IBM 360 series and has bidden that product for almost 20 years. And if his 25-35 MIPS Triology product comes off, the 360 will live til at least 1990 when, who knows, something may replace the 10,20,360,VAX and the 60's, 70's crowd.

What is my point? We have a purely religious customer base in the 10/20 space. They are committed to the 10/20. A conversion to VAX is not easy nor is it easy to convert to IBM. But when they are faced with conversion they don't convert to VAX, they convert to IBM's 43XX's and 30XX's. Got it? Let me repeat -- they convert to IBM. So your impassioned technical argument is beautiful from a computer scientist's point of view, but stinks from a business point of view. I love your memos, but question your business acumen.

I fully understand the conversion issue!
Selling more

I'd like a clear positioning of JUPITER, VENUS, NAUTILUS, SON OF STAR, etc.

Gordon - your smoke is costing us dearly. And I also understand your hero approach.

/sc

14-DEC-82 10:54:32 S 02196 MLCG
MLCG MESSAGE ID: 5184704899

"CC" DISTRIBUTION:

ROSE ANN GIORDANO
KEN OLSEN
WIN HINDLE
JACK SHIELDS
BILL JOHNSON
JACK SMITH

APR 17 1979

CHUCK MONROE
to
Bill Kaiserman

FOONLY INCORPORATED: HISTORY, CAPABILITES, AND METHODS

Foonly Inc. is a small computer hardware and software company, with the capability to develop and produce both complete systems and subsystems of many types. Our particular aim is to serve customers whose special requirements are not efficiently met by the standardized offerings of larger companies.

The company was formed in 1976 by three experienced computer scientists from Stanford University and M.I.T. All the principals of Foonly Inc. have a history of ten years or more of designing and implementing computer hardware and software, at university projects, as independent consultants, and occasionally in industry. We combine extensive use of computer-aided design tools with a method of manufacturing through subcontractors to enable us to achieve very unusual flexibility and economy in providing custom tailored systems.

The newness and small size of Foonly, together with the nature of what we offer, has occasioned some natural doubts and questions regarding the risks of dealing with us. We recognize the importance of this subject, and would like to have all such issues discussed frankly with us. It is our belief that examination of our record, coupled with detailed explanations of our somewhat unusual modes of operation, will provide adequate assurances of the credibility of our offerings. Below are short explications of some commonly raised points; please allow us to discuss in depth with you any matters not covered to your satisfaction.

What We Have Done--

We have designed, built, checked out, and delivered the first Foonly model F1 processor; it has been in heavy use since March, 1978, and has experienced one hardware failure during that period. This machine consists of 5000 ECL logic circuits in a complex, pipelined architecture; it is similar in performance to an IBM 370/68. It has exceeded its performance specifications, and was completed in 21 months (25% longer than estimated).

In the first quarter of 1979, we are delivering three model F2 computer systems, including our own main memory, disk drive controllers, tape drive controllers, and several special I/O interfaces.

Foonly, Inc. has also delivered two different versions of the model C1 Multiple Data channel. This device is of a size and complexity similar to that of the model F2 system (about 1200 TTL circuits). Neither C1 was more than 18 days late; the first has been in operation for two years, and the second for one year.

How We Do It--

Our capabilities are based on the use of techniques of design and production which are somewhat unusual outside research projects.

In developing (or modifying) a hardware design, we make heavy use of computer design aids (mainly the SUDS system). This increases the efficiency of the design process and enhances the accuracy of the result.

Equally important, it eliminates entirely many labor-intensive and error-prone manual procedures, including drafting and wire listing. Foonly Inc. can therefore execute design projects more rapidly and much less expensively than is usual.

Construction of hardware is carried out almost entirely by subcontractors and vendors. Wiring, in particular, is done using automated machinery controlled by output tapes from our computer aided design system. Only the final stages of assembly are performed by Foonly personnel. The result is high quality and accuracy of fabrication coupled with very low capital and manpower requirements for Foonly.

Checkout of our hardware is greatly facilitated by the extensive documentation produced automatically by the design system, and by the accuracy of the machine-controlled wiring. Most of our designs include special logic to allow the use of an on-line computer to aid checkout (the F1 includes its own diagnostic computer).

It should be emphasized that we construct all units using the above techniques, so we encounter no new difficulties going from prototype construction to production models. The automated design system ensures that corrections to errors discovered in the prototype are accurately incorporated into later units, and that quality of fabrication does not deteriorate.

SOFTWARE

The background of the principals of Foonly Inc. includes extensive programming experience, including maintenance and development of timesharing monitors, compilers, assemblers, and numerous other types of system and applications programs. In addition, Foonly has available the services of several highly qualified consultants for all kinds of software work.

All the software being offered with the F1 and F2 systems is free from DEC licensing control. Most of it is in the public domain, and has been obtained from universities and customers. Certain items will be proprietary to Foonly Inc. or will be acquired on license from other parties; for these items there may be a licencing fee.

SUPPORT AND DOCUMENTATION POLICY

All hardware is delivered with the extensive documentation generated by our computer-aided design system, as well as programming and theory of operation manuals. The price of a system includes training of the customers maintenance personnel, but we can usually arrange commercial maintenance services if desired. Design defects are repaired by us free of charge.

All software is delivered with program sources and all available documentation for no additional charge. At the customers option, we offer various levels of system and applications software maintenance, which we provide through our software consultants.

The Foonly Model F1 Processor

* Pipelined architecture
* Compatible with DEC PDP-10
* 2 times as fast as KL-10
* Microcoded execution unit
* High speed ECL 10K logic
* Internal multi-port I/O processor interface
* Console computer for control and diagnostics

The F1 is a cpu which is program compatible with the DEC PDP-10 line and twice as fast as the fastest DEC model. Microcoded control allows the F1 to provide detailed compatibility with KA10, KI10, or KL10 cpu's.

A pipelined architecture provides separate functional units for instruction fetch, address and operand preparation, and microprogrammed instruction execution. A high degree of overlap between these separate units allows most instructions to be processed in 1 or 2 machine cycles.

Memory fetches are mediated by a central M-box which contains a 4 way set-associative cache. The M-box can do the translation of a virtual address and fetch the addressed word from the cache in 1 machine cycle.

A small computer (mostly PDP-10 compatible) called the Console Computer (or CC) is a permanent part of an F1. The CC has extensive connections to the logic and memories of the F1, and programs on the CC provide the functions of console lights and controls. The CC is also the main tool for diagnosing the F1; using the CC's ability to examine and modify the states of the F1's logic and microprocessor, programs on the CC can determine in detail the nature and location of any failure. The CC is provided with a modem for connection to a telephone line, so that diagnostic procedures can be carried out from a remote location.

The F1 is uses a very simple single-port, four word wide memory. Only one port is needed on the memory because the F1 includes eight "APA" ports for the connection of memory using I/O devices (i.e. data channels), as well as an I/O bus similar to the DEC one for simple devices. The eight APA ports provide memory mapping (paging) for the devices, which we believe to be a unique and useful feature. These ports also operate in a way that guarantees consistency between memory data as viewed from the cpu and various devices, despite the presence of the cpu's cache. Flushing or sweeping of the cache is never necessary.

ECL 10000 series logic is used in the F1, and about 5000 16-pin packages are required (compared to about 4000 for the KL10). Total power required is about 5 KW, and no special cooling of the ambient air is needed.

SUMMARY OF THE CHARACTERISTICS OF THE F1

The F1 is twice as fast as any System-10 or System-20 processor.

The F1 can use both IBM compatible and DEC compatible I/O gear.

The F1 allows the I/O device connection flexibility of a System-10 with 8-port external memory, while using much less expensive internal memory (and requiring no flushing or partial disabling of the cache).

The F1 provides memory mapping (paging) for its I/O devices.

The F1 possesses unique hardware aids for performance analysis and debugging of programs, for both systems and application software.

The F1 offers low maintenance costs because a complete spare parts inventory is inexpensive, and low down-time because of extensive facilities for remote diagnosis of faults.

STANDARD I/O EQUIPMENT

The F1 can use either IBM plug-compatible or System-10 compatible I/O devices (or both):

Any number of IBM compatible device controllers may be connected to an F1 through one or several Foonly model C1 Multiple Data Channels, which simulate 'block-multiplexer' IBM channels; any desired degree of parallel operation of devices can be achieved, up to a very high total data rate. This approach provides access to the large (and competitive) selection of disks, tapes, and line printers offered for IBM systems by independent peripheral manufacturers. The C1 connects to the system via the Auxilliary Processor Adapter described below, and so operates with memory mapping and no cache flushing.

For the connection of DEC type devices, the F1 can be equipped with the Brand X Conversion Unit, or XCU. The XCU provides a System-10 compatible I/O Bus and from two to six ports which accept a standard DEC memory bus. The memory ports connect to ports of the Auxilliary Processor Adapter (see below), so the performance of the DEC memory-using devices is enhanced by its mapping and buffering features.

Foonly, Inc. can provide a simple data-communication (terminal) controller for up to 64 lines, and a line printer controller for various common types of line printers.

SPECIAL I/O and the APA

In the design of the F1, particular attention was given to the ease of attaching high performance I/O devices, both standard and custom, as well as to the simplicity of the system programming needed to utilize them. We feel that both points have been given insufficient consideration in the past.

In the System-10, multi-port memory systems have been required for high speed I/O; such memories are expensive, especially since the new CPU's use four ports themselves. With the advent of high performance CPU's using cache memories and memory mapping ("paging"), special programming complications have arisen; this is because the CPU and devices on other memory ports usually find a given set of data at different addresses, and sometimes find different data even if the addresses are the same. The need for multi-port memory can now be avoided in DEC systems by use of "Mass Bus" devices, but at considerable cost in flexibility: since the "Mass Bus" has no address part, it is not equivalent to a memory port, and can be used only by devices designed not to need random access to system memory.

The F1 reduces these difficulties by means of a unique design feature. In addition to a processor-controlled bus for simple I/O gear, the F1 contains an Auxilliary Processor Adapter, or "APA". The APA provides access to the F1 system's memory for devices such as data channels for disks and tapes, high speed communication gear, and other subsidiary processors. There are eight physical APA ports, each of which can access any of 32 "virtual memory ports". A virtual memory port provides fully addressable random access to main memory, exactly like a

connection to a multi-port memory system, and also performs three other functions:

1) Cache consistency checking -- Like other fast processors, the F1 uses a high speed buffer memory, or "cache". Normally a cache greatly complicates the use of I/O devices connected to multi-port memories; usually the cache must be "flushed" fairly often or disabled entirely for certain regions of memory. The APA operates so as to relieve the system software of any need to handle such complications.

2) Data buffering and pre-fetching -- For normal serial I/O transfers, the APA buffers (and optionally pre-fetches) data in blocks of four words. This reduces the loss of system performance caused by memory interference between the CPU and the I/O devices, and also makes the system memory appear faster to the I/O gear.

3) Memory Mapping -- The addresses of the I/O data transferred may be mapped using the same page tables that the CPU uses. This unusual capability avoids much system programming pain, and makes feasible the direct use by user programs of special I/O gear.

PROGRAM DEBUGGING AND PERFORMANCE ANALYSIS AIDS

Three unusually powerful hardware aids are provided by the F1 for monitoring the operation of either system or applications programs; two of these mechanisms are believed to be unique.

The first is the PC History Memory, which maintains a record of the last 1024 instructions executed (or the last 1024 branches taken). It can be enabled to operate in executive mode, user mode, or both; it can also be activated only within a selected region of memory. The information it collects is invaluable during the checking out of new programs, and can be used for flow-tracing analysis of correctly working routines in order to evaluate their efficiency.

Complementing the PC History Memory are four Memory Address Break registers (MABR's); the MABR's are loaded under program control, and can be used in executive or user modes. A given MABR can be set so that when the program refers to the memory location in the MABR in some particular way, the processor either halts, executes a trap (to either the system or the user program), or increments a counter. Two of the MABR's can be set with a pair of bounds, and detect memory references anywhere between those limits. The MABR's are most useful for both debugging and analyzing the operation of all kinds of software.

Four general purpose system monitoring counters are also included in the F1. They can count clock cycles, instruction executions, memory references, MABR hits, and other events, and can be enabled by conditions such as user mode or interrupt level. The counters are thus able to perform all ordinary accounting functions, as well as collecting statistics on many details of system operation.

Utilizing the combined capabilities of the History Memory, MABR's, and counters should make possible a previously unobtainable degree of detailed knowledge about system operation.

MAINTENANCE

There is a functional test program, which may be run under timesharing. It checks all aspects of processor operation, beginning with a rapid verification of basic functions, and proceeding to a series of detailed tests of the more complex features of the machine. If this program detects any failures, a fault location routine (running mostly on the Console Computer) attempts to localize the malfunction; usually this can be done to within two to five circuit packages. (This high level of failure localization is made possible by the unusually extensive diagnostic connections between the Console Computer and the internal logic circuits of the F1.) The implicated IC's are then replaced as a group, a procedure taking about five minutes.

This style of maintenance enables a complete set of spare parts to be acquired for two or three thousand dollars, because only a few IC's of each of thirty or so standard types are needed. It will never be necessary to suffer addtional down time waiting for replacement cards.

For the occasional failure too subtle to be found by the automatic fault location routines, the F1 system is equipped with uniquely powerful aids for the expert debugger; these aids are mostly based on the intimate access to the low-level circuitry of the F1 enjoyed by the Console Computer. Using a display terminal, the debugger can examine a wide selection of dynamic displays showing logic states and register contents, referring to signal names symbolically. He can load and control test routines operating at the Console Computer, the microcode, or the macrocode level; he can set individual registers or control signals to desired states, stop and single step the main clock or vary its speed, and even adjust the skew of various timing signals.

The software debugging aids of the F1 are equally useful here, too: the PC History Memory can be set to record the microcode pc (and there are several more history memories specially devoted to hardware debugging); the counters can be used to control multiple stepping of the clock; and the MABR's can be set to watch the internal operation of the memory control. Finally, the

Console Computer has a digital interface to a "logic analyzer" instrument, which allows it to observe arbitrary logic signals through moveable probes.

These elaborate maintenance procedures can all be performed from a remote location, via a phone line connection to the Console Computer. On-site personnel need only sufficient training to replace IC's (and perhaps place logic analyzer probes), given explicit directions. Even computer operators could do it.

F1 HARDWARE MAINTENANCE AND SOFTWARE SUPPORT

Hardware Maintenance--

Foonly Inc. will provide complete hardware maintenance service in either the San Francisco Bay area or the Los Angeles area; the monthly rate for a basic system is $2500.

Alternatively, we will provide free training for the customer's own service personnel, together with complete diagnostic programs and maintenance documentation. See the F1 Processor description for a discussion of the unusual maintenance features of the F1. A complete set of spares costs $5000.

SOFTWARE

PDP10 Compatibility--

The F1 can be microprogrammed to be software compatible with any model of the DEC PDP10 line. An F1 system therefor has available a huge amount of sophisticated software, much of which can be obtained free or for a nominal charge.

Timesharing Monitor--

An F1 system will normally be delivered with the Foonly model M1 timesharing monitor, at no extra charge. This is a modified version of Tenex (release 134), a highly successful system in use at many installations; M1 preserves all the features of Tenex, and includes support programs for file transfer, etc. Most user programs from other PDP10 systems will operate under M1 with no modification. All source code, listings, and documentation for M1 will be provided free of charge; system software maintenance service is available in California for $750 per month.

If the customer has available some other monitor, the F1 can be delivered running that monitor, after appropriate changes are made to the disk and tape control sections of the code. Foonly Inc. will provide quotations for making such changes.

Conversion of User Programs and Custom Monitor Features--

Fconly Inc. provides a consulting service for making the changes which are sometimes needed to convert a user program from one system for use on another. A typical conversion costs around $1000.

Custom extensions to the timesharing monitor will also be made on a consulting basis. An example might be monitor code to support a special I/O device of some kind; for most devices this would probably take one to two weeks.

FOONLY MODEL F1 PRICE and EQUIPMENT LIST

21 August 1978

F1 CPU $300,000

Including Console Computer, Auxilliary Processor Adapter, etc. See processor descpription for complete list of features.

MEMORY

512k words	$60,000
1024k words	$100,000
2048k words	$185,000

MOS memory with error-correcting code
Larger and smaller sizes available

C1B DATA MULTIPLE DATA CHANNEL

1 channel	$20,000
2 channels	$25,000
3 channels	$30,000
4 channels	$35,000

Each channel either contains a controller for 1 to 8 disk drives (see below) or emulates an IBM "Block Mulitplexor Channel" (for connection of any IBM compatible device controller).

For more than 4 channels, several C1B's can be used.

DISK DRIVES

capacity	price from mfr.	price from Foonly Inc.
80MB	about $7,000	$11,000
300MB	about $12,000	$16,000

Drives plug directly into C1B channel - no controller needed.

High performance, latest removable-media technology. Made by CDC, CalComp, Ampex, and perhaps others.

TAPE DRIVES

	Price from STC	Price from Foonly
controller (for 1 to 8 drives)	about $20,0000	$25,000
800/1600bpi, high speed, self-loading drive	about $16,0000	$20,000

Any IBM plug-compatible tape system will work; available from STC, AMPEX, IBM, and others.

SERIAL LINE INTERFACE

128 lines $10,000

Can be customized.

LINE PRINTER CONTROLLER

-for Data Products or similar printers $5,000

Can be modified for any printer. Printers cost from $10,000 up (about $25,000 for 600LPM printer).

CUSTOM I/O INTERFACES OR DEVICES QUOTED ON REQUEST

ANY IBM PLUG-COMPATIBLE DEVICE CAN BE USED WITHOUT MODIFICATION.

KL at Stanford

Foonly

Ratio

1 - BASIC CLOCK CYCLE IS 40 NSEC.
2 - INDEXING TAKES 40 NSEC. → 0
3 - INDIRECT TAKES 200 NSEC. — 200
4 - INDEXING AND INDIRECT TAKES 320 NSEC. — 200
5 - MOVEI TAKES 320 NSEC. — 100 — 3.2
6 - MOVE FROM AC TAKES 440 NSEC. — 100 — 4.4
7 - MOVE FROM MEMORY TAKES 480 NSEC. — 200 — 2.4
8 - HRR FROM MEMORY TAKES 520 NSEC. — 200 — 2.6
9 - SETOM 0 TAKES 680 NSEC. — 100 — 6.8
10 - JRST TAKES 360 NSEC. — 200 — 1.8
11 - JSR TAKES 681 NSEC. — 200 — 2+
12 - PUSHJ TAKES 841 NSEC. — 200 — 4.2
13 - ADD FROM MEMORY TAKES 520 NSEC. — 200 — 2.6
14 - MUL (9 ADD/SUB - 18 SHIFTS) TAKES 2.44 USEC. 2.20 — 1.1
15 - DIV TAKES 5.50 USEC. — 1.1 —
16 - FIX A FLOATING POINT ONE TAKES 1.04 USEC. — 400 —
17 - FLTR AN INTERGER ONE TAKES 1.84 USEC.
18 - FAD (1 RIGHT SHIFT) TAKES 1.88 USEC. — 1.1 —
19 - FAD (8 SHIFT RIGHT - 3 LEFT) TAKES 2.16 USEC. — 1.1 —
20 - FMP (7 ADD/SUB - 14 SHIFTS) TAKES 2.80 USEC. — 3.2 —
21 - FDV TAKES 5.72 USEC.
22 - DMOVE FROM MEMORY TAKES 800 NSEC. 250 — 3+
23 - DFAD (1 RIGHT SHIFT) TAKES 2.36 USEC.
24 - DFAD (8 SHIFT RIGHT - 1 LEFT) TAKES 2.36 USEC.
25 - DFMP (7 ADD/SUB - 32 SHIFTS) TAKES 4.84 USEC.
26 - DFDV TAKES 10.25 USEC.
27 - CONO PI TAKES 1.92 USEC.
28 - CONI PI TAKES 3.36 USEC.
29 - DATAO APR TAKES 1.56 USEC.
30 - DATAI APR TAKES 1.76 USEC.
31 - MOVE TO MEMORY TAKES 840 NSEC. — 150 — 5+
32 - LOGICAL SHIFT (35 PLACES LEFT) TAKES 640 NSEC. — 100 — 6.4
33 - LOGICAL SHIFT (35 PLACES RIGHT) TAKES 760 NSEC. — 100 — 7.6
34 - LOGICAL SHIFT COMBINED (71 PLACES LEFT) TAKES 1.04 USEC.
35 - LOGICAL SHIFT COMBINED (71 PLACES RIGHT) TAKES 1.08 USEC.
36 - INCREMENT BYTE POINTER TAKES 1.00 USEC.
37 - INCREMENT AND LOAD BYTE TAKES 1.44 USEC. — 300 — 4.7
38 - INCREMENT AND DEPOSIT BYTE TAKES 1.80 USEC. — 300 — 6
39 - JFCL TAKES 360 NSEC. — 100 — 6.8
40 - CAI TAKES 480 NSEC. — 100 (200 FOR CAIA) — 4.8 (2.4)
41 - JUMP TAKES 480 NSEC.
42 - CAM TAKES 600 NSEC.
43 - EQV AC TO AC TAKES 480 NSEC. — 100 — 4.8
44 - EQV MEMORY TO AC TAKES 520 NSEC. — 200 — 2.6
45 - SETOB TAKES 840 NSEC.
46 - AOS TO MEMORY TAKES 840 NSEC. — 250 — 3.1
47 - EXCHANGE AN AC WITH AN AC TAKES 640 NSEC.
48 - EXCHANGE AN AC WITH MEMORY TAKES 840 NSEC.
49 - EXECUTE TAKES 641 NSEC. — 200
50 - BLT MEMORY TO MEMORY TAKES 1.92 USEC.
51 - BLT AC TO MEMORY TAKES 1.88 USEC. — 100
52 - DATAI TAKES 7.21 USEC.
53 - DATAO TAKES 7.52 USEC.

THE FOONLY F2 SYSTEM

- o PDP10 compatible
- o Price about one half that of equivalent DEC system
- o Complete system- standard and custom peripherals
- o Free timesharing monitor and support programs
- o Hardware and software maintenance available

HARDWARE

Processor--

An F2 system is based on the model F2 processor, a microprogrammed computing engine capable of emulating any of the versions of the PDP10; its performance level is about equal to that of a 2020. The F2 is built of TTL integrated circuits, and comes with up to 1 million 36-bit words of MOS memory (with ECC), which is housed in the same 19-inch cabinet as the processor. This cabinet also contains all the I/O controllers, the power supplies, a control/maintenance panel, and one magnetic tape drive. All integrated circuits are mounted on plug-in cards for ease of maintenance. The maintenance panel can be made unpluggable as an option. The machine needs no air conditioning at normal room temperatures.

Basic Peripherals--

Up to eight 300MB, high performance, removable-pack disk drives can be plugged directly into the internal disk controller; additional controllers are optional. The drives are made by CDC, CalComp, Ampex, and others, and may be purchased through Foonly Inc. or direct from the manufacturer.

Standard tape drives for the F2 are modern, medium performance 9-track drives made by Kennedy and Pertec; from one to four drives plug in to the internal tape controller. Controllers for higher performance drives can be supplied as an option.

Our standard data line scanner interfaces 32 or 64 terminals. Custom communication gear can be supplied to meet special requirements.

A line printer controller is available for Data Products and other similar line printers; interfaces to other types of printers or plotters will be provided as options.

Special I/O Equipment--

The F2 has a general purpose I/O interface called the F Bus, which allows very simple and efficient connection of custom hardware devices of all kinds. The F Bus provides access to both the processor and memory. Foonly Inc. will provide consulting services for the design and/or construction of special devices, or of adapters for connecting the customer's own devices.

Hardware Maintenance--

Foonly Inc. will provide complete hardware maintenance service in either the San Francisco Bay area or the Los Angeles area; the monthly rate for a basic system is $ 500.

Alternatively, we will provide free training for the customer's own service personnel, together with complete diagnostic programs and maintenance documentation. Repair of electronic failures will be simple and rapid, since all circuits are on plug-in cards; a set of spare cards for the mainframe, memory, and standard I/O controllers costs $10,000.

SOFTWARE

PDP10 Compatibility--

The F2 can be microprogrammed to be software compatible with any model of the DEC PDP10 line. An F2 system therefor has available a huge amount of sophisticated software, much of which can be obtained free or for a nominal charge.

Timesharing Monitor--

An F2 system will normally be delivered with the Foonly model M1 timesharing monitor, at no extra charge. This is a modified version of Tenex (release 134), a highly successful system in use at many installations; M1 preserves all the features of Tenex, and includes support programs for file transfer, etc. Most user programs from other PDP10 systems will operate under M1 with no modification. All source code, listings, and documentation for M1 will be provided free of charge; system software maintenance service is available in California for $[illegible] per month.

If the customer has available some other monitor, the F2 can be delivered running that monitor, after appropriate changes are made to the disk and tape control sections of the code. Foonly Inc. will provide quotations for making such changes.

Conversion of User Programs and Custom Monitor Features--

Foonly Inc. provides a consulting service for making the changes which are sometimes needed to convert a user program from one system for use on another. A typical conversion costs around $1000.

Custom extensions to the timesharing monitor will also be made on a consulting basis. An example might be monitor code to support a special I/O device of some kind; for most devices this would probably take one to two weeks.

F2 PRICE AND EQUIPMENT LIST -- 17 August 1978

F2 CPU	256k mem	$68,000
	512k mem	$90,000
	1024k mem	$140,000

-compatible with any model of the PDP10
-delivered running the system of your choice
-complete disk drive controller and channel,
  for 1 to 8 drives of the 'storage module'
  family -- total storage 80 to 2400 MB
-tape drive controller and channel
-console terminal interface

-built of standard TTL logic, packaged on plug-in cards
-housed in one 19 inch cabinet (except for disk drives)
-requires no special cooling or power wiring
-memory is MOS, with error-correcting code

DISK DRIVES

capacity	price from CalComp	price from Foonly
80 MB	about $7000	$11000
300 MB	about $12000	$16000

These drives (also available from CDC and others) represent the latest removable-media technology. The F2 can support 1 to 8 drives with its standard controller; additional controllers are available for $15000.

TAPE DRIVES

price from Kennedy	price from Foonly
about $3000	$5000

These are 800 or 1600 bpi, 9 track, 45 ips, industry standard format. They are modern, reliable drives of modest performance, intended (in an F2 system) mainly for disk file backup and information exchange with other systems. Foonly Inc. will provide quotations on controllers for higher performance drives on request.

TERMINAL INTERFACE (SERIAL LINE SCANNER)

32 lines	$3000
64 lines	$5000

LINE PRINTER CONTROLLER

-for Data Products or similar line printers $4000

CUSTOM I/O INTERFACES QUOTED ON REQUEST

Software Available on Foonly F1 and F2 Computers

The Foonly, Inc., F1 and F2 computers will (1) emulate a Digital Equipment Corporation PDP-10 (KA processor) and a Bolt, Beranek, and Newman, Inc., pager and (2) run the TENEX operating system. TENEX has the largest repertoire of software for interactive timesharing of any operating system ever written. It can run software written for the TOPS-10 and TOPS-20 operating systems for the PDP-10, as well as programs written expressly for TENEX.

Below is a partial list of the software systems available, by type of software. Most are free; those few requiring a licensing agreement with some third party are so specified.

Operating system

TENEX	PDP TEN EXecutive, Version 1.34, developed by BBN, with extensions by SRI International and USC Information Sciences Institute
PA1050	Simulates the TOPS-10 1050 monitor

User executive

EXEC	User EXECutive, Version 1.54, provides commands for running programs, manipulating files, and monitoring status

Assemblers

MACRO-10	Standard multi-pass MACRO assembler for the PDP-10
FAIL	Fast one-pass assembler from Stanford Artificial Intelligence Laboratory
TNXFAIL	TENEX FAIL
MIDAS	MIT assembler with extensive string and macro capabilities
MACN11	Cross assembler for the PDP-11
PAL11	Cross assembler for the PDP-11
PALX	Cross assembler for the PDP-8
MIX	For Knuth's pseudo-computer, from Stanford

Programming language compilers and interpreters

INTERLISP	INTERactive LISP, developed at BBN and Xerox Palo Alto Research Center
QLISP	QA4 capabilities within INTERLISP, from SRI
MACLSP	MIT Project MAC LISP
UCI-LISP	University of California at Irvine LISP
MLISP	LISP from Stanford which uses an ALGOL-like notation
MLISP2	Extension to MLISP with pattern matching, backtracking, and extensibility
BLISP	LISP from Rutgers University
MICROPLANNER	AI language based on PLANNER, from MIT
STRIPS	STanford Research Institute Problem Solver
SAIL	Stanford Artificial Intelligence Language, a superset of ALGOL
MAINSAIL	MAchine INdependent SAIL
PASCAL	PDP-10 PASCAL
L-10	ALGOL for NLS, from SRI and Tymshare
BASIC	Beginner's All-purpose Symbolic Instruction Code
TBASIC	TENEX BASIC
SNOBOL	StriNg-Oriented SymBOLic Language, from Bell Telephone Laboratories
FASBOL	FASt SNOBOL, compiler for SNOBOL-4
SITBOL	Stevens Institute of Technology SNOBOL [license required]
F40	FORTRAN 4.0
SITGO	Fast load-and-go FORTRAN from Stevens Institute of Technology [license required]
RATFOR	RATional FORTRAN, from Bell Labs
FLECS	FORTRAN extension preprocessor
BLISS	System implementation language, from Carnegie Mellon University

BLISS-10 DEC version of BLISS for the PDP-10
BLISS-11 BLISS for the PDP-11
BCPL ALGOL-like system implementation language, from MIT Lincoln Labs
11BCPL PDP-11 BCPL
PPL Polymorphic Programming Language, extensible, from Harvard
ECL Extensible programming system, including EL-1 language, from Harvard
RPG Report Program Generator
LOGO Programming language for children
SIMULA SIMULAtion language [license required]
POOMAS SIMULA-like language written in BLISS, from CMU
REDUCE Symbolic manipulation of algebraic formulas, from Stanford and University of Utah
RLISP REDUCE LISP
AID Algebraic Interpretive Dialog

Symbolic debuggers
DDT Dynamic Debugging Technique, many versions
RAID Display debugger, from Stanford

Text editors
TECO Text Editor and COrrector, from BBN and MIT
SOS Son Of Stopgap, from Stanford
XOS Variant of SOS
TV Editor for Datamedia TV displays and other terminals, from Stanford
EMACS Display editor developed at MIT
NLS ONLine System, a tree structured document editor from SRI and Tymshare
DNLS Display NLS
XED Line-oriented teXt EDitor
POET Page Oriented EdiTor for TENEX
DED Diminutive EDitor
NETED ARPA NETwork Common EDitor
SPELL SPELLing checker and corrector from Stanford

Document compilers
RUNOFF Program to RUN OFF simply formatted documents
XOFF XGP RUNOFF
POX Program to Output to the XGP, from Stanford
TEX Provides book quality typesetting features, from Stanford
PUB Formats documents for online viewing, line printer, or XGP, from Stanford

Message handlers
SNDMSG SeND MeSsaGe
MSG MeSsaGe reading, filing, and answering
MM Message Munger, from MIT
HERMES BBN proprietary successor to MSG [license required]

ARPAnet software
NCP Network Control Program, from BBN
NETSER TELNET SERver, from BBN
FTPSRV FTP SeRVer, from BBN
TELNET Connecting to foreign host, from BBN
FTP File Transfer Protocol, from BBN
DFTP Datacomputer File Transfer Protocol
RSEXEC Resource Sharing EXECutive, from BBN
TIPCOPY Send data files to a TIP port, from BBN
TIPSTAT Check TIP STATus of ports, from BBN

Binary file manipulators
LOADER BBN modification of DEC LOADER
LINK10 LINKing loader for the PDP-10
LINK11 LINKing loader for the PDP-11

BEDIT Binary file EDITor
FUDGE2 File UpDate GEnerator for relocatable binary programs
BINCOM BINary File COMparison

Other systems software

BATCON BATch job submission and CONtrol
MULTI MULTIfork job manager
DELVER DELete VERsions of unwanted files
DIRED DIRectory EDitor, from Stanford
CIPHER Encrypt files
CCL Command Control Language
DO DO a parameterized list of EXEC commands
RUNFIL Take commands from a file
SORT File SORTer
FAILSAFE Tape copying program
BSYS Backup disk file SYStem on magnetic tape
DUMPER Obsolete alternative to BSYS
SOUP Merge and update source files
CREF Cross-REFerence listing generator
SRCCOM SouRCe File COMparison
EBCDIC EBCDIC ↔ ASCII converter
LCWCASE Converts a text file to LOWer CASE
UPCASE Converts a text file to UPper CASE
FINGER Gives information about current system users, from Stanford
PLAN Provide a personal schedule to FINGER
LD Gives LoaD average and other information about current jobs
GRPSTS Gives machine utilization by pie-slice GRouP STatuS
FLOW FLOWcharts FORTRAN programs
11COPY Converts files from PDP-10 to PDP-11 format
TAPCNV TAPe CoNVerter which converts IBM formatted card image tapes to TENEX format

Statistical analysis programs

BMDP BioMeDical Computing Statistics Package, from UCLA
SPSS Statistical Package for the Social Sciences
STP STATPAK interactive statistical analysis program
DATA-TEXT Interactive statistical analysis for the social sciences
MLAB Modeling LABoratory for curve fitting and solving differential equations

Computer aided design programs

SUDS Stanford University Drawing System
AUG3 Produces a wirelist for AUGat and other PC boards, from CMU
ISPL Instruction Set Processor Language compiler, from CMU
ISPSIM ISP SIMulator, from CMU

Games, experimental research systems, etc.

CHESS Greenblatt CHESS Program from MIT
TECH Carnegie Institute of TECHnology Chess Program
TECH II Alan Baisley's chess program from MIT
CHECKERS CHECKERS game developed at IBM and Stanford
BKG BacKGammon
GO GO
JOTTO From MIT
KALAH From Stanford
ADVENT ADVENTure game from Xerox and Stanford
LIFE Game from Scientific American
MASTER The codebreaking game of MASTERmind
DOCTOR Simulated Rogerian psychiatrist, from MIT
MYCIN Physicians' consultant on bacterial infections, from Stanford
PARRY Simulated paranoiac, from Stanford and UCLA
MAXIM TENEX user quotations

Other user programs

ECAP	Electronic Circuit Analysis Program, from IBM
CMNIGRAPH	Graphics subroutines callable from SAIL or FORTRAN
BUDGET	Prepares a BUDGET
CALENDAR	Keeps a personal CALENDAR of events
LESCAL	Generates custom-made CALendar, from Stanford

SPECIFICATIONS

GENERAL

Each channel of the C1 simulates an IBM Block Mulitplexor Channel for the attachment of IBM-compatible device controllers to the memory and I/O busses of a KL10. Performance is sufficient to allow the simultaneous operation (by three channels) of three 3330 II type disk controllers. The C1 provides the IBM "Bus Extension" option, which allows the connection of high-speed devices, and the "Command Retry" feature, which automatically retries operations experiencing certain errors.

PROGRAMMING

The C1 comes in two models. The programming for the standard model is described in the attached manual. The model C1-K is program compatible with the Systems Concepts SA-10B except for diagnostic functions, which are as in the standard model, and the addition of the MRT command (see next paragraph).

MRT Command

In addition to supporting the normal type of channel programs for disks, which require many commands to read or write each record of data, the C1 provides a special single command which reads or writes one or more records, automatically performing the necessary SEEK and SEARCH functions. This 'MRT' (Multiple Record Transfer) command can reduce by a factor of 20 the amount of channel program which must be compiled to write a page of memory on the disk.

MODULARITY

A C1 consists of a Bus Interface unit and from one to four Channel units, interconnected by cables. Channel units can be moved from one C1 to another, and a spare can be kept on hand for rapid substitution in case of a hardware failure. Channel units cost $5000.

DIAGNOSTIC HARDWARE and SOFTWARE

The PDP-10 can access over 200 bits of registers and control signals from each channel of the C1; the same information is accessible through the indicator panel. The PDP-10 can start, stop, and single-step the channel's control microprocessor, and can test the microprocessor's memory. Because this memory is writeable, the PDP-10 can also set breakpoints in the microprogram or load special diagnostic programs.

A very important diagnostic aid is a facility for the PDP-10 to simulate the actions of a device controller, thus allowing the channel to be thoroughly tested independently of any attached devices; this is a great help in localizing failures. In this mode, the PDP-10 (via the I/O Bus) can read directly all the signals going out of the channel to the controllers, and can directly control the signals which normally come from a controller to the channel.

The diagnostic software supplied with the C1 runs either stand-alone or under timesharing, so if one channel is down, it is not always necessary to take the system down to fix it (if, for instance, all controllers can be switched or re-plugged to the working channel). The diagnostic provides both automatic testing of the channel and interactive (keyboard controlled) features. The latter include DDT-like functions, using the hardware capabilities mentioned above.

DOCUMENTATION

In addition to the programming manual, there is an extensive theory of operations manual designed for efficient use during maintenence. As a further aid to maintenence there are physical layout drawings and two types of wire lists (sorted by signal name and by pin location), with a signal index that locates all the logic drawings relevant to the signal.

Sources are provided for the diagnostic programs, and sources and an assembler are provided for the microprogram.

PHYSICAL

one equipment bay with doors and skins

POWER

117 volts ac, less than 1KW (3-prong twist-lock plug)

INTERFACES

KL10 compatible memory bus, DEC Cajolet type plugs (in and out)
(This interface may be connected directly to KL10 type memories, and provides 22 bits of addressing; it does not utilize other special KL or KI type features.)

KL10 compatible I/O bus, DEC Cajolet type plugs (in and out)
(This interface may be connected to a KL I/O bus, but the channel does not utilize any non-KA features)

IBM compatible bus and tag connectors

PERFORMANCE (assuming average memory access time of ≤ 1000 ns)

Byte rate per channel ≥ 1.25 MB per second.

Total word rate for all channels ≥ ~~8.07 MW~~ 3 MB per second.

Delay between chained commands ≤ 25 microseconds

tion and physical size as a KA10, but with a factor of 2.2 more performance. In scientific applications requiring double precision computation, this performance differential is much greater. Ironically, the TTL/Schottky (TTL/S) series was first available in production quantities about the time of the delivery of the KI10. The KI10 design was started earlier and design options chosen so as to preclude the subsequent advances in speed, power, and density that the TTL/S gave.

The other important logic advances employed in the KI10 were the MSI register file and associative memory packages. The register file provided four sets of accumulators and thus decreased the context switching time. (This probably had a higher psychological than real value but was useful where special devices were operated on a high speed, real time basis.) The environment.

The KL10 provides almost a factor of five performance improvement over the KA10 for programs using the basic instruction set. An even larger performance improvement is realized for Cobol or extended precision scientific programs. The organization and much of the base work for the KL10 was done by Dave Poole, Phil Petit, John Holloway, and Jack Wright at the Stanford Artificial Intelligence Laboratory.

The KL10 is microprogrammed using a memory based on the one Kbit bipolar RAM. A cache memory is also constructed from the one Kbit chips. The KL10 is implemented in the Emitter Coupled Logic (ECL) 10K series rather than the TTL/Schottky of the original Stanford design. It was felt that the ECL speed advantage with 3 nsec. gate delay vs 7 nsec. for Schottky was worth the extra design effort especially since the

Table VII. Implementations for DECsystem 10 hardware.

Processor	PDP-n	KA10	KI10	KL10	
Design start	2/63	1/66	12/69	1/72	
First ship	6/64	9/67	5/72	6/75	
Logic	Germanium, Silicon transistors	Discrete Silicon transistors and diode	TTL/H (MSI) Registers; assoc. memory	ECL 10K; Fast, 1 Kbit memories	
MIPS(avg.)	0.25	0.38	0.72	1.3	
Packaging (slice of Pc)	1-bit of AR, MB, MQ, AD:38 transistors, 2-sided PC etch; 2, 18-pin & 2-22-pin conn. (11" × 9" boards)	implemented in R, S, W-series flip chip (discrete) modules (5½ × 5¼ boards)	implemented in R, S, W, M-series flip chip (discrete – MSI) modules 5½ × 5¼ boards	6-bits of AR, ARX, MQ, BR, BRX, AD, ADX;70 MSI ECL per module; 216 pin connector; 15" × 15" boards)	
Pc. size	2 bay	2 bays	2+ bays	½ bay (including internal channels)	
Pc. price	$120K	$150K	$200K	$250K	
Control Design	async. & subroutine logic	same as PDP-6	clocked sync.	KL20 is clocked sync.; microprogrammed	
Module Size	large modules	small modules wire wrap	same	large modules (16 Kword core memory module)	
Registers	16	16	4 × 16	8 × 16	
I/O calls	prog. interrupts UUO traps;	same	vectored interrupts		
I/O transmission	I/O & Memory Bus	added channels		integrated controller for MASSBUS; I/O via PDP-11 computers	
Memory Management	18-bit phys. addr. protection & relocation regs.	2 protection & relocation regs. for shared program segments	22-bit phys. addr: paged using 32 word associative memory	22-bit phys addr. paged, using associative memory via cache	
ISP	see Table III (integers, floating)	conversion to assist d.p. float	hardware d.p. float	string & conversion for d.p. integers	
Parallelism		simpler (faster) data path	instruction look-ahead (1 word) fetch	instruction look ahead; 2 Kword cache memory	
Fabrication	(100) large modules	Gardner-Denver automatic wire wrap for backpanel interconnection	semiautomatic wirewrap for twisted pair	large (hex) modules with many pins; low cost minis front end	(KL20) integrating Pc and Mp together – eliminating Memory Bus⇒ high density core memory modules
Consequences	served as PDP-10 production prototype	buildable in production	more performance (scientific & real time); and paging for operating systems	more performance via cache; microprogramming for better COBOL ISP; I/o computers	lower cost

Send to Gordon Ball

Gordon – surprised to hear Ned
Natt you didn't know we have a
large system mfg organization – This should
make your day! Charlie Bridstrup is Ulf's
mfg team and they are even collocated! You'll
like Charlie, he is an effective manager and has
the right motivations. pages for flavor.

LARGE SYSTEMS MANUFACTURING

CHARTER

See first few
Dave Knell
9/12/80

CHARTER STATEMENTS

* DOMAIN
* RATIONALE
* RESPONSIBILITIES/ACTIVITIES

FIVE YEAR BUSINESS PLAN

ORGANIZATIONAL RESPONSIBILITIES

REV "A"
8/08/80

LARGE SYSTEMS MANUFACTURING

O R G A N I Z A T I O N

LARGE SYSTEMS MANUFACTURING
RESPONSIBILITIES/ACTIVITIES

1. TO COMMIT FOR ALL OF DIGITAL MANUFACTURING TO ALL OF DIGITAL FOR OUR DOMAIN.
2. TO MANAGE ACROSS THE INTERFACES WITH PRODUCT MARKET GROUPS, DESIGN ENGINEERING, FIELD SERVICE, AND MANUFACTURING TO ACHIEVE THE ALLOCATED REVENUE STREAM FOR LARGE SYSTEMS.

Is this our explicit view PGMs role?

3. TO ASSURE THAT THE PROPER PORTION OF MANUFACTURING'S RESOURCES SUFFICIENT TO ACHIEVE THE ALLOCATED REVENUE ARE COMMITTED TO THIS BUSINESS REGARDLESS OF WHO "OWNS" THE RESOURCES.
4. TO MANAGE AND ASSURE THAT THE REQUIRED MANUFACTURING RESOURCES WILL BE AVAILABLE TO SUPPORT THE PRODUCT GROUPS LONG RANGE PLAN RELATIVE TO LARGE SYSTEMS.
5. TO MANAGE AND COMMUNICATE THE SHORT TERM PERFORMANCE OF LARGE SYSTEMS BY PRODUCT GROUPS.
6. TO REPRESENT PRODUCT AVAILABILITY, QUALITY, COST, CONTRIBUTION, FLEXIBILITY, ETC. TO THE PRODUCT GROUPS.
7. TO ASSURE THAT ENGINEERING DESIGN PRODUCTS THAT PERMIT MANUFACTURING TO ACHIEVE THE NEEDED COST AND QUALITY (I.E., MERGEABILITY) GOALS, DELIVER DESIGNS ON TIME AND DOCUMENTED SO AS TO MINIMIZE INTRODUCTION COST.
8. TO REPRESENT ADVANCED MANUFACTURING REQUIREMENTS TO ENGINEERING.
9. TO ASSURE THAT MANUFACTURING COMMITS TO THE SYSTEM QUALITY LEVELS AND COST REDUCTIONS VITAL IN OUR MARKETPLACE.

LARGE SYSTEMS MANUFACTURING

DOMAIN

THIS CHARTER ENCOMPASSES A MANUFACTURING MANAGEMENT GROUP RESPONSIBLE FOR THE TOTAL MANUFACTURING PROCESS OF LARGE SYSTEMS WITHIN DIGITAL.

THE CORE DEFINITION FOR LARGE SYSTEMS WILL BE THAT WHICH IS SPECIFIED BY SYSTEMS PRODUCT MANAGEMENT IN ITS' STRATEGIES. TODAY, THE DOMAIN OF THIS GROUP, FROM AN ENGINEERING AND MANUFACTURING VIEWPOINT, IS:

Call products designed by us.

1. ALL 36 BIT SYSTEMS;
2. 32 BIT SYSTEMS OF ENTRY LEVEL PRICE BETWEEN $100-$250K SPECIFIED ON A SYSTEMS PROJECT BASIS (VENUS) INCLUDING ALL ASSOCIATED HARDWARE AND SOFTWARE.

THE DOMAIN DEFINED IN THIS WAY ACKNOWLEDGES THAT THERE WILL BE OVERLAP AND COEXISTENCE AMONG THE PRODUCT FAMILIES IN THE NEXT FEW YEARS. AS OLDER PRODUCTS PHASE OUT, AND MIGRATION STRATEGIES BECOME CLEAR, THE CORPORATE STRATEGY DEFINITION BECOMES DOMINANT. A PROJECT-BY-PROJECT BASIS ENSURES CLEAR RESPONSIBILITIES IN THE NEAR TERM.

LARGE SYSTEMS MANUFACTURING

RATIONALE

REDUCTION IN MANUFACTURING COMPLEXITY BY MAINTAINING VERTICAL INTEGRATION WILL RESULT IN INVENTORY REDUCTIONS, IMPROVED FLEXIBILITY AND LOWER DISTRIBUTION AND INDIRECT LABOR COSTS.

POSITIVELY IMPACT EMPLOYEE SATISFACTION BY GIVING PEOPLE AN OPPORTUNITY TO UNDERSTAND HOW THEY SUPPORT THE WHOLE, AND BY IMPROVING CONTROL OF THE BUSINESS.

MAINTAINING AND ENHANCING THE EXISTING ENGINEERING LINKS IN LARGE SYSTEMS MANUFACTURING BY HAVING AN ORGANIZATION ARRANGED BY SIZE AND TYPE AND STRUCTURED IN SUCH A WAY AS TO TAKE ADVANTAGE OF THE LEARNING THAT ALREADY EXISTS IN THE PROCESSOR FACTORY.

LARGE SYSTEMS MANUFACTURING

TECHNOLOGY MANAGEMENT

THE ORGANIZATION AND RESPONSIBILITIES DESCRIBED BELOW PROVIDES THE NECESSARY ENGINEERING LINKS TO THE LARGE SYSTEMS MANUFACTURING WORLD AND FOCUSES ON THE SPECIFIC DISCIPLINES AND COMMITMENTS TO BE MANAGED.

NEW PRODUCTS INTRO/PROJECT MANAGEMENT CHARTERED WITH THE RESPONSIBILITY FOR SUCCESSFUL NEW PRODUCT INTRODUCTION PROJECT MANAGEMENT. THIS GROUP TO BE ORGANIZED AROUND THE 32 AND 36 BIT SYSTEM ARCHITECTURE WITH SPECIALIZATION IN LARGE SYSTEMS MANUFACTURING PERIPHERALS AND PROGRAM MANAGEMENT. THEY WILL ALSO BE THE PRIMARY LINK WITH THE LARGE SYSTEMS HARDWARE DESIGN ENGINEERING FUNCTION.

PRODUCT ENGINEERING MANAGEMENT WILL HAVE "WOMB TO TOMB" OWNERSHIP OF ALL PRODUCTS INCORPORATED INTO LARGE SYSTEMS MANUFACTURING. THE PRIMARY VOLUME MANUFACTURING/ENGINEERING INTERFACE, THE GROUP WILL BE STRUCTURED AROUND MASS STORAGE DEVICES, TERMINALS, MEMORIES, PCB, FAB, LSI, ETC., THE ACCOUNTABILITY FOR PHASE IN, MAINTENANCE OF SUPPLY AND PHASE OUT AS WELL AS TECHNICAL DIRECTION TO THE PLANTS RESIDES HERE.

PROCESS ENGINEERING MANAGEMENT WILL DEVELOP, DOCUMENT AND INSTITUTE THE LARGE SYSTEMS MANUFACTURING PROCESSES REGARDLESS OF WHERE BUILT TO ENSURE REPRODUCIBILITY. CHARTERED WITH EFFECTING COST REDUCTION/AVOIDANCE IN THE MANUFACTURING PROCESSES THEY WILL INFLUENCE DECISIONS AROUND PRODUCT DESIGN AND WILL HAVE ONWERSHIP FOR ASSEMBLY AND TEST METHODS. THEIR INVOLVEMENT IS VIEWED AS A KEY ELEMENT IN EVOLVING LARGE SYSTEMS MANUFACTURING'S MERGE PROCESSES.

LARGE SYSTEMS MANUFACTURING

TECHNOLOGY MANAGEMENT

THE SOFTWARE ENGINEERING GROUP WILL ENSURE THAT THE LARGE SYSTEMS MANUFACTURING GROUP MANAGES THE INTRODUCTION AND COORDINATION OF SOFTWARE RELEASES INTO PRODUCTION. THE PRIMARY INTERFACE TO THE SOFTWARE DEVELOPMENT ENGINEERING GROUP, THEY COMPLIMENT THE NEW PRODUCTS INTRODUCTION ORGANIZATION AND ARE A RESOURCE TO THE PRODUCT ENGINEERING AND PROCESS ENGINEERING GROUPS.

LARGE SYSTEMS MANUFACTURING

LARGE SYSTEMS MANUFACTURING MANAGER — (Charlie Bradshaw)

* MARLBORO PLANT - LARGE SYSTEMS VOLUME
* FRANKLIN PLANT - LARGE SYSTEMS FA&T, DISTRIBUTION
* LARGE SYSTEMS MANUFACTURING TECHNOLOGY MANAGER (M.E.)
  - NEW PRODUCTS
    32, 36 BIT - PROGRAM MANAGEMENT
  - PRODUCT ENGINEERING MANAGER
    MASS STORAGE
    LSI
    TERMINALS, ETC., ETC.
  - PROCESS ENGINEERING MANAGER
    ASSEMBLY, TEST METHODS
    PACKAGING, COST REDUCTION
  - SOFTWARE ENGINEERING MANAGER
    TOPS 10, 20, VMS
  - FUNCTIONAL MANAGER TO PLANTS
* LARGE SYSTEMS MANUFACTURING RESOURCE MANAGER
* LARGE SYSTEMS MANUFACTURING PLANNING
  - "THE PLAN", MARKETING INTERFACE, LARGE SYSTEMS MANUFACTURING BLUE BOOK, CAPACITY, MANAGEMENT PROCESS
* LARGE SYSTEMS MANUFACTURING FINANCE
  - MARLBORO FINANCE
  - FRANKLIN FINANCE
  - LARGE SYSTEMS MANUFACTURING MIS
* LARGE SYSTEMS MANUFACTURING PERSONNEL
  - MARLBORO PERSONNEL
  - FRANKLIN PERSONNEL
* LARGE SYSTEMS MANUFACTURING QUALITY ASSURANCE
  - CUSTOMER INTERFACE
  - FIELD SERVICE INTERFACE
  - PRODUCT LINE INTERFACE
  - FUNCTIONAL MANAGER TO PLANTS