

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	HARDWARE OVERVIEW	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55	APPENDIX A	55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12	HARDWARE OVERVIEW	12
13		13
14		14
15	INTRODUCTION	15
16	-----	16
17		17
18	This section is an overview of Burroughs large systems	18
19	hardware. Because of hardware differences among large systems	19
20	the hardware overview is divided into five sections as	20
21	follows:	21
22		22
23	B7800 hardware	23
24		24
25	B7900 hardware	25
26		26
27	B5900 hardware	27
28		28
29	B6900 hardware	29
30		30
31	B5900/B6900 I/O Overview	31
32		32
33		33
34	B7800 HARDWARE OVERVIEW	34
35	-----	35
36		36
37		37
38	GENERAL	38
39	-----	39
40		40
41	The Burroughs B7800 information processing system is a large	41
42	scale, multiprogramming and multiprocessing computing system.	42
43	The hardware is controlled by the Burroughs Master Control	43
44	Program (MCP) and can be tailored to the processing needs of a	44
45	user by arranging Central Processor Modules (CPM),	45
46	Input/Output Modules (IOM), and Memory Control Modules (MCM)	46
47	on an electronic grid, or exchange (figure 1-1). The MCP can	47
48	be changed dramatically by merely setting or resetting run	48
49	time options with simple operator input commands. The system	49
50	may be balanced by other operator instructions (and	50
51	dynamically by the MCP) that control the interaction of the	51
52	independently operating CPM's, IOM's, and MCM's. The	52
53	throughput of the system as a whole is maximized, but the	53
54	performance of no single element of the system is maximized to	54
55	the neglect or detriment of others.	55
56		56
57	The key to the efficient balanced use of the system is the	57
58	Burroughs Master Control Program, a unique executive software	58
59		59
60		60

1		1
2		2
3		3
4	operating system that automatically makes optimum use of all	4
5	system resources. It is this operating system that makes	5
6	multiprogramming and multiprocessing both functional and	6
7	practical by dynamically controlling system resources and	7
8	scheduling jobs in the multiprogramming mix. The MCP allocates	8
9	system resources to meet the needs of the programs introduced	9
10	into the computer. It continually and automatically reassigns	10
11	resources, starts jobs, and monitors their performance.	11
12		12
13	Further implications of the modularity and flexibility of the	13
14	system are its expandability (a capacity to add hardware	14
15	modules without reprogramming) and its increased reliability	15
16	(and, thus, increased availability to the user). This	16
17	reliability is achieved by the use of failsoft techniques that	17
18	(error detection and correction, redundancy of power supplies)	18
19	exclude faulty modules from the system and permit processing	19
20	to continue (without reprogramming) with a temporarily reduced	20
21	configuration.	21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

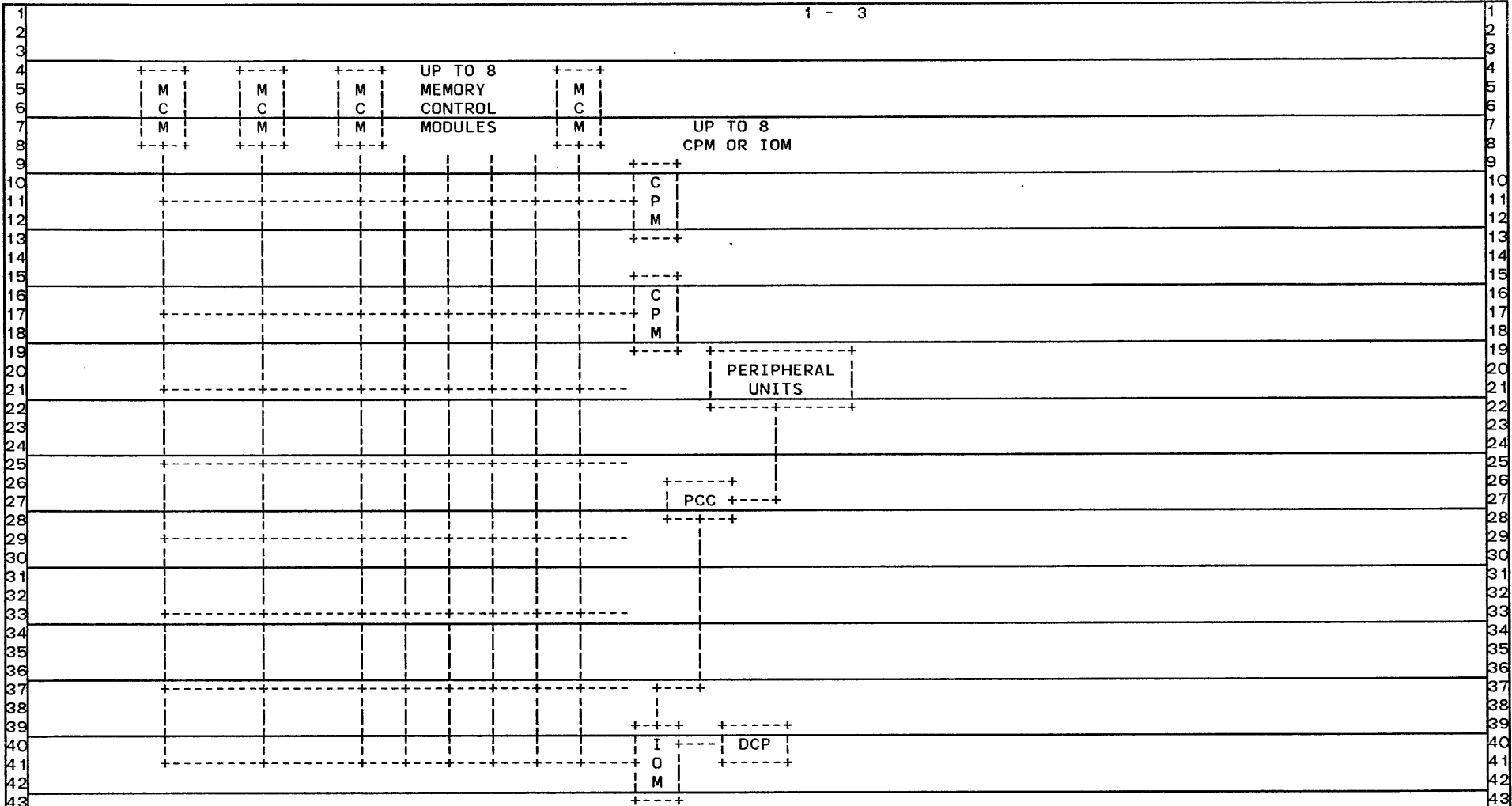


Figure 1-1. B7800 Exchange

SYSTEM CONFIGURATION

Physically, the components of the system fall into three categories, as follows:

1. Central components of the system: Central Processor Module (CPM), Input/Output Module (IOM), Memory Control Module (MCM), Maintenance Diagnostic Processor (MDP), and operator's console.
2. Standard Burroughs cabinets that contain peripheral controls and exchanges, Data Communications Processor (DCP), and AC power supplies.
3. Peripheral devices that are joined to the central system by means of peripheral controls and exchanges. Remote devices that are joined to the central system by means of line adapters and the data communications processor.

The arrangement of these components into a system and the size of the system depend on the application and workload of the user.

HARDWARE REVIEW

Before looking at the MCP it is important for the student to have a good understanding of the hardware. Each module on the system will be discussed. The discussion will start with a general overview of the system and follow with detailed information on each module.

Figure 1-2 is a diagram of a system which is composed of Central Processor Modules (CPM), Input/Output Modules (IOM), Memory Control Modules (MCM) and Data Communication Processors (DCP).

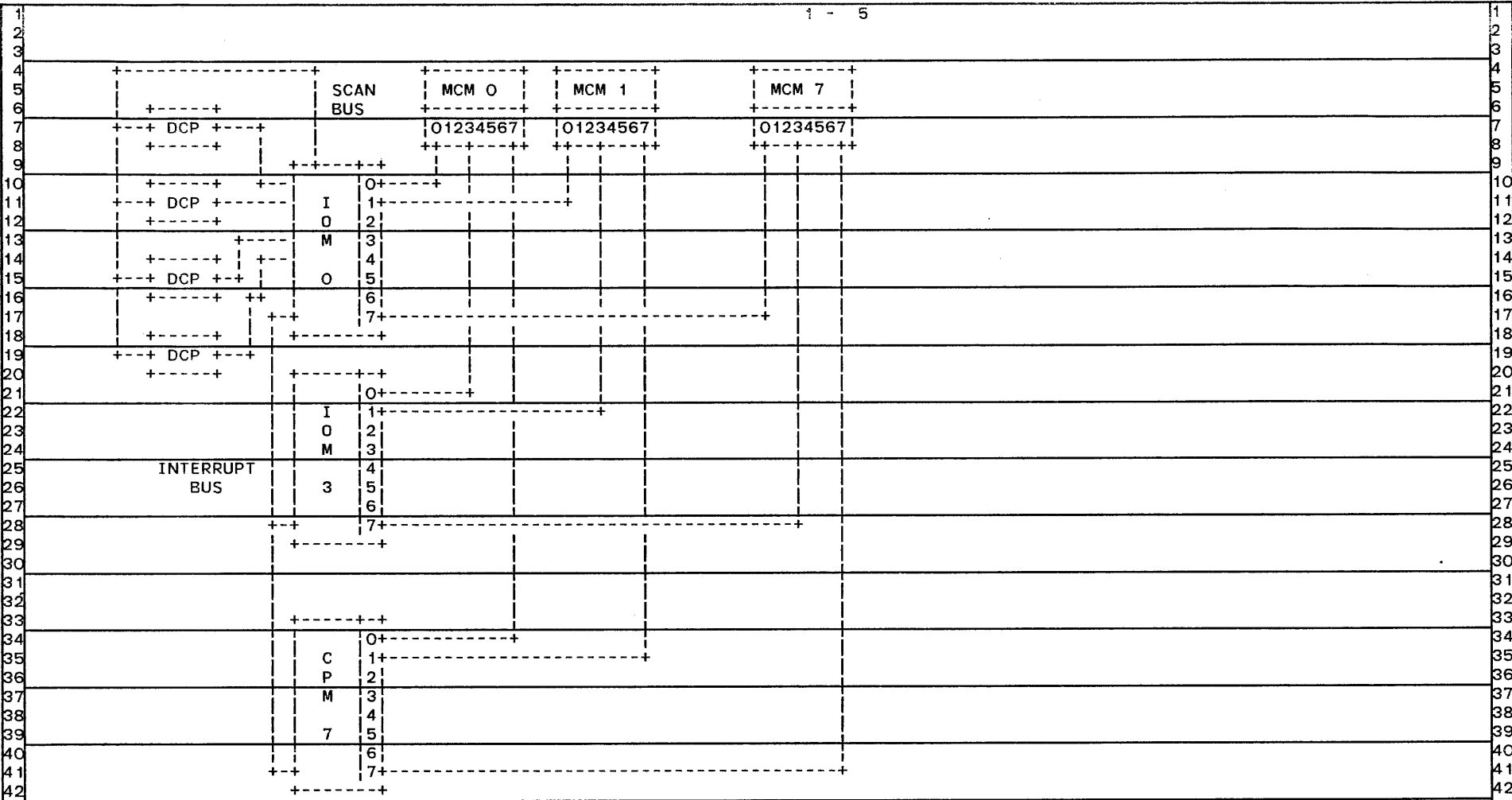


Figure 1-2. B7800 SYSTEM

The following general comments can be made about the system.

1. Each module on the system is independently powered.
2. All CPM's and IOM's are connected with an interrupt bus.
3. There is a scan bus that connects an IOM and the DCP's on that IOM.
4. The DCP's on an IOM will use the IOM's memory access logic.
5. Each CPM has a time of day clock.
6. There is one master clock for the system. It has it's own power supply.

A general discussion of each system component follows. This discussion is intended to give the student an overview of the system.

#### MEMORY

All memory will do single bit error correction and multi-bit error detection. Access to memory is phased (interleaved). All requestors will take advantage of memory phasing. The IOM will make four word requests to an MCM. CPM's will make four or eight word requests to an MCM. A planar memory MCM will return at most four words. An IC memory MCM will return at most eight words. Thus, the number of words returned for a given request is based on the type of memory.

#### CPM

The CPM has a 2K word data buffer and a 2K word code buffer. There is a 32 word store queue which will buffer data before it is sent to memory. Thus, memory accesses can be grouped into multi-word stores. Also repeated stores to an address can be eliminated, only the newest data will be written to memory. The local buffers in the CPM will attempt to keep memory requests to a minimum and will try to have code and data ready for the execution unit. Thus, the CPM can overlap fetching of data with execution of instructions.

IOM

An IOM has up to 28 fixed channels. Each channel has its own (2 four word) buffers. Data is transferred to and from the IOM in 4 word groups. The IOM does it's own path checking and can handle a queue of I/O's, not just one at a time. I/O results are reported in a queue. Thus, an IOM need not interrupt a CPM for an I/O finish. The IOM will handle conditional seek logic for disk packs. The IOM uses MCP generated I/O queues and an I/O map.

MDP

Maintenance Diagonistic Processing (MDP) is used for testing mainframe modules (CPM, IOM and MCM), testing logic cards and PROM programming. MDP can be run online with a CPM or with the Maintenance Processor (MP). The MP is a B800 system. Thus, maintenance operations can be run by a CPM or the B800. The MP can be connected to a datacomm line so remote diagnostics can be done.

MAXIMUM CONFIGURATION

- 8 Processors (IOM and CPM)
- 8 MCM's (2.5 million words usable)
- 255 Units
- 8 DCP's



The following part of this review will give a more detailed discussion of the MCM, CPM and IOM. In addition, the MCP interface with the IOM will be discussed.

#### MEMORY

-----

Memory Control Modules (MCM's) form the interface between a requestor (IOM and CPM) and memory. Each requestor is connected to each MCM. A system can be configured with up to 8 MCM's. The MCM is connected to memory storage cabinets which contain actual memory storage (planar or IC memory).

Figure 1-3 shows two MCMs with planar memory. Each MCM can have up to 2 Memory Storage Cabinets (MSC). Each MSC has 2 Memory Storage Units (MSU). A MSU has 65K words. Planar memory will phase at most 4 words. Thus, the maximum number of words that can be transferred in one operation is 4 words. Following is the amount of time required to transfer 4 words using planar memory.

1 MSC/MCM 3.75 microseconds (2 word phased)

2 MSC/MCM 2.125 microseconds (4 word phased)

Figure 1-4 is a diagram of a MCM with IC memory. Each MCM can have up to 2 Memory Storage Cabinets (MSC). A MSC can have up to four Memory Storage Units (MSU). Each MSU has 131K words. Memory is phased at the MSU level and will phase at most 8 words. Thus, the maximum number of words that can be transferred in one operation is 8 words. Following is the amount of time required to transfer the number of words indicated using IC memory.

1.87 microseconds for 4 words

2.37 microseconds for 8 words

There are two models of MCM's (model II and model III). A model II MCM can control 256K words of memory. A model III MCM can control 1 million words of memory. This configuration (1 MCM with 1 million words) would provide only one path to memory and could cause performance problems. Better performance could be achieved by using two model III MCM's with each MCM controlling 500K words.

A requestor can address 1 million words. Therefore a monolithic system can use at most 1 million words. More memory can be configured but would not be usable. Any additional memory could be used if the system was split into two systems. A system running tightly coupled can use up to 2 million words (2.5 million words using model III MCM's). This tightly coupled configuration (2.5 million words) would require 5 model III MCM's, 4 CPM's and 4 IOM's.

Words stored in memory are 60 bits as follows:

8 bits for error correction and detection.  
This allows single bit error correction and multi-bit error detection.

52 data bits are transferred to and from requestor.  
48 data  
3 tag  
1 parity

The MCM is responsible for:

Buffering multi word transfers to and from a requestor (memory phasing).

Locking out MCM's to requestors. This is used when the system is split or running tightly coupled. Requestors are locked out by setting requestor inhibit switches on the MCM. These switches can be set by the operator or the MCP.

Controlling requestor priority.

Doing error correction and detection.

Broadcasting which addresses the MCM controls. The address range switches (UPPER and LOWER) specify the memory addresses a MCM spans. These switches can be set by the operator or MCP. The register contains the six most significant bits of the address that is controlled by this MCM.

Maintaining the MSU status register which specifies which MSU's are available. These switches can be set by the operator or MCP.

One should note a requestor (not the MCM) must keep track of how many words are received from a memory request. If all words that were requested were not received a request for the remaining words must be made. For example, if a request is made for 8 words from an MCM that is only 4 way phased, only 4 words are returned. The requestor must change the word count and memory address in the MCM control word and make the request to the MCM again.

As an example of how these switches are set, assume a system is configured as follows: (see figure 1-3)

2 MCM (MCM 1 and MCM 4) 4 MSU each (planar MSU)  
1 IOM (IOM 1)  
1 CPM (CPM 5)

The switches should be set as follows:

MCM	LOWER	UPPER	INHIBIT	MSU STATUS
1	00000	001111	11011101	1111
4	010000	011111	11011101	1111

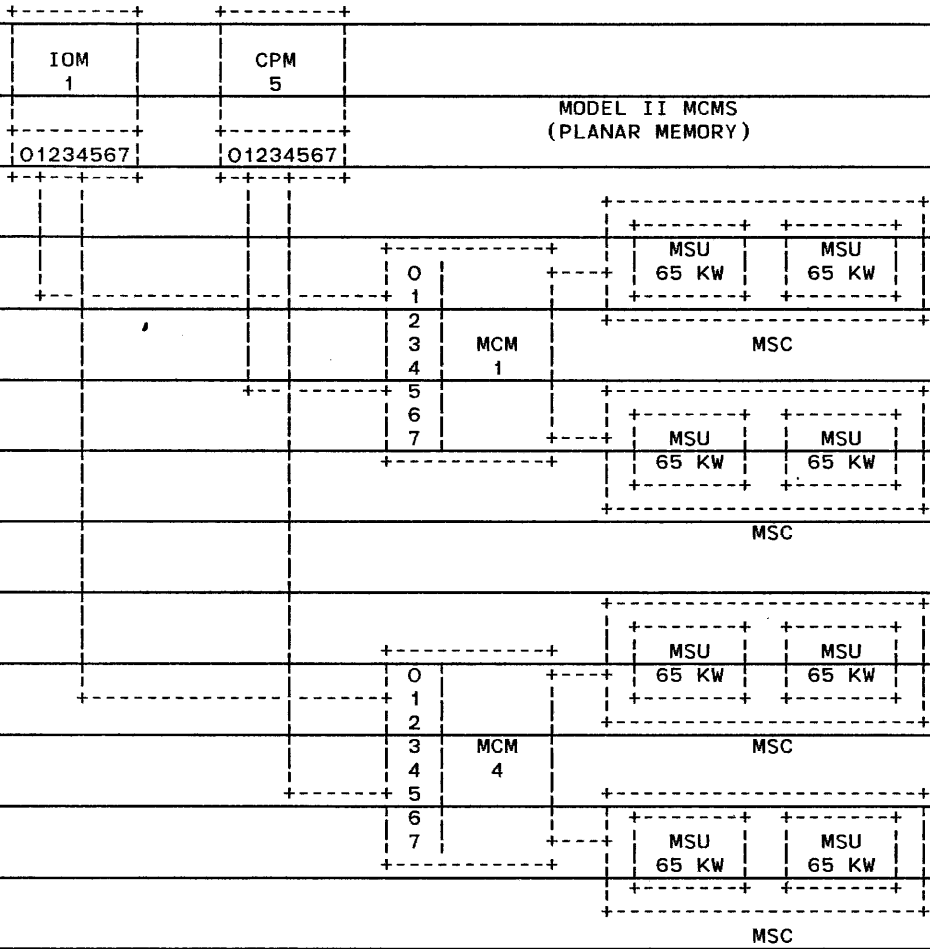


Figure 1-3. MCM Configuration and Planar Memory

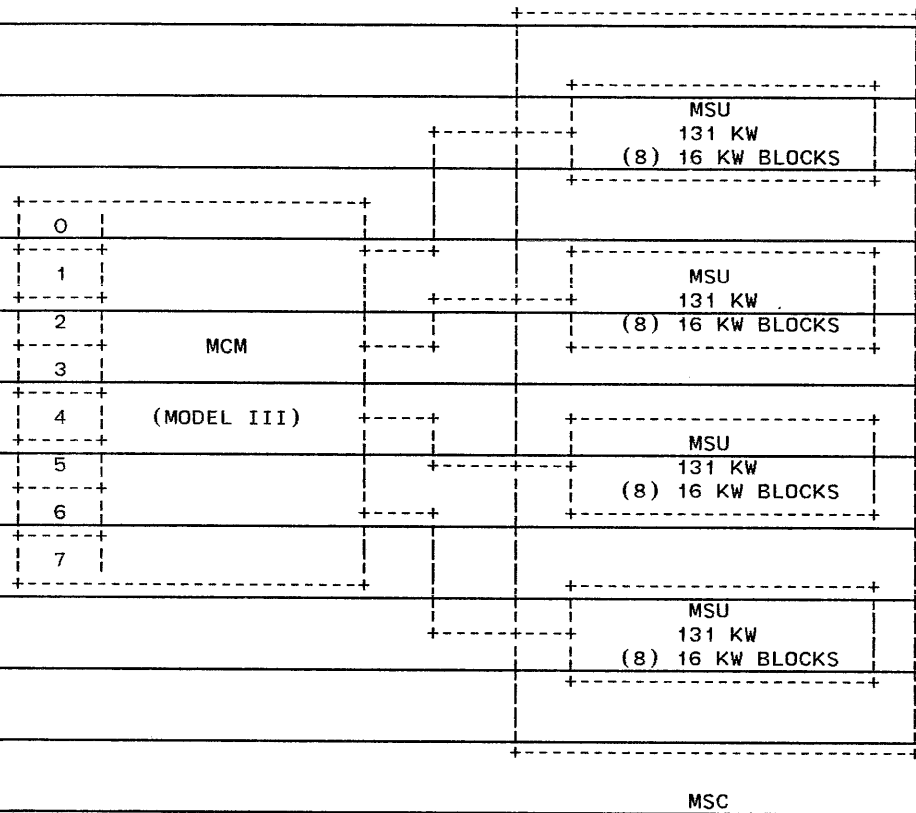


Figure 1-4. MODEL III MCM Block Diagram (IC Memory)

## CPM

---

The CPM is composed of several independent asynchronous units. It is this independence that allows data fetches to be overlapped with instruction execution. The basic functions of these units in the CPM are described in the following paragraphs (see figure 1-5).

## MEMORY ACCESS UNIT (MAU):

The memory access unit provides the memory interface for the CPM. The MAU is capable of doing simultaneous fetches and stores to memory as long as the addresses are not in the same MCM.

## PROGRAM CONTROL UNIT (PCU):

The Program Control Unit (PCU) interprets object code and builds micro operators for the execution unit. These micro operators are placed in a queue for the execution unit. The PCU will pre process code doing operations such as taking unconditional branches, generating absolute addresses from relative addresses and operator concatenation. That is, join more than one instruction into a single micro op.

In addition, the PCU is responsible for allocation of registers in the Central Data Buffers (CDB).

## EXECUTION UNIT (EU):

The Execution Unit (EU) performs the actual operations. These operators come from a queue built by the PCU.

The EU includes logic for short arithmetics, add operations less than 20 bits and multiply operations with results less than 16 bits.

## DATA REFERENCE UNIT (DRU):

The Data Reference Unit (DRU) fetches data from associative memory or main memory so it can be used by the EU.

## STORE QUEUE (SQ):

All stores to main memory go thru the store queue. The store queue will try to group adjacent words into a multi word write. In addition, it will test for repeated stores to the same address. This will help cut down on memory traffic.

LOCAL MEMORY (ASSOCIATIVE MEMORY):

The program buffer will hold up to 2K words of object code in the CPM. The data buffer will hold up to 2K words of data and control information.

CENTRAL DATA BUFFER (CDB):

The central data buffer (64 words) acts as storage and an exchange between CPM units.

INTERRUPT BUS:

Module to module interrupt line used to interrupt other CPM's and IOM's.

The CPM will do internal residue, parity and continuity checking. If errors are detected a CPM fail word is generated and an interrupt occurs. The CPM is also capable of instruction retry.

The CPM must read a register (62) every 8 to 16 seconds. If this is not done an EGG TIMER interrupt is generated. This will prevent the CPM from looping in control state.

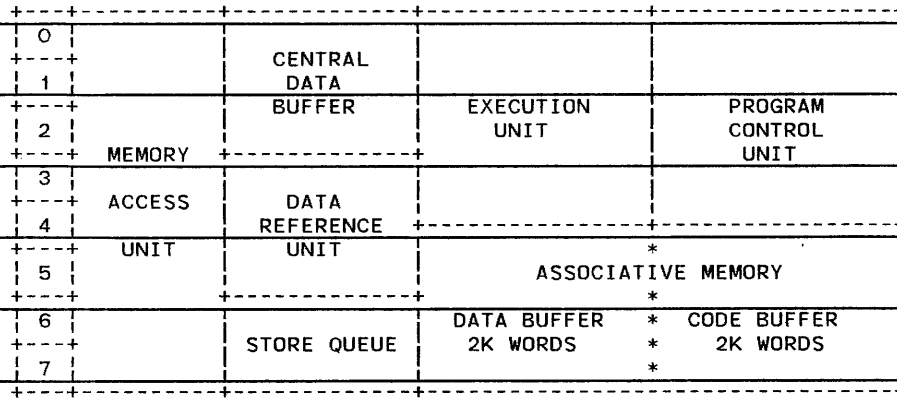


Figure 1-5. CPM Block Diagram



The following discussion on associative memory is based on the B7800 CPM.

Figure 1-6 is a diagram of how the code buffer is set up. Three data structures are used. These structures are the address array, priority array and the actual code buffers.

To access a word in this part of associative memory bits [7:6] of the 20 bit address are used to get the block number. This block number is used to index the address array. By comparing bits [19:12] of the address with the eight groups in the block you can determine if the word you are looking for is present.

Assuming the address is present, bit [0:1] selects the EVEN or ODD buffers. The group number (determined in the address array) is used to choose a 128 word buffer. Bits [7:7] of the address are used to index into the 128 words.

The priority array has a three bit entry for each of the 64 blocks. It contains the next group number to be overlayed. When that group is overlayed the array value is incremented by one. When seven (111) is reached it will go back to zero (000).

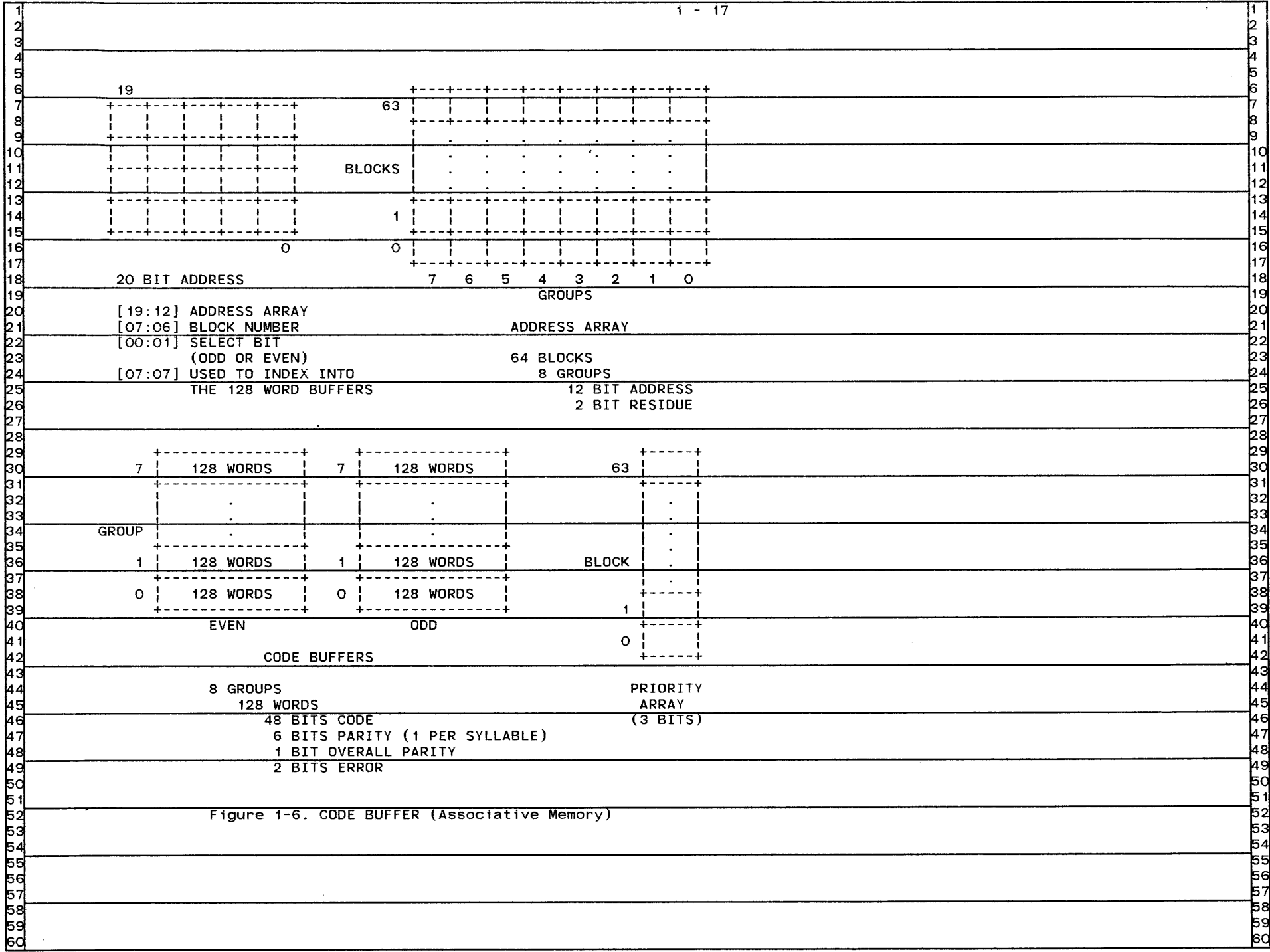


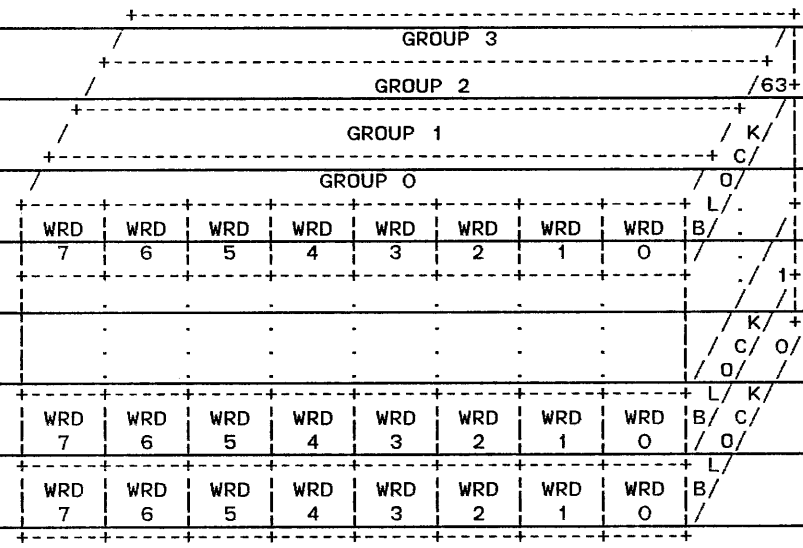
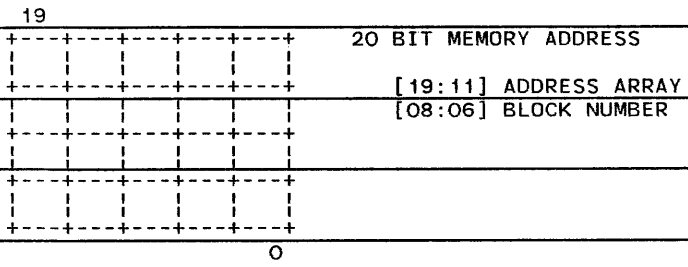
Figure 1-7 is a diagram of how the data buffer is set up. Again three data structures are used. These include the address array, the priority array and the data array.

To access a word in this part of associative memory bits [8:6] of the 20 bit address are used to get the block number. As with the code buffer, this block is indexed in the address array. Bits [19:11] of the address are compared with the four groups to see if the address is present.

Assuming the address is present, the block number and the group number are used to locate the proper eight word group in the data array. Bits [2:3] of the address are used to get the proper word from the eight word group.

The priority array has 64 entries, 5 bits each. Two bits indicate the 'oldest' group in the block. Two bits indicate the 'second oldest' group in the block. The fifth bit determines the order of the other two groups. The fifth bit is set if the 'newest' group (number) has a greater value than the 'second newest' group (number).

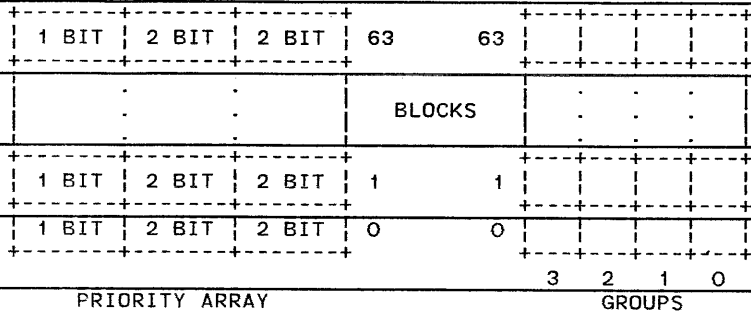
The priority array is updated whenever new data is written to the data buffer and also whenever a word local to the buffer is accessed. If new data is to be loaded, the 'oldest' group entry is overlayed. If a word already local in the buffer is accessed it is moved to the 'newest' position in the priority array.



DATA ARRAY WORDS

64 BLOCKS  
4 GROUPS  
8 WORDS

Figure 1-7. DATA BUFFER (Associative Memory) (1 of 2)



64 ENTRIES	ADDRESS ARRAY
2 BITS = OLDEST ENTRY	64 BLOCKS
2 BITS = 2ND OLDEST	4 GROUPS
1 BIT = SET IF	'NEWEST' ENTRY > 11 BIT ADDRESS
	'SECOND NEWEST' 2 BIT RESIDUE
	ENTRY

Figure 1-7. DATA BUFFER (Associative Memory) (2 of 2)

## IOM

---

The IOM serves as a buffer and control unit for all I/O operations. The IOM is composed of several asynchronous units. These units can be seen in figure 1-8 and are described in the following paragraphs.

## MEMORY INTERFACE UNIT

The memory interface unit provides memory interface for the IOM. It buffers and controls all IOM memory access. The memory interface unit controls access priority. The priority is:

DSU - Data Service Unit

PCI

DFIA

DFIB

Lowest number channel in each unit  
has highest priority

Data comm interface unit

Translator

## TRANSLATOR UNIT

The translator is the control unit of the IOM. It fetches commands from memory and starts I/O operations. The translator sends interrupts for I/O finish and ODT status change. It holds 20 bit addresses for Home Address (HA), Unit Table (UT), Queue Head (QH) and Status Queue (SQ). In addition, the translator contains channel status information, peripheral status and controls conditional I/O operations.

## DATACOMM INTERFACE UNIT

The datacomm interface unit transfers information between the IOM and DCP. Thus, it provides the memory interface for the DCP.

## SCAN INTERFACE UNIT

The scan interface is for scan type operations between the IOM and DCP's connected to the IOM.

## DATA SERVICE UNIT

The data service unit is made up of peripheral interface unit, Disk File Interface "A" (DFIA), and Disk File Interface "B" (DFIB). It forms the buffer between memory and peripherals. There are two (2) four word buffers on each channel which allows the IOM to take advantage of 4

word phased access to memory.

CONNECTED TO DATA SERVICE UNIT

A Peripheral Control (PC) bus is used to connect the IOM to the Peripheral Control Cabinet (PCC). The PCC contains peripheral controls. Data is transferred to the IOM in one or two byte groups. The PCC is independently powered or powered by B6700 style AC mods.

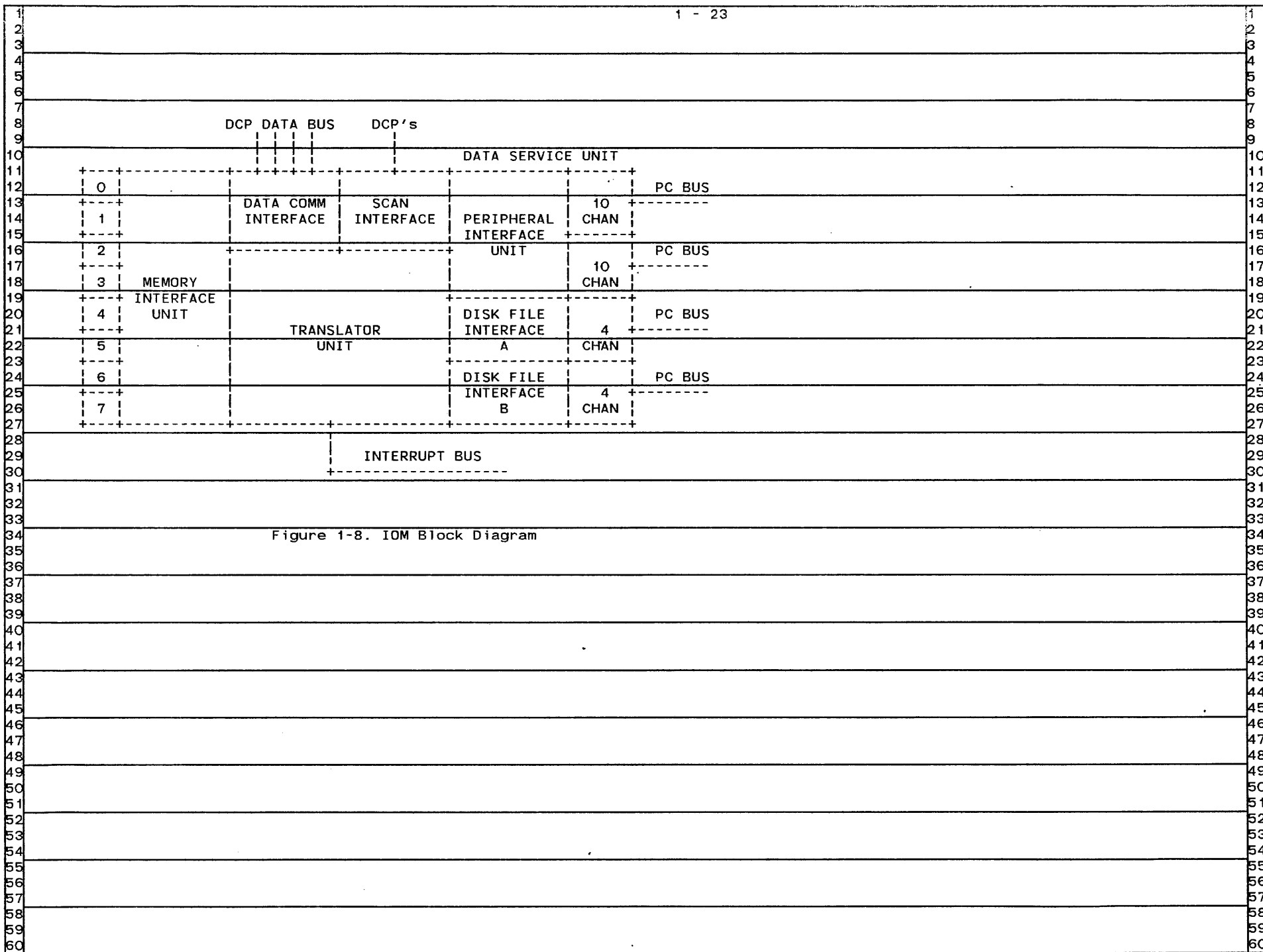


Figure 1-8. IOM Block Diagram



The IOM and MCP share several data structures. These structures are located in main memory and are shown in figure 1-9. The following paragraphs describe these structures.

#### I/O QUEUE:

All I/O'S for a unit are linked to each other. The first I/O Control Block (IOCB) is pointed to by the I/O queue head word. The last IOCB is pointed to by the I/O queue tail word. The linkage from IOCB to IOCB is thru a link word in the IOCB. All IOM's will use the same I/O queue.

A queue for a unit is started when the first IOCB is placed in the I/O queue. The IOM will process the next IOCB in the queue without MCP intervention. Thus, a queue is only started when a IOCB is placed in a empty queue.

#### RESULT QUEUE:

When an I/O has finished, the IOM will place the IOCB in the result queue. Each IOM has it's own result queue. All I/O operations done by an IOM are placed in the same result queue.

The operator can decide when it wants the MCP to be interrupted by the IOM. This interrupt will cause the MCP to look at the result queue. The system can be set up (see SBP ODT command) so the IOM will interrupt the MCP when:

- An I/O queue is empty or CPM is idle
- Each I/O finishes
- A task is waiting on an I/O or CPM is idle
- The CPM is idle

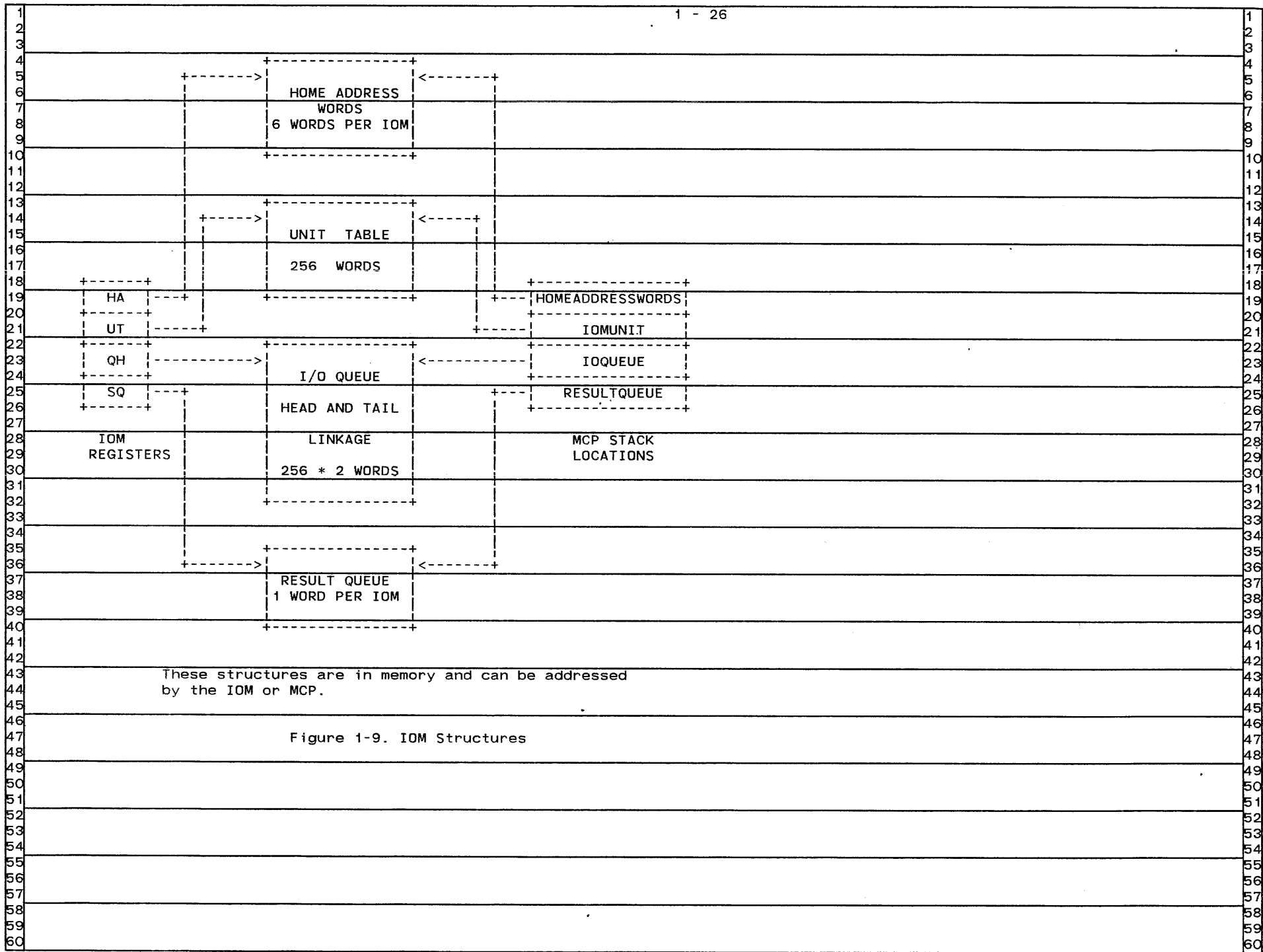
#### UNIT TABLE:

The unit table contains unit information used by the IOM and MCP. This information includes ring walk information, number of channels on an exchange and a lock bit for the unit table and I/O queue.

#### HOME ADDRESS:

The home address words are used by the MCP to give the IOM commands. The types of commands to an IOM include load registers (HA, UT, QH, SQ), start I/O, interrogate peripheral status and scan out DCP (INITIALIZE, HALT and ATTENTION NEEDED). There are 6 home address words and are defined as follows:

1		1
2		2
3		3
4	HA[0]	4
5	Command to IOM.	5
6	HA[1]	6
7	Special information (peripheral status).	7
8	HA[2]	8
9	HALOCKWORD. Buzzed by MCP to get	9
10	control of Home Address words.	10
11	HA[3]	11
12	Not used.	12
13	HA[4]	13
14	IOM fail word, or hard H/L	14
15	result descriptor.	15
16	HA[5]	16
17	Sync I/O result descriptor.	17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



These structures are in memory and can be addressed by the IOM or MCP.

Figure 1-9. IOM Structures

## IOCB's

An I/O operation is defined by an IOCB (18 words). The first 7 words are used by the IOM (hardware) and the MCP. The other words are used by the MCP only. The words in an IOCB used by hardware are:

- 0 IOMNEXTLINK.  
Link to next IOCB.
- 1 IOMSIDELINK.  
Not used by hardware.
- 2 IOMAREADDESC.  
Buffer descriptor, address and length.
- 3 IOMIOCW.  
IOCW for I/O.
- 4 IOMCDL.  
CDL built by software for IOM.
- 5 PHYSICALRD.  
Hardware result descriptor.
- 6 IOMTIMECELL.  
Channel busy time.

The other words of an IOCB can be found in the MCP. These words include I/O mask, reference to event, reference to FIB and other information.

## I/O FLOW

The flow of an IOCB will be traced. This flow will start with the MCP building an IOCB. The required words are placed in the IOCB. The MCP will link the IOCB into the proper I/O queue. The following procedure is used:

```

BUZZ47(IOMUNIT[UNITNO]) % LOCK UNIT TABLE AND I/O QUEUE
GET TAIL WORD FOR UNIT
IF THE TAIL WORD IS 0 THEN
  PLACE ADDRESS OF THIS IOCB IN TAIL AND
  HEAD WORD FOR THE QUEUE
ELSE
  PLACE THE ADDRESS OF THIS IOCB IN THE IOMNEXTLINK
  WORD OF THE LAST IOCB IN THE QUEUE.
  PLACE THE ADDRESS OF THIS IOCB IN THE
  TAIL WORD FOR THE QUEUE.
UNLOCK AND STORE IOMUNIT[UNITNO]
IF THIS I/O IS THE FIRST IOCB IN THE QUEUE THEN
  BUZZ(HALOCKWORD)
  BE SURE LAST COMMAND IS CLEAR
  STORE A START I/O COMMAND IN HA[0]
  INTERRUPT THE IOM
  UNLOCK THE HALOCKWORD

```

Before we look at what the IOM does when it is interrupted a few comments need to be made about the

start I/O process.

Bit 47 of a home address command is called the lock bit. This bit must be on for all IOM commands.

All IOM's that have a path to the unit will be interrupted. Only one IOM will do the I/O operation. The MCP maintains information of which IOM's to interrupt for a given unit.

The IOM will read HA[0] then zero it. Before a CPM places a new command in HA[0] it will make sure the IOM has cleared the last command. If the last command has not been cleared the CPM will wait .5 sec. If the command is still not clear the CPM will interrupt the IOM again. If the command is still not clear after .5 sec the IOM will be removed from the system.

Once ring walk is entered the IOM will follow the next unit field of the unit table word looking for a unit with the JB bit on. This bit indicates the I/O was not done because there was no channel available at the time. Ring walk will give all units on an exchange equal priority.

The IOM has a thruput scheduler for I/O's which are marked by the MCP (DISK, PACK and TAPE). A count of the I/O's in process will be kept and compared against a maximum value (set by F.E.). When the maximum value is reached the IOM will queue requests to start more I/O's. This will only be done for units in the PCI. The queue size for start I/O commands is 16. If the queue is exceeded, the IOM will not respond to a start I/O command.

Fail IOCB's will be generated by the IOM if an error is detected by the IOM which is not associated with a unit. The IOM will use IOCB's in unit 0. It will place a fail word in the PHYSICALRD word of the IOCB and place it in the result queue. The CPM will be interrupted.

An I/O error to a unit will stop processing on that unit.

Figure 1-10 thru 1-14 are flow charts of start I/O and finish I/O operations.

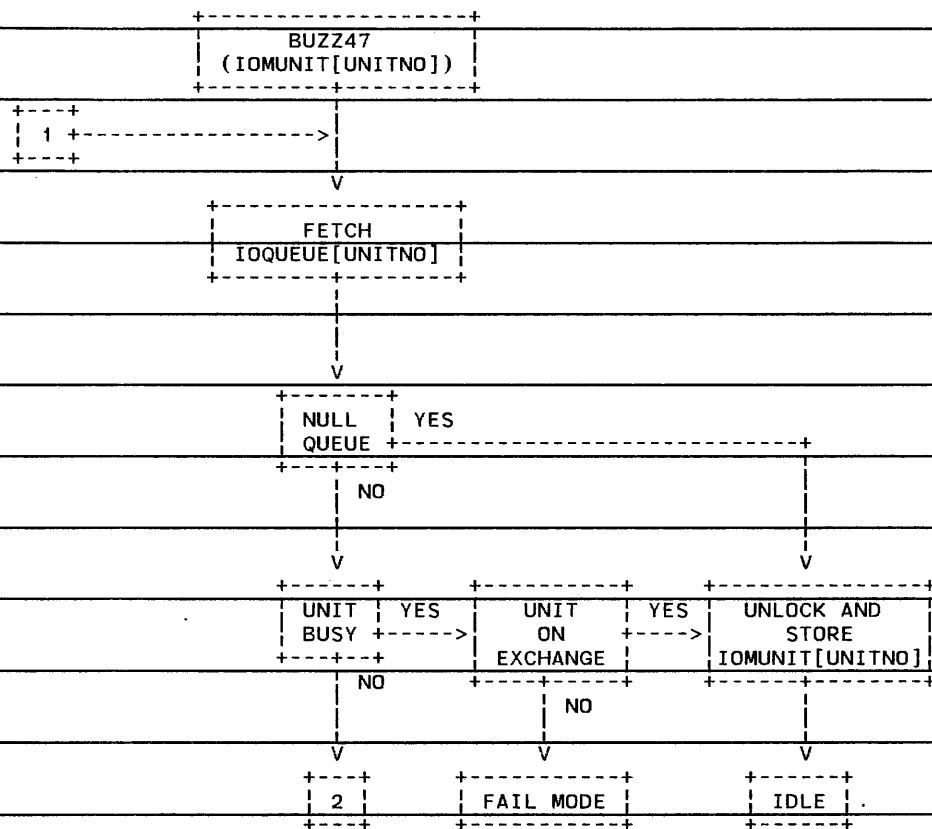


Figure 1-10. START I/O (PAGE 1 OF 2)

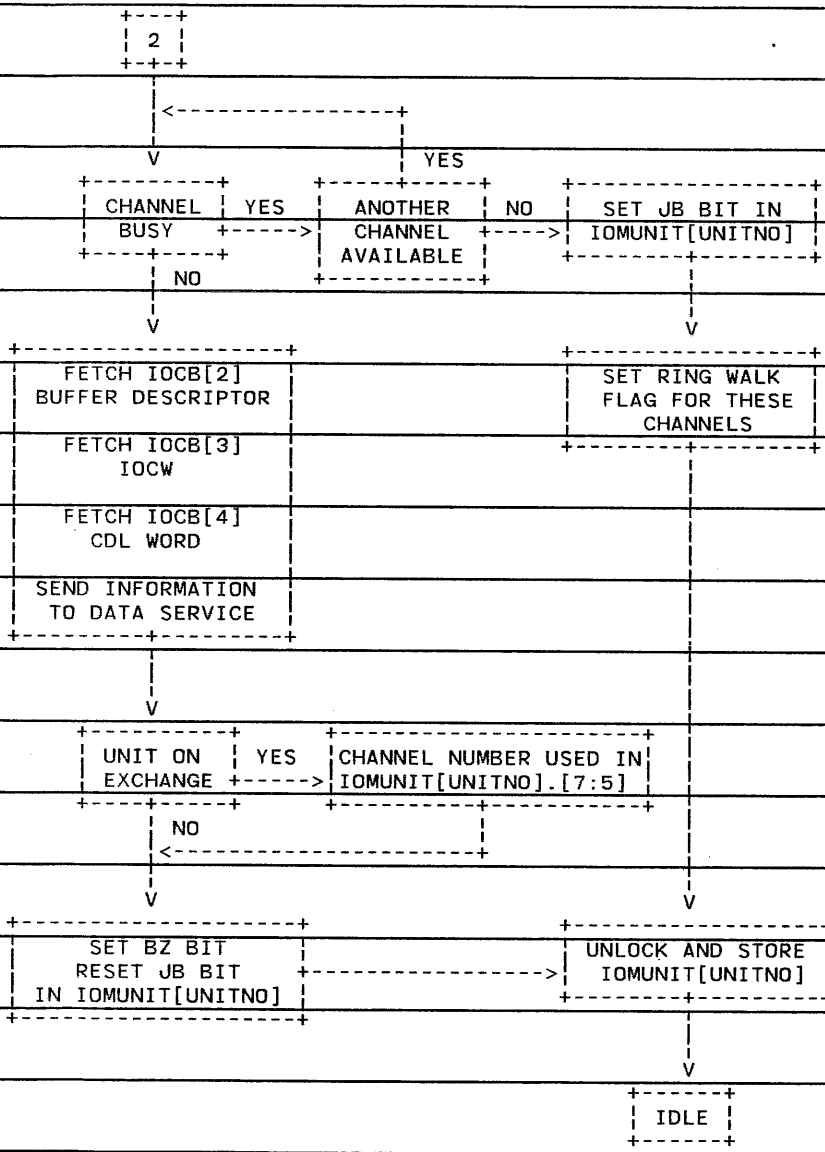


Figure 1-11. START I/O (PAGE 2 OF 2)

BUZZ47  
(IOMUNIT[UNITNO])

v

FETCH IOQUEUE[UNITNO]

v

GET RESULT DESCRIPTOR  
FROM DATA SERVICE AND  
STORE IN IOCB[5]

STORE I/O TIME IN  
IOCB[6]

v

FETCH NEXT LINK FROM  
IOCB[0]

STORE NEXT LINK IN  
IOQUEUE[UNITNO]

v

NEXT LINK  
EQL 0

YES

STORE 0 IN  
IOQUEUE[UNITNO+256]

NO

v

BUZZ47  
(STATUSQ)

v

3

Figure 1-12. TERMINATE I/O (PAGE 1 OF 3)



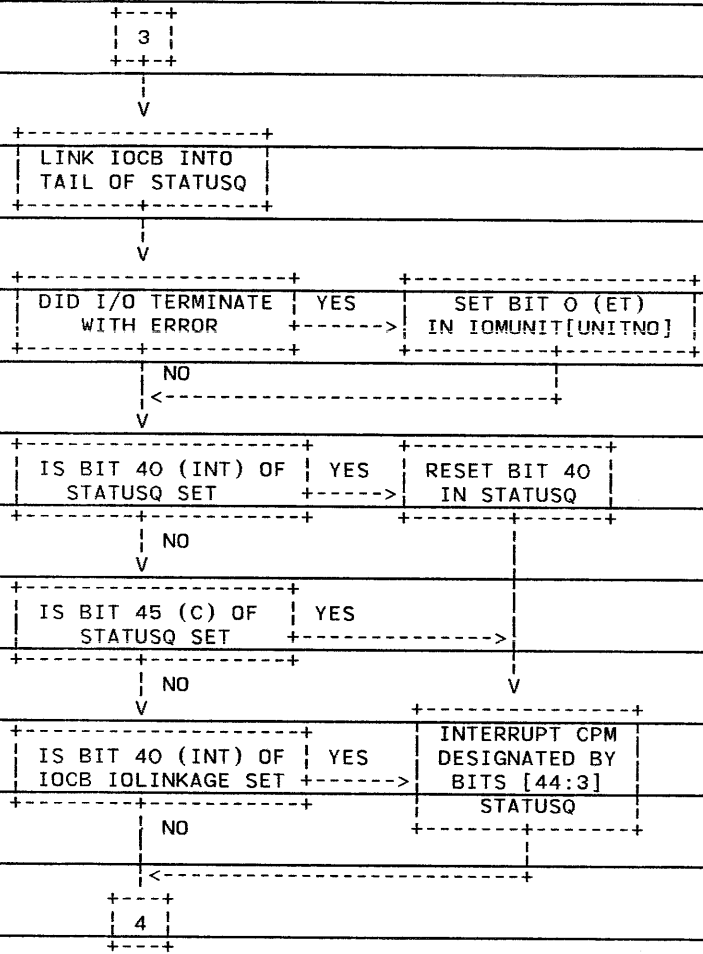


Figure 1-13. TERMINATE I/O (PAGE 2 OF 3)

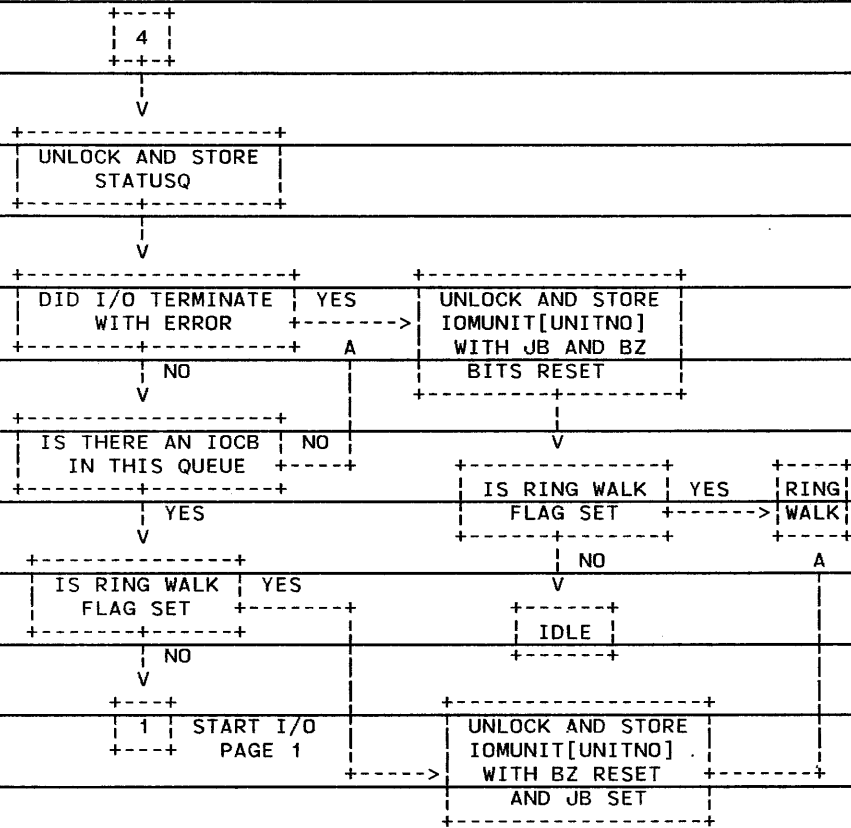


Figure 1-14. TERMINATE I/O (PAGE 3 OF 3)

1  
2  
3  
4  
5  
6 B7800 TIGHTLY COUPLED SYSTEMS  
7  
8

9 The primary goal of tightly coupling is to achieve a  
10 multiprocessor system with more addressable memory than could  
11 be configured for a monolithic multiprocessor system. The 20  
12 bit address field limits a monolithic system to one million  
13 words (6 MB) of addressable memory. A Tightly Coupled (TC)  
14 system, by making certain address ranges visible only to  
15 certain requestors, allows those ranges to be duplicated, and  
16 expands the total amount of memory accessible to the TC system  
17 to more than one million words. The B7800 implementation of  
18 Tightly Coupled Systems was accomplished under the restriction  
19 that it require absolutely no special hardware changes.  
20

21 ARCHITECTURE  
22

23 B7800 architecture dictated that a TC system be initialized  
24 (and run), as would any other multiprocessor system, from a  
25 single Halt/Load unit. A B7800 Tightly Coupled system has its  
26 low address memory visible to all on-line requestors; the  
27 MCM(s) comprising this shared memory constitute the SHARED or  
28 GLOBAL BOX. Above the highest SHARED mod is local memory:  
29 those MCMs which are visible only to specific requestors. A  
30 local BOX on a B7800 TC system consists of one or more CPMs,  
31 one or more IOMs and the high address MCM(s).  
32

33 An important configuration restriction imposed on B7800 TC  
34 systems is that no unit may have in-use paths to more than one  
35 local box. Units may be exchanged across IOMs as long as the  
36 IOMs are in the same local box.  
37

38 CONFIGURATION FILES  
39

40 The configuration file is used to hold GROUP descriptions  
41 which define how the system is to be configured. The  
42 RECONFIGURE ODT command causes the MCP to assume a specified  
43 GROUP configuration.  
44

45 The utility program, SYSTEM/CONFIGURATOR, is used to construct  
46 object configuration files (which can be used by the MCP) from  
47 symbolic configuration files. The internal file names used by  
48 CONFIGURATOR are SOURCE for the input file and OBJECT for the  
49 output file (defaults: input TITLE SYMBOL/CONFIGURATION,  
50 output TITLE SYSTEM/CONFIGURATION).  
51

52 Input to SYSTEM/CONFIGURATOR consists of one or more GROUP  
53 descriptions, each with a unique GROUP name and its associated  
54 mainframe and peripheral specifications. The output file from  
55 SYSTEM/CONFIGURATOR is designated as the configuration file  
56 with the CF ODT command. The RECONFIGURE GROUP AS <GROUP  
57 Name> ODT command causes the configuration found to be  
58 displayed; an operator OK causes the MCP tables on the H/L  
59  
60

unit to be updated. The system will deadstop to allow the operator to make whatever changes are necessary to mainframe switch settings for the new configuration. When the system is loaded it will result in initialization based on the new configuration.

SUBSYSTEM DEFINITIONS

A TC system contains one GLOBAL memory subsystem and a local memory subsystem for each box. The local memory subsystems are identified by their BOX numbers and the GLOBAL memory is identified by the letter G or GLOBAL.

Subsystems may be given names in the form of alphanumeric identifiers. Subsystem identifiers are defined by the operator through the ODT command MS (Make Subsystem). Subsystems may be defined including any one or more memory subsystems in the system.

The user specifies a subsystem by using the task attribute SUBSYSTEM which may assigned a value only when the task is inactive. The SUBSYSTEM task attribute can be used in WFL, CANDE, programming languages and on Job Queues. The SUBSYSTEM attribute is treated as a request to guide placement of a task in memory.

B7900 HARDWARE OVERVIEW

This sub-section provides an overview of B7900 hardware and software. The information has been extracted from the 34 B7900 D-notes. In many cases the information has been copied from these D-notes. It is provided in this document for completeness.

INTRODUCTION

A B7900 system is comprised of several different modules. These include Central Processing Modules (CPM), Memory Subsystem Modules (MSM), I/O Subsystem Modules (IOSM), I/O Expansion Cabinets, a System Control Cabinet (SCC) and the System Console.

The MSM allows the B7900 to address more memory than any previous large system and allows simultaneous read/write operations.

The B7900 CPM is a pipelined processor with multiple (data and code) high speed cache (associative) memories.

The IOSM is comprised of two separate processing modules: the Auxiliary Processor (AP) and the Host Data Unit (HDU). The AP is a small, strictly serial processor which functions as a maintenance processor or in the B7900 system as an auxiliary processor.

The HDU manages all I/O operations on the system. The I/O subsystem it supports is the DLP-based Input/Output subsystem used by all of the 900 series systems. The IOSM also contains four base modules which contain the Data Link Processors which control the peripherals. The I/O Expansion Cabinet is used if additional base modules are required to configure the I/O subsystem. It contains up to six base modules.

The SCC houses three components: the master clock, the Maintenance Exchange (MEX), and the HDU/MSM profile card test station. The master clock provides clock signals to each of the boxes in the system. The MEX provides connection between the components which may act as "maintenance processors" and the maintenance interface in each box in the system.

The System Console provides table space for the system ODTs. For system initialization and maintenance, a modified MTS2 series terminal (MMTS2) is used. The MMTS2 acts as a maintenance processor, using two ICMD drives packaged in the leg of the maintenance console for program and file storage. The MMTS2 has access to the MEX for system initialization and

1  
2  
3  
4 maintenance, and to the I/O Testbus for off-line DLP testing.  
5

6 In the text that follows, these new features are explained in  
7 terms of their impact on the hardware and software that  
8 comprise the B7900 system. This explanation is by no means  
9 comprehensive.

10  
11  
12 CENTRAL PROCESSING MODULE (CPM)  
13 -----  
14

15 The B7900 CPM is a powerful pipelined processor with high  
16 speed code and data caches. It is object-code compatible with  
17 all previous B5000, B6000 and B7000 machines except in two  
18 respects: it does not support the six bit character data type  
19 (BCL) and, like the B7800, it does not support vector-mode.  
20

21 In addition to these changes, the basic design of the B7900  
22 includes many improvements over the B7800. The two most  
23 noticeable of these are the more accurate arithmetic  
24 operations and more efficient branch handling.  
25

26 The B7900 CPM performs arithmetic operations more accurately  
27 than any previous machine (except the B5900). This increased  
28 accuracy should cause numerical algorithms to converge  
29 slightly faster and may cause minor differences in arithmetic  
30 results as compared to previous machines.  
31

32 As the B7900 CPM is a pipelined processor, it has the  
33 traditional difficulty handling conditional branches within a  
34 program. It must always choose a particular direction for a  
35 branch and prepare to execute the proper code for that  
36 direction. For example, the B7800 always assumes that the  
37 condition of an IF statement in FORTRAN or ALGOL will be true  
38 and fills its pipeline with the code that would be executed  
39 based on this assumption. Performance of the pipeline is  
40 penalized when branches do not go according to their assumed  
41 direction. The B7900 mitigates this penalty by remembering  
42 the direction which the branch last took, and assuming the  
43 same branch will be taken the next time.  
44

45 The B7900 CPM has an increased memory addressing capability  
46 over the B7800 CPM, which is accomplished by memory  
47 environment registers. Because of the 20-bit address field,  
48 the CPM still only addresses one million words at any one  
49 time, but it can address all of memory by changing its  
50 environment registers to access different million-word  
51 environments. Extended Memory is discussed in the Memory  
52 Management sub-section.  
53

54 The CPM consists of the following hardware modules: Program  
55 Control Unit, Data Reference Unit, Execution Unit, Store  
56 Queue, Memory Access Unit, and Card Test Station.  
57  
58  
59  
60

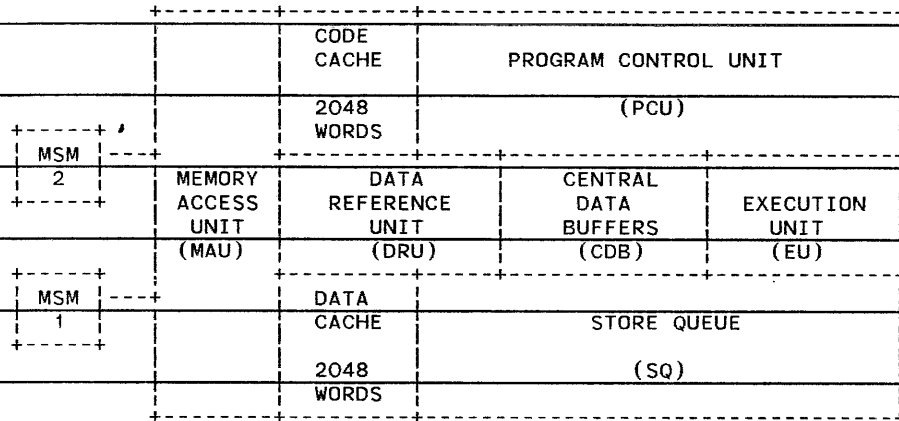


Figure 1-15. B7900 Central Processor Module (CPM)

The Program Control Unit (PCU) extracts code from the code stream and generates instructions for the Execution Unit and Data Reference Unit. These instructions are placed in queues along with a job number and addresses in the Central Data Buffer where input data can be found and where the result is to be stored. The PCU is the key unit for the pipeline by causing data to be ready for the Execution Unit ahead of time. Operator concatenation is also performed by the PCU.

The Data Reference Unit (DRU), upon command from the PCU, fetches data either from associative memory or main memory. Extensive use of pipelining allows one-clock references to associative memory.

The Execution Unit (EU) performs all arithmetic and logical operations using data in the Central Data Buffer. The result is either placed in the Central Data Buffer or may be sent to associative memory and the store queue.

The Store Queue (SQ) is used to reduce traffic to main memory by buffering data. Repeated stores to the same address will be performed as one store to main memory. Also, adjacent stores will be grouped into one multi-word store by the store queue.

The Memory Access Unit (MAU) provides the interface to main memory. The MAU can interface to two MSMs. On memory references the CPM Environment Register along with the 20-bit address are sent to the MSM.

The Card Test Station is used to run maintenance tests on CPM-style cards.

MEMORY SUBSYSTEM MODULE (MSM)

The B7900 includes one or two Memory Subsystem Modules (MSMs) totaling up to 96 megabytes. A MSM has a bandwidth of 72 MB/SEC. Each MSM contains from one to four Memory Storage Units (MSUs) and a Memory Control (MC).

The Memory Storage Unit (MSU) contains one or two sub-modules, each consisting of 6 megabytes. The memory is eight-word phased (transmission of words at clock rate) within each sub-module. This means the system can run with half an MSU or one sub-module without losing the speed of the eight-word phasing.

The Memory Control consists of the following major modules: Requestor Interface Adapter (RIA), Memory Interface Adapter (MIA), Priority Resolution Module, Error Module, Requestor Interrupt Module and Maintenance Module.

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56  
57  
58  
59  
60



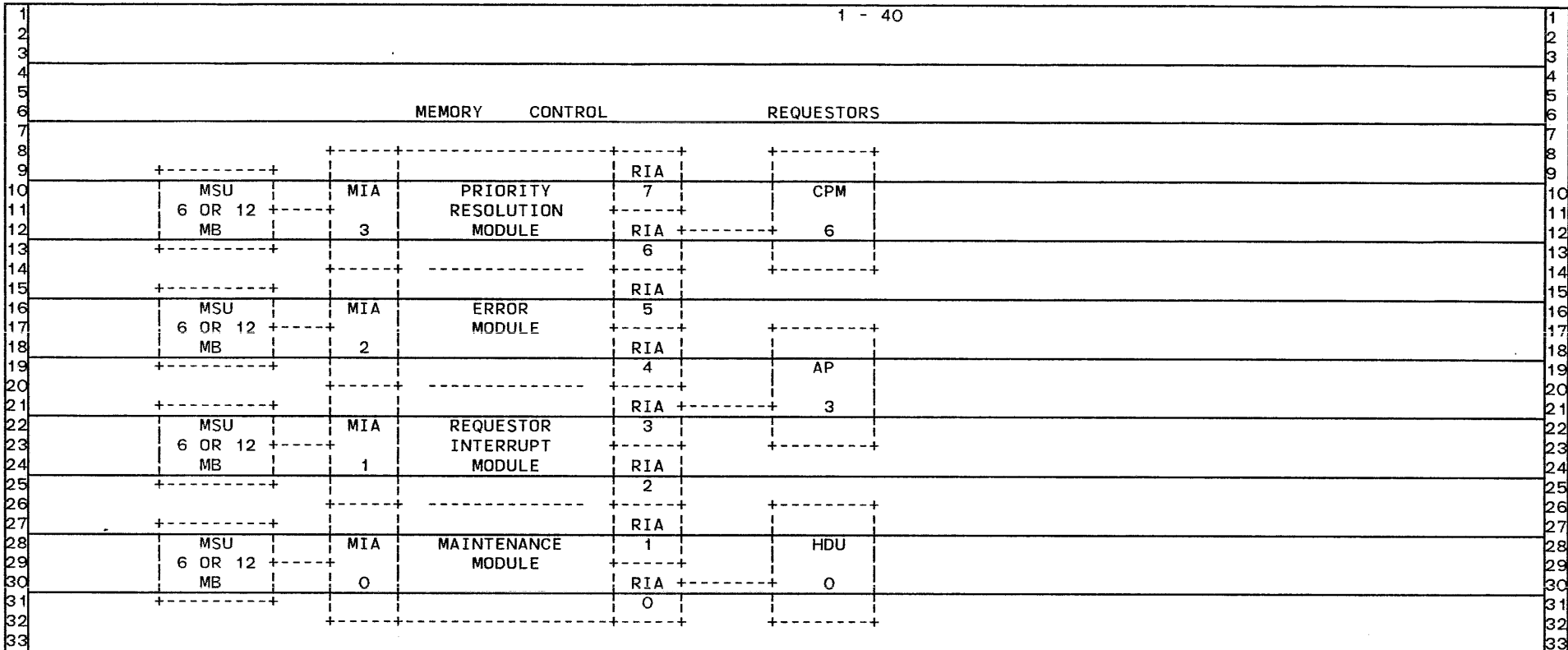
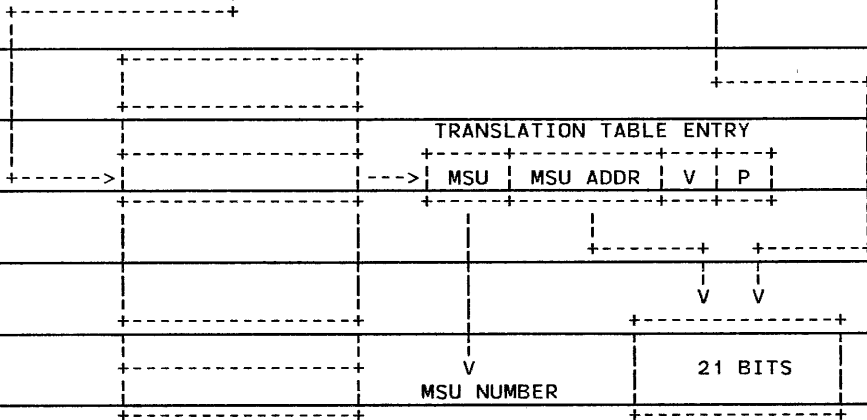
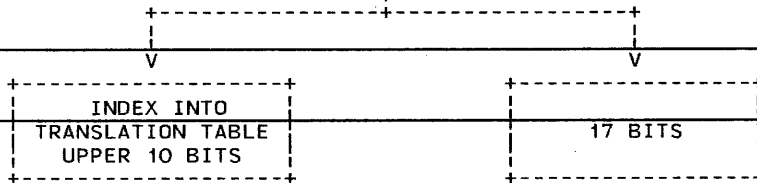
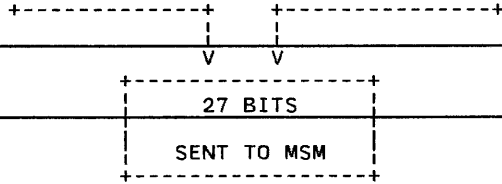
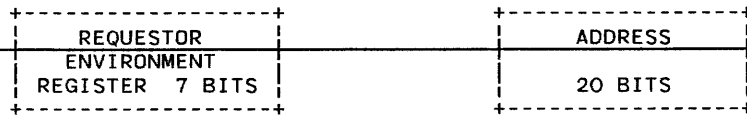


Figure 1-16. B7900 Memory Subsystem Module (MSM)

Each requestor interfaces to memory through its own Requestor Interface Adapter (RIA). For any memory operation, a control word and up to eight data words are sent by the requestor to its RIA. The RIA checks these words for errors, reports any errors to the requestor, and stores them in a buffer memory until the priority resolver commands their transmission to the Memory Interface Adapter (MIA).

Each RIA contains an address translation table which determines the validity of the address for this MSM and where the address resides physically in storage. That is, the address translation table provides a mapping from environment number and address to physical location in memory. Each address translation table contains 1024 eight bit words. Each entry provides information for a page (128K words) of memory. The table entry contains a MSU number (2 bits), MSU relative address (4 bits), validity bit and parity bit.



MSU NUMBER AND 21 BIT ADDRESS ARE USED TO ACCESS PHYSICAL MEMORY

Figure 1-17. MSM Translation Table

The Memory Interface Adapter (MIA) controls the flow of information between the memory control internal bus structure and the Memory Storage Unit (MSU). A memory control contains from one to four MIAs with the capability of all MIAs making simultaneous requests to the MSU to which they are connected.

The order in which system requests are handled by the MSM is determined by the Priority Resolver Module. Requests are processed by a modified First In First Out (FIFO) algorithm. Snapshots of memory requests are recorded on a basic time interval as they arrive. The snapshots of requests are processed in the order they are taken.

Within a snapshot, requests will be serviced in the following sequence: read operations, write operations, special operations (except for write translation table), write translation table. The higher number requestor will be given the higher priority, if there are duplicate requests for a particular type, although this should have no effect on system performance.

By means of a modified Hamming Code, the Error Module can detect and correct single bit errors and detect two-bit errors. Any errors detected during MSU operations or internal memory control operations are reported to the RIA. The Error Module will report the control word or the fail word or both if requested by the requestor.

The requestor interrupt module is responsible for the correct routing of interrupts and for interrupting the destination requestor via unique interrupt lines on receipt of an incoming interrupt. There are two types of interrupt buffers maintained by the MSM for each requestor: hardware and software. These buffers are used to accumulate interrupts and can be read by the requestor via a system operation. The software interrupt buffer contains a bit mask of requestors from which interrupts have been received. The hardware interrupt buffer is an eight-bit mask which designates the type of interrupt.

The maintenance module performs all maintenance type functions for the MSM. In system mode, the maintenance module handles reading and writing the fields in the MSM BOXID. All other maintenance operations are done through the maintenance interface.

#### AUXILIARY PROCESSOR (AP/AMP)

-----

The Auxiliary Processor/Auxiliary Maintenance Processor (AP/AMP) is based on the B5900 Entry Level System and is compatible with the B7900 CPM. The AP/AMP has two distinct modes of operation: AP and AMP. To support these two

different modes two unique versions of micro-code, which communicate with different sets of hardware, are required.

In Auxiliary Processor (AP) mode, the AP/AMP acts as a co-processor to the B7900 CPM with the purpose of handling I/O finishes when the CPM is busy. The system has the capability of running with no CPMs (AP only mode). The AP will perform all the functions a CPM would do in this case. When the last CPM is removed from the system, the MCP notifies the operator (with an RSVP) that no CPMs remain. The operator must OK this waiting entry in order to allow jobs to run on the AP.

In Auxiliary Maintenance Processor (AMP) mode, the AP/AMP utilizes its own local memory and HDP hardware section of the processor to exist as a standalone system. The primary function of the AMP is to perform maintenance on the B7900 system. The AMP operationally is the same as the B5900.

The AP/AMP can be divided into five major units: Processor, Local Memory (LM), Cache and System Memory Controller (APMC), Interrupt and Maintenance (APIM) and the Host Dependent Port (HDP).

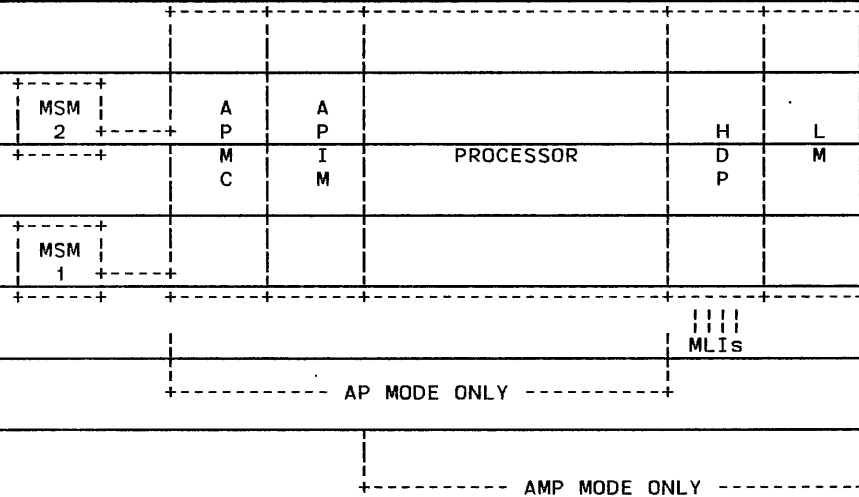


Figure 1-18. Auxiliary Processor (AP/AMP)

The processor module performs all arithmetic and logical operations for the AP/AMP.

When the AP/AMP is running in AMP mode, the HDP section becomes active. The HDP provides the I/O subsystem interface

for the AMP.

When the AP/AMP is running in AMP mode, the local memory becomes active and the B7900 MSM memory is no longer used. The local memory has a capacity of 512K words. Error correction and logging are features provided by the LM.

When the AP/AMP is running in AP mode, the AP interfaces to the B7900 MSM memory by means of the APMC (Cache and System Memory Controller) module. The APMC can interface with two MSMs. The cache memory consists of 128 words of code and 128 words of data. The cache hardware performs automatic prefetching of code. Any word in the cache can be accessed in 1 clock and the cache can be purged in 2 clocks.

The AP/AMP Interrupt and Maintenance Module (APIM) interfaces to the MSM system interrupt bus and Maintenance Exchange (MEX), contains AP box identification information, and provides support for the AP Memory Control/Cache (APMC).

HOST DATA UNIT (HDU)

-----

The HDU contains a Memory Bus Control (MBC) which handles the HDU's interface to the MSM, the Queue Manager (QM) which handles the I/O queue structures and flow control, and three Host Dependent Ports (HDPs) which connect to the I/O subsystem. Each HDP is capable of operating at a burst rate of 8 Megabytes/Second for a total of 24 Megabytes/Second per HDU.

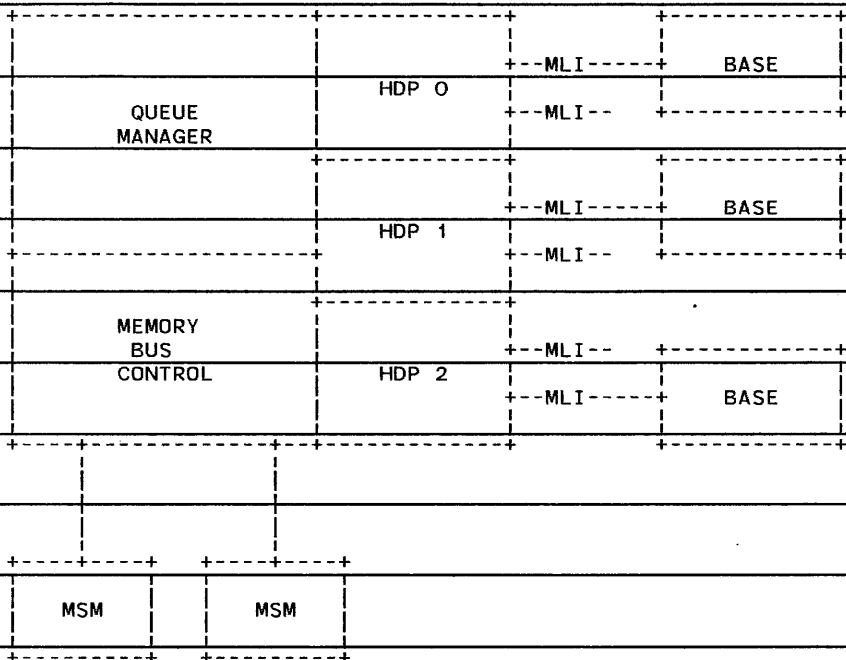


Figure 1-19. Host Data Unit (HDU)

The Queue Manager (QM) is responsible for managing the I/O queue structures and the scheduling of jobs for the HDPs. It has a 4K word local memory which is used to selectively store copies of the IOCBs that have been issued to the I/O subsystem. 256 words of this local memory are used to contain the code for the MCP Boot and the Minimal Configuration information needed to initialize the system. It is called the HDU's Halt Load RAM.

The Memory Bus Control (MBC) provides the HDU's interface to the memory subsystem. It can connect to up to two MSMs. The MBC handles up to eight-word transfers and uses the same memory addressing mechanism (Environment Register plus 20-bit address) as the CPM. It also handles the requestor to requestor interrupts which use the memory buses.

The Host Dependent Port (HDP) provides the HDU's interface to the I/O subsystem using the Message Level Interface (MLI). Each HDP controls two MLI ports, each of which connects to one

MLI cable. The HDPs control the MLI and are responsible for all data transfer for the system.

DLP-BASED I/O SUBSYSTEM COMPONENTS  
-----

Each HDP in the HDU contains two MLI ports which are connected to I/O bases via an MLI. An I/O Base is a rack which contains a Base Control Card (BCC), Distribution Cards (DCs), a Maintenance Card (MC), Data Link Processors (DLPs), and, optionally, a Path Selection Module (PSM), and/or Line Expansion Module (LEM).

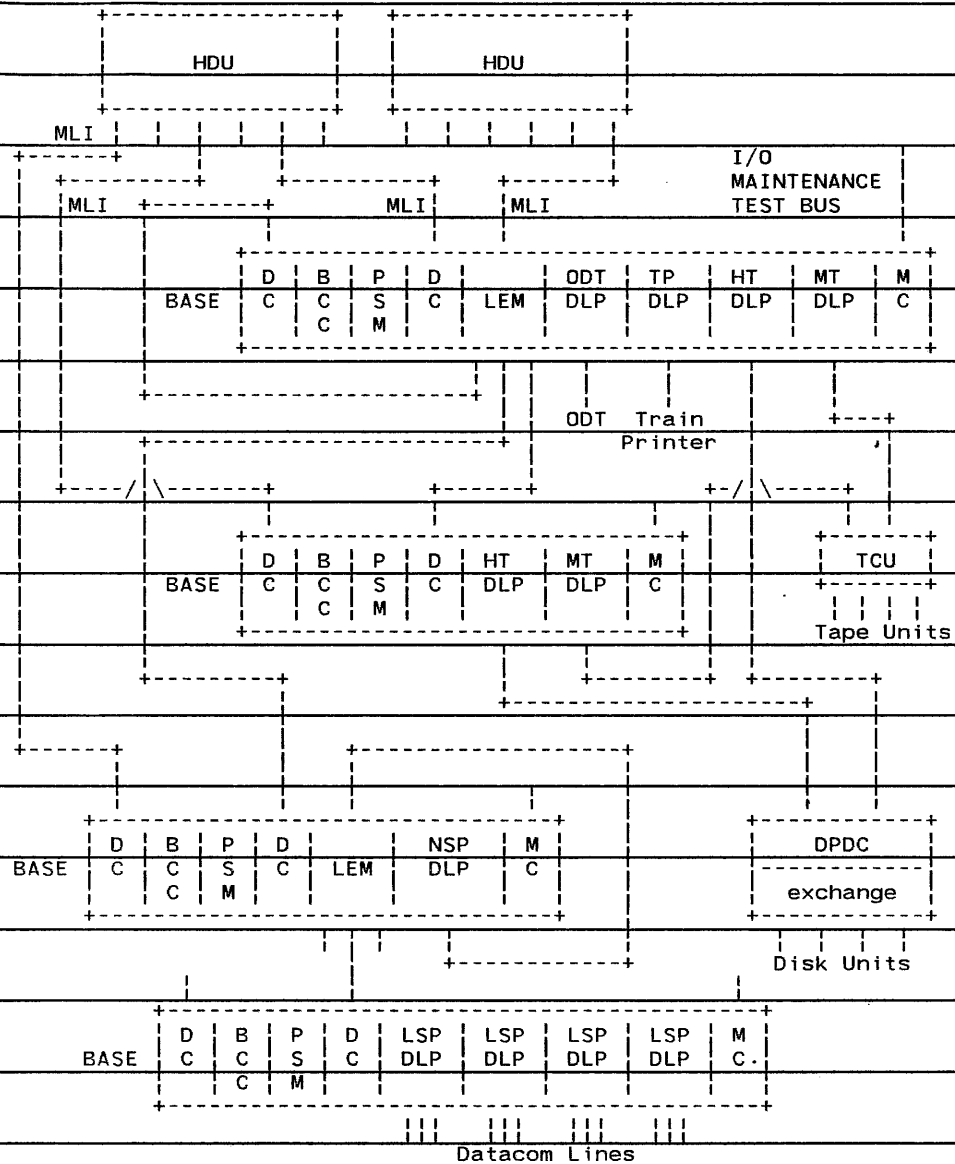


Figure 1-20. UID Subsystem Components



The Maintenance Card (MC) provides maintenance functions for the base. It connects to the I/O Maintenance Test Bus which is used to perform off-line maintenance.

A Distribution Card (DC) provides the interface between the base module and the host. One MLI cable from a host's MLI Port connects to a DC in the Base Module. Through the DC, a host can communicate with the BCC and the DLPs in the base.

There can be only six DCs in a base although the "addressing" allows for eight. DCs are "addressed" by jumpering an ID from 0 to 7.

A Path Selection Module (PSM) coordinates the activities of multiple DCs in a base, and is required in a base which has more than one DC. Several hosts can be actively communicating with DLPs in a base, but only one DC can access the BCC or have access to the base backplane at any time. The PSM resolves these conflicts.

The PSM also uses the masks in the BCC in order to clear only the DLPs assigned to the host issuing a Master Clear by issuing a Selective Clear or Selective Master Clear (see below) to each assigned DLP.

The PSM contains eight field-installed jumpers (one for each possible DLP in the Base) which, when set, will cause a Selective Master Clear (instead of a Selective Clear) to be issued to the corresponding DLP when the host system issues a Master Clear.

The Base Control Card (BCC) provides identification and access control for a Base Module. The BCC contains two types of masks which are used to control access into a base and its DLPs. These are the DC Enable Mask and the DLP Enable Masks.

The eight-bit DC Enable Mask indicates which DCs are allowed access to the base. A locked-out DC will not respond to the host or execute a host-generated Master Clear.

The BCC also has eight 16-bit DLP Enable Masks, one for each possible DC in the base. A DC's DLP Enable Mask indicates to the PSM which DLPs this DC is allowed to access. The PSM also uses these masks in order to determine which DLPs will be cleared when a MLI Master Clear is received via a DC. Only those DLPs enabled to that DC will be cleared.

Along with the eight DLP Enable Masks, the BCC contains an Acquire Enable Bit for each possible DC in the base. This bit indicates whether or not another DC may "steal" DLPs from this DC.

The BCC also has a 32-bit Maintenance Mask that indicates which devices in the base (DLPs, DCs, BCC, PSM, LEM) are enabled on the I/O Maintenance Test Bus. Devices are

identified in the mask by their Maintenance Addresses (or Unit Numbers).

The DC Enable Mask and the DLP Enable Masks can be cleared only by one of the following manual operations:

1. Powering up the Base.
2. Pushing the BCC Clear Switch on the BCC.
3. Pushing the Base Clear Switch on the MC.

When the BCC is cleared, the DC Enable mask is set so all DCs are enabled in the Base, the DLP Enable masks are set so all DLPs are disabled from all DCs, all Acquire Enable bits are set to disable the "steal" function and Maintenance is enabled to all devices in the Base.

The DC Enable Mask can be set only via a DC (by software) if it is enabled in the DC Enable Mask. A DC can enable a DLP in its DLP Enable Mask only if the DLP is not already enabled in another DC's DLP Enable Mask. The exception to this rule is the ability to "steal" a DLP. In order for a DC (DC 1) to steal DLPs from another DC (DC 2), DC 2's host must have previously set DC 2's Acquire Enable bit. When DC 1's host issues a variant of the LOAD DLP MASK command, the specified DLPs will be disabled from DC 2's DLP Enable mask and enabled in DC 1's DLP Enable Mask. A DC may also share DLPs with other DCs and transfer DLPs to another DC.

A Line Expansion Module (LEM) enables a host to be connected to more bases by expanding one MLI to up to seven MLIs. It resides in a base.

A LEM consists of three types of cards. The type "A" card contains the basic LEM logic, two MLI ports (ports 6 and 7) and the Maintenance Address jumpers.

The 1x2 LEM ID, or type "C" card, provides two MLI ports (ports 1 and 2) and LEM identification. The 1x3 LEM Exchange, or type "B" card, provides three MLI Ports (ports 3, 4 and 5).

A Data Link Processor (DLP) provides the interface between hosts and peripheral devices. The operation of a DLP is device-dependent, and there are different types of DLPs to control different types of devices. All DLPs follow the MLI protocols to communicate with a host, perform a set of standard DLP operations, and return certain standard results. Included in these standard DLP operations is the TEST IDENTIFY operation which returns the DLP type and the DLP's strapped ID.

The DLPs that are currently supported on the B7900 are:

DLP	Peripheral
---	-----
CR1	Card Reader
HC2	ISC Host Control
HT1	(Host Transfer) 206,207,659,677 Disk Pack
MT1	Mag Tape: PE
MT2	Mag Tape: PE/GCR
MT3	Mag Tape: NRZ
ODT1	Operator Display Terminal
LSP1	Datacom Line Support: Sub-broadband
TP2	Buffered Printer: 1200/2000 LPM
NSP3	Datacom NSP: Blocked

There can currently be up to eight DLPs in a base although there are 16 "addresses" reserved. A DLP's "address" is determined from jumpers on the DLP's logic cards and is also its Maintenance Address.

DLP address assignment affects hardware priority algorithms: For proper operation of the system, these algorithms must be understood and the DLP addresses assigned accordingly.

Within each Base, concurrent requests by multiple DLPs are arbitrated by the Base Control Card based on the Base Module Priority of the DLPs. The Base Module Priority of a DLP is determined directly by its address; the DLP with address 7 has a Base Module Priority of 7, which is the highest possible priority. When presented with multiple DLP requests, the Base Control Card selects the DLP with the highest Base Module Priority.

DEVICE NAMING

The B7900 will have its I/O configuration described in a Peripheral Configuration Diagram (PCD). The PCD is the B7900 equivalent of the B7700/B7800 Unitcards and contains descriptions of all devices and connections in the peripheral subsystem. In the PCD all devices (including units, DLPs, LEMs, Bases and MLI Ports) are assigned arbitrary numbers, called Device Numbers, which will be used when referring to a device. The PCD also contains the information necessary to determine the paths to the device to be used in performing I/O operations.

The B7900 supports larger device numbers than previous systems, which supported eight-bit device numbers. Device numbers for the B7900 are currently 15-bit numbers (or between 1 and 32767). Although devices may be numbered between 1 and 32767, only 255 peripherals, NSPs and LSPs may be in a B7900

partition at one time. The B6800, B5900/B6900 and B7700/B7800 systems are still restricted to eight-bit device numbers (between 1 and 255).

The device number for an NSP or LSP must be an integer from 1 to 255. This device number must match the NDII program. An NSP can control up to 8 LSPs at any time.

The device number for an HC must be an integer from 1 to 255 if BNA is to be used.

The B7900 MCP will use the PCD to determine the paths to a device and will attempt to verify the information in the PCD before using a device.

#### AMP DEVICE NAMING

-----

The B7900 AMP, which is a modified B5900, uses the B5900 device naming rules. B5900 systems use the strapped ids in the I/O subsystem components in order to identify or name the peripheral units controlled by them. Currently the MCP only allows for eight-bit device numbers for the B5900. If a DLP can have only one unit attached to it, the lower eight bits of that unit's number are the same as the DLP's strapped ID. If there can be more than one unit controlled by a DLP, the unit's number is the DLP's strapped ID plus the unit's address relative to the DLP.

Peripherals not qualified on the B5900 are not supported on the AMP.

#### I/O RECONFIGURATION

-----

Online I/O reconfiguration on the B7900 has similar capabilities to the B7800. Many of the functions are identical, in syntax and semantics. Because of the difference in the hardware components, though, there are some differences in syntax and functionality, and there are several new and unique functions.

B7900 peripheral reconfiguration (FREE, ACQUIRE, RY, SV, UR, UR-, MOVE, REPLACE and CL) is identical to B7800 peripheral reconfiguration with two exceptions. The first is that the RY command with path selection to append devices and the SV (-) command to remove them are not supported. These functions can be accomplished by loading a new Peripheral Configuration Diagram (PCD). The second is that the syntax of the LH (Load Host) command (to load a controlware file to a disk pack drive controller) is slightly different. The DLP device number is used to specify the path instead of CH <#> IOM <#>.

There are two B7900 I/O reconfiguration capabilities which are functionally similar to B7800 capabilities. FREE/ACQUIRE IOP (IOP is used to refer to an HDU in the B7900 I/O ODT commands) corresponds to FREE/ACQUIRE IOM. UR DLP <dip device number> is similar to UR CH <#> IOM <#>.

The unique B7900 I/O reconfiguration capabilities include FREE/ACQUIRE DLP, UR/UR- MLI, UR/UR- BASE, UR/UR- BASE MAINT and UR/UR- DLP MAINT.

DLPs are not shareable between partitions and, therefore, must only be ACQUIRED by one partition at any time (FREE/ACQUIRE DLP). DLPs may be reserved in order to perform on-line maintenance or to suspend operations to the DLP via UR/UR-DLP. Off-line maintenance via the I/O Maintenance Test Bus can be performed using UR/UR- DLP MAINT.

Bases are shared by all B7900 partitions to which they are connected. Bases may be reserved in order to suspend operations through the Base or reserved for off-line maintenance.

MLIs are implicitly ACQUIRED with the HDUs to which they are connected. They may be reserved in order to suspend operations through a particular MLI.

#### I/O UNIT/DLP/MLI LOAD BALANCING

-----

Each Base typically has multiple Distribution Cards and, thus, there are typically multiple paths into each Base. Each of these paths is connected directly to an MLI or fans out from an MLI via a LEM. Between a running partition and a particular Base, there is at any given time a set of viable MLIs. This set of MLIs is relatively static; it can change only when HDUs are FREEd/ACQUIRED or MLIs are URed/UR-ed/BROKEN. The MCP automatically attempts to evenly distribute I/O traffic over all viable MLIs. The balancing algorithm is always active and does not require operator action.

Pack and tape units (which can be connected to an exchange) may be accessible by more than one DLP. (Note: each DLP or peripheral should be ACQUIRED in only one partition at a time.) Between a running partition and a particular string of exchanged units, there is at any given time a set of viable DLPs. This set of DLPs is relatively static; it can change only when HDUs, MLIs, Bases, or DLPs are reconfigured or marked BROKEN. The MCP automatically attempts to evenly distribute I/O traffic over all viable DLPs. The balancing algorithm is always active and does not require operator action.

System Option 39 (PATHBALANCING) has no effect on B7000

Systems, including B7900 Systems.

#### I/O TIME

--- ----

Reported "I/O Time" for tasks using moving-head media (e.g., packs) will generally be greater on HDU Systems (e.g., B7900) than on IOM Systems. This discrepancy results from the different handling of seek operations by the two I/O subsystems and does not indicate that the actual I/O operations are slower on the B7900.

On IOM Systems, I/O Time on moving-head media is the sum of rotational latency and data transfer time. On HDU Systems, I/O Time on moving-head media is the sum of seek time, rotational latency, and data transfer time.

#### MAINTENANCE HARDWARE

-----

The B7900 incorporates "soft" access to low-level machine state. Using this mechanism, initialization and maintenance software may interrogate and set flip-flops, registers and RAM storages in the various boxes in the system.

Within any system, every box is cabled to the System Control Cabinet (SCC) which houses the Maintenance Exchange (MEX). The MEX provides for connection from any of the "maintenance processors" in the system to each of the boxes in the system. The overall management of the maintenance bus is a "master-slave" relationship, where "maintenance processors" are masters and the various boxes they access are slaves. The MEX includes Fan In Adaptors (FIAs) and Fan Out Adaptors (FOAs). There is a FIA for each of the possible masters, and a FOA for each of the slaves in the system. Within each box (slave) there is "on-board" maintenance interface logic to allow for state access to that box. The overall arrangement of FIAs, MEX, FOAs and on-board maintenance logic is commonly called the "maintenance bus".

A MIP is a hardware component housed in the "leg" of a B7900 Console. It is cabled to a pair of ICMD (floppy) drives, also housed in the console leg. The MIP may also be cabled to the I/O Testbus. A MIP is driven by a Modified MTS2 terminal; only one MMTS2 may be cabled to any particular MIP, so the MMTS2/MIP will typically be paired and thought of as one unit.

A MMTS2/MIP which is cabled to the MEX may act as a "maintenance processor" for the system.

In any system, every processor will be cabled as a master to the MEX. This allows any "running" B7900 partition to act as a "maintenance processor" for hardware maintenance of boxes not in the partition. The maintenance software used in this

mode is IDA.

In any system, one or more APs will be configurable as AMPs. Using its connection to the MEX, an AMP may act as "maintenance processor"; this is done when it is not possible to configure a B7900 partition (for example, all MSMs down). The maintenance software used in this mode is IDA.

Built into the CPM cabinet, and appearing over the maintenance bus as another "slave" station, is the CPM profile card test station. The software used for CPM card testing is IDA.

#### MAINFRAME MAINTENANCE

-----

Mainframe maintenance for the B7900 is divided into two groups: the AP/AMP and the other mainframe boxes (MSM, HDU, CPM).

Maintenance for the AP/AMP is done using the MMTS2/MIP on the System Console. Two software packages are used: BEAM and APCON. BEAM provides for "low-level" static testing of the AP/AMP. The test patterns for these static tests are packaged on a set of ICMD diskettes, and the BEAM program uses the MIP to run the patterns against the AP/AMP hardware. APCON provides for dynamic tests. The dynamic tests are written in AP/AMP microcode, and execute on the AP/AMP itself, with APCON serving to load the tests from their ICMD diskette and to monitor their execution.

The CPM, MSM and HDU are maintained from a "running system", using the IDA software. IDA is an ALGOL program, supported by a set of maintenance libraries. It runs under MCP control, using normal system resources (such as CANDE). When enough hardware remains operational to configure and run a B7900 partition, IDA is executed on that partition. When it is not possible to configure a B7900 partition (for example, in a model B7900F with the MSM down), the AP/AMP is configured and run as an AMP, and IDA executes there. In either case, IDA uses the system processor's connection to the MEX to gain maintenance bus access to the box under test. Both dynamic and static tests are provided for each of the box types. IDA is also used to drive the two card test stations: the CPM profile test station, packaged in the CPM itself, and the HDU/MSM profile test station, packaged in the SCC.

#### I/O SUBSYSTEM MAINTENANCE

-----

The B7900 I/O subsystem will be maintained either in an off-line mode or an on-line mode. Off-line maintenance allows for the execution of diagnostic tests on the I/O components in a Base. These tests are run from the MMTS2/MIP on the

Maintenance Console, accessing state in the off-line device via the I/O Maintenance Test Bus. The test patterns are shipped on ICMD diskettes, along with the various I/O components.

On-line maintenance allows for confidence and maintenance testing of DLPs and the peripherals they control. These tests will be performed by running Peripheral Test Language (PTL) programs through Peripheral Test Driver (PTD) under MCP control.

#### PARTITIONING

In a B7900 system, "Partitioning" provides a means whereby a subset of the boxes in the system may be managed by a single MCP. Moreover, partitioning prevents any box not in the subset from in any way interfering with the successful execution of the subset. Such a subset is called a partition. This capability may be used in two ways:

A broken box may be tested while the remainder of the boxes executes as a partition. In this scheme the partition is protected against interference from the broken box.

A given system may be "split" into more than one partition, with each partition run independently. In this scheme each partition is protected against interference from actions in the other partitions.

Partitioning is established by software, but enforced by hardware. It involves the control of three interfaces: the system interface, the I/O subsystem interface, and the maintenance interface. Partition membership is marked in the Partition Identification Register (PID) in each box.

For example, in a Model H or K B7900 with the PAC kit, the system can be run as a single partition, including all boxes. A "broken" box may be removed from the partition, tested, and returned without jeopardizing the continued execution of the partition. Further, the system may be "split" into two partitions; each half would be run completely independently of the other.

The software involved in making partitions are SYCON and the MCP. SYCON will initially establish a partition upon instructions from the user. This includes setting the controls for the system and maintenance interface, and the PIDs in all boxes in the partition. SYCON will then start the MCP running. SYCON is further discussed in the section on System Initialization.

The MCP will, as part of its initialization, establish control



over the maintenance interface, thus disabling the maintenance bus and excluding the "maintenance processors" from further access over the bus.

Boxes may be moved into a running partition only by the MCP executing in that partition, using the MCP's ACQUIRE command.

During this acquisition, the system interface controls in the boxes currently in the partition will be updated, to acknowledge the existence of the incoming box, and the system

interface controls in the incoming box will be updated to reflect the partition's membership. The PID in the incoming box will be set to match the partition number. Similarly, a

box may be removed from a running partition only by the MCP executing in that partition, using the MCP's FREE command.

During that process all boxes remaining in the partition will have their system interface controls updated such that they ignore the outgoing box.

SOFT CONFIGURATION

-----

Soft configuration is a software function that allows control over how the hardware resources are partitioned into systems and how each system's memory is partitioned into address spaces. Since the B7900 hardware has much greater architectural flexibility than any previous machine, extensive changes to the soft configuration mechanism were required.

The SYMBOL/CONFIGURATION file is the human interface to the soft configuration mechanism. This file is created or modified to define the possible configurations of the hardware resources. The SYMBOL/CONFIGURATION file is converted into a SYSTEM/CONFIGURATION file by the utility program SYSTEM/CONFIGURATOR.

SYSTEM/CONFIGURATION is the MCP's interface to the soft configuration mechanism. This interface is established through the ODT CF command. Once these files have been created, and the SYSTEM/CONFIGURATION file has been attached through the CF command, actual reconfiguration is done through the ODT RECONFIGURE command. Reconfiguration can also be performed from the Loader by using the LOAD CONFIGURATION and RECONFIGURE commands. The syntax and semantics of the CF and RECONFIGURE commands are specified in the Operator Display Terminal (ODT) Reference Manual (Form # 5011687).

DATA COMMUNICATIONS

-----

The B7900 Data Communications Subsystem is substantially different at the hardware level from B7700 and B7800 (DCP) data communications. In place of the DCP and Adaptor Cluster modules, the B7900 uses the Network Support Processor (NSP)

and Line Support Processor (LSP) modules.

The NSP and the LSP are Data Link Processors (DLPs) that communicate with the system through the HDU. An LSP can support up to 16 full or half-duplex data comm lines, and an NSP can control up to eight LSPs with a maximum of 96 lines.

A Network Definition Language, called NDLLI, was developed for use with this new hardware. NDLLI is described in the B5000/B6000 Series NDLLI REFERENCE MANUAL (form no. 5011828). This document also describes the logical partitioning of data comm functions between the NSP and the LSP.

The SOURCENDLLI symbolic provides an example of how NDLLI programs are written. In order to provide a compatible interface to an MCS, certain NDLLI programming conventions must be followed. The SOURCENDLLI symbolic contains numerous comments explaining these conventions in detail.

In order to initialize an NSP, a NIF file and firmware file must be provided. A standard firmware file called FIRMWARE/NSP is provided. The NIF file is provided by the user by compiling a source NDLLI program, such as the example SYMBOL/SOURCENDLLI. The file name of the NIF must have "NIF" as the last node of its identifier. If <prefix>/NIF is the NIF file, the MCP creates an auxiliary file <prefix>/DCPCODE when the NSP is initialized with the NIF for the first time.

The ID (Initiate Datacom) ODT command is used to initialize an NSP, interrogate or alter the setting of various data comm options and direct commands to the DCCONTROL process.

One of the functions that can be initiated through the ID command is data comm auditing. A program, SYSTEM/DCAUDITOR, can be used to analyze the files produced during the auditing process.

A dump of the local memory of a specified NSP can be requested through the DUMP option of the ID command. The file produced can be analyzed with a program, SYSTEM/NSPDUMPANALYZER.

A report of the network configuration can be obtained through the DCSTATUS utility.

B5900 HARDWARE OVERVIEW

The B5900 central processing unit represents an architecture based on a series of fully microprogrammed function processors supported by a multi-bus structure. Despite the simplicity of its internal operation the B5900 is fully code compatible with all other large systems.

The processor is divided into five main areas that operate concurrently. These modules are serviced by two high-speed communication buses that are capable of simultaneous data transfer (see figure 1-21).

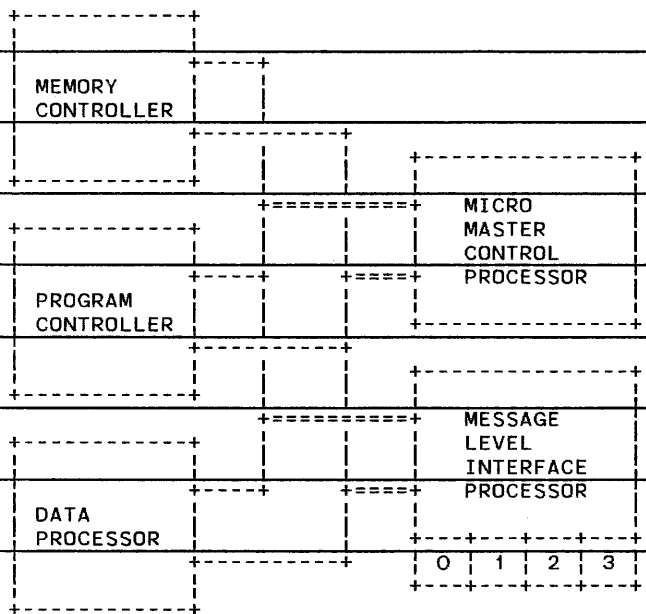


Figure 1-21. B5900 System

## BUS STRUCTURE

There are two major buses on the B5900, the Main Data Bus (M-Bus) and the Control Bus (C-Bus). The M-bus is 52-bits wide to allow the movement of one word (48 data-bits, 3 tag-bits and 1 parity-bit) through the system at high speed. The C-Bus is 30-bits wide and is used by the Micro Master Control processor to control the other modules in the CPU.

## MICRO MASTER CONTROL PROCESSOR

This is the master processor designed to execute the large system instruction set. It is a micro-programmed processor that guides the other processors through the execution of an instruction. This is done by addressing commands to a specific CPU component via the C-Bus.

## PROGRAM CONTROLLER

This processor is designed to independently decode instructions for the Micro Master Control processor as well as work directly with the Memory Controller to fetch additional program words.

## DATA PROCESSOR

This unit is responsible for the actual arithmetic and logical processing done on the B5900. It is fed a series of commands from the Micro Master Control processor via the C-Bus.

## MESSAGE LEVEL INTERFACE PROCESSOR

This unit provides an interface between the CPU and the DLPs actually doing the I/O. It will manage up to four MLI ports and is responsible for providing the DLPs with a high speed path to and from memory.

## MEMORY CONTROLLER

The Memory Controller is responsible for providing the system with access to up to 6.2 MB of memory. It also provides the pathway to GLOBAL(TM) Memory.

B6900 HARDWARE OVERVIEW

The Burroughs B6900 system is high speed computer system derived from the B6800 series architecture. It was the first large system to support Universal I/O (UIO) hardware. It provides reliable high speed processing with a relatively low environmental cost in either floorspace or power.

The system is divided into three major parts, each operating to provide independant services to maximize system throughput. A schematic of the B6900 is shown in figure 1-22.

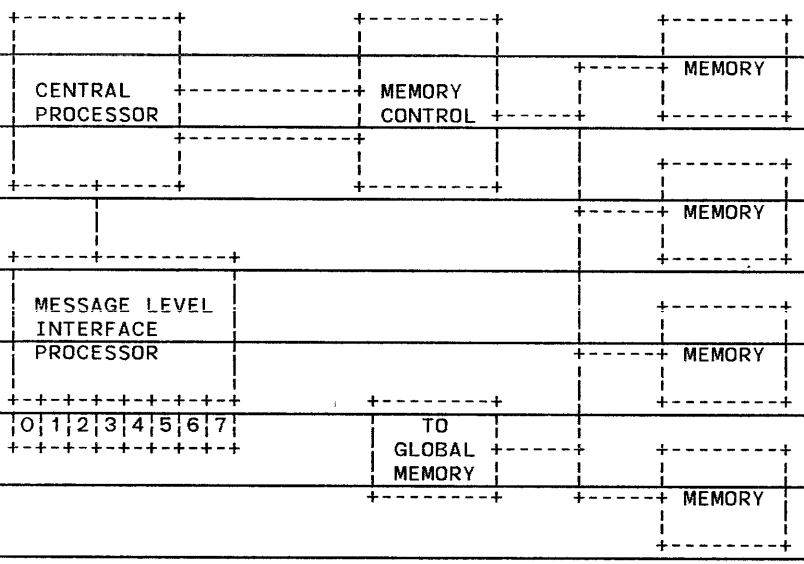


Figure 1-22. B6900 System

## CENTRAL PROCESSOR

The Central Processor is the unit responsible for all program and MCP code execution. It does all arithmetic and logical operators based on the operator strings it is directed to execute. The CPU itself is almost identical with the B6800 CPU. A two words top-of-stack buffer is maintained in the CPU to speed operation. Also there is an automatic 1-word code look-ahead that is done via the CPU's second port to the Memory Control. This second path is dedicated to the look-ahead function.

## MEMORY CONTROL

This unit provides the CPU with access to up to 6.2 MB of high speed memory. It acts as an exchange to allow either of the two CPU memory request ports with access to up to five memory modules. It is the fifth memory module that is the B6900's path to GLOBAL(TM) Memory.

## MESSAGE LEVEL INTERFACE PROCESSOR

This processor is responsible for all the management of I/O done to and from the DLPs. It handles all main-memory queue manipulation as well as provides a path to memory for the DLP when data transfer is required. The MLIP must share the CPU's data path to memory, however, it is designed to work in a "burst" mode so as to make the memory accessing as efficient as possible. The MLIP is capable of handling up to 8 MLI ports. All input/output, both peripheral and datacom, is managed by the MLIP independantly of the CPU.



## B5900/B6900 I/O OVERVIEW

-----

This section deals with the technical aspects of the B5900/B6900 Universal Input Output (UIO) hardware operation. The details of both the hardware and software operation of UIO are extremely well covered in the manual "OVERVIEW OF PHYSICAL I/O ON B5900/B6900 SYSTEMS" (form #5013758). This section will simply review the main concepts behind UIO.

### MAIN COMPONENTS

-----

Below are listed the main components of the UIO system hardware:

1. Message Level Interface Processor - That part of a B5900/B6900 mainframe that is given the task of managing the I/O by the central processor. Its duties will include I/O queue handling, path queue handling, data transfer to and from memory, DLP initiation and generating an interrupt to flag the CPU to handle all queued results.
2. Message Level Interface - A standard hardware/software protocol to allow the development of an I/O subsystem that is independant of the characteristics of a given mainframe.
3. Line Expansion Module - A device to allow a Message Level Interface port to connect to up to 7 I/O Bases rather than just one.
4. I/O Base - A physical card cage to house up to 8 DLPs.
5. Data Link Processor - A specialized microprocessor with sufficient intelligence and storage to complete 1 entire I/O to a given peripheral. DLPs are designed to interface to specific peripheral devices and represent the connection port of the mainframe to the peripheral.

### I/O INITIATION

-----

The general flow for I/O on a UIO system is as follows:

1. The CPU formats an IOCB (I/O Control Block). This data structure totally defines the nature of the I/O to be done including unit number and location, I/O control bits, buffer location, I/O queue locations and reserved space for I/O results and times.

2. The processor executes the CUIO operator passing it a descriptor to the IOCB.
3. The MLIP links the IOCB into the unit's command queue.
4. If this is the first I/O in the command queue then the unit's DLP is checked to see if the I/O can be initiated. If it can, the IOCB is delinked from the command queue and the DLP is passed the IO to initiate.
5. If the unit is sharing a path with other units and the DLP is busy then the MLIP will link the unit's entire command queue into the specified horizontal queue. The command queue will remain in this state until some other unit on the same path completes its command queue, freeing the DLP.
6. When the MLIP senses that an I/O has completed, it links the IOCB into the designated result queue. If the interrupt bit is on in the IOCB's MLIP control word, or if an I/O exception occurred or if the command queue is empty then the CPU will be given an I/O finish interrupt. If the command queue is empty then the MLIP will also check the unit's horizontal queue to see if there is another command queue queued for activation.
7. Upon receipt of an I/O finish interrupt the CPU will process the entire result queue. For each IOCB the following will be done:
  - delink from result queue
  - perform exception handling if necessary
  - perform all statistical accounting
  - wake up any task waiting on the I/O's completion

TABLE OF CONTENTS		
1		1
2		2
3		3
4	1 - 1	4
5		5
6	INTRODUCTION . . . . .	1 - 1
7		7
8	B7800 HARDWARE OVERVIEW . . . . .	1 - 1
9		9
10	GENERAL . . . . .	1 - 1
11		11
12	SYSTEM CONFIGURATION . . . . .	1 - 4
13		13
14	HARDWARE REVIEW . . . . .	1 - 4
15		15
16	MEMORY . . . . .	1 - 8
17		17
18	CPM . . . . .	1 - 13
19		19
20	IOM . . . . .	1 - 21
21		21
22	B7800 TIGHTLY COUPLED SYSTEMS . . . . .	1 - 34
23		23
24	B7900 HARDWARE OVERVIEW . . . . .	1 - 36
25		25
26	INTRODUCTION . . . . .	1 - 36
27		27
28	CENTRAL PROCESSING MODULE (CPM) . . . . .	1 - 37
29		29
30	MEMORY SUBSYSTEM MODULE (MSM) . . . . .	1 - 39
31		31
32	AUXILIARY PROCESSOR (AP/AMP) . . . . .	1 - 42
33		33
34	HOST DATA UNIT (HDU) . . . . .	1 - 44
35		35
36	DLP-BASED I/O SUBSYSTEM COMPONENTS . . . . .	1 - 46
37		37
38	DEVICE NAMING . . . . .	1 - 50
39		39
40	AMP DEVICE NAMING . . . . .	1 - 51
41		41
42	I/O RECONFIGURATION . . . . .	1 - 51
43		43
44	I/O UNIT/DLP/MLI LOAD BALANCING . . . . .	1 - 52
45		45
46	I/O TIME . . . . .	1 - 53
47		47
48	MAINTENANCE HARDWARE . . . . .	1 - 53
49		49
50	MAINFRAME MAINTENANCE . . . . .	1 - 54
51		51
52	I/O SUBSYSTEM MAINTENANCE . . . . .	1 - 54
53		53
54	PARTITIONING . . . . .	1 - 55
55		55
56		56
57		57
58		58
59		59
60		60

1			1
2	SOFT CONFIGURATION. . . . .	1 - 56	2
3			3
4	DATA COMMUNICATIONS . . . . .	1 - 56	4
5			5
6	B5900 HARDWARE OVERVIEW . . . . .	1 - 58	6
7			7
8	B6900 HARDWARE OVERVIEW . . . . .	1 - 61	8
9			9
10	B5900/B6900 I/O OVERVIEW. . . . .	1 - 64	10
11			11
12	MAIN COMPONENTS . . . . .	1 - 64	12
13			13
14	I/O INITIATION. . . . .	1 - 64	14
15			15
16			16
17			17
18			18
19			19
20			20
21			21
22			22
23			23
24			24
25			25
26			26
27			27
28			28
29			29
30			30
31			31
32			32
33			33
34			34
35			35
36			36
37			37
38			38
39			39
40			40
41			41
42			42
43			43
44			44
45			45
46			46
47			47
48			48
49			49
50			50
51			51
52			52
53			53
54			54
55			55
56			56
57			57
58			58
59			59
60			60

TABLE OF ILLUSTRATIONS		
1		
2		
3	Figure 1-1. B7800 Exchange. . . . .	1 - 3
4	Figure 1-2. B7800 SYSTEM. . . . .	1 - 5
5	Figure 1-3. MCM Configuration and Planar Memory . . . . .	1 - 11
6	Figure 1-4. MODEL III MCM Block Diagram (IC Memory) . . . . .	1 - 12
7	Figure 1-5. CPM Block Diagram . . . . .	1 - 15
8	Figure 1-6. CODE BUFFER (Associative Memory). . . . .	1 - 17
9	Figure 1-7. DATA BUFFER (Associative Memory) (1 of 2) . . . . .	1 - 19
10	Figure 1-7. DATA BUFFER (Associative Memory) (2 of 2) . . . . .	1 - 20
11	Figure 1-8. IOM Block Diagram . . . . .	1 - 23
12	Figure 1-9. IOM Structures. . . . .	1 - 26
13	Figure 1-10. START I/O (PAGE 1 OF 2). . . . .	1 - 29
14	Figure 1-11. START I/O (PAGE 2 OF 2). . . . .	1 - 30
15	Figure 1-12. TERMINATE I/O (PAGE 1 OF 3). . . . .	1 - 31
16	Figure 1-13. TERMINATE I/O (PAGE 2 OF 3). . . . .	1 - 32
17	Figure 1-14. TERMINATE I/O (PAGE 3 OF 3). . . . .	1 - 33
18	Figure 1-15. B7900 Central Processor Module (CPM) . . . . .	1 - 38
19	Figure 1-16. B7900 Memory Subsystem Module (MSM). . . . .	1 - 40
20	Figure 1-17. MSM Translation Table. . . . .	1 - 41
21	Figure 1-18. Auxiliary Processor (AP/AMP) . . . . .	1 - 43
22	Figure 1-19. Host Data Unit (HDU) . . . . .	1 - 45
23	Figure 1-20. UID Subsystem Components . . . . .	1 - 47
24	Figure 1-21. B5900 System . . . . .	1 - 59
25	Figure 1-22. B6900 System . . . . .	1 - 62
26		
27		
28		
29		
30		
31		
32		
33		
34		
35		
36		
37		
38		
39		
40		
41		
42		
43		
44		
45		
46		
47		
48		
49		
50		
51		
52		
53		
54		
55		
56		
57		
58		
59		
60		

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	CODE FILE INFORMATION	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55	APPENDIX B	55
56		56
57		57
58		58
59		59
60		60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1									1
2									2
3									3
4									4
5									5
6									6
7									7
8									8
9									9
10									10
11									11
12									12
13									13
14	COMPILERS	Large Systems Code File Format 3.5.0 Release					00010200		14
15	BINDER						00010205		15
16	MCP				06-16-75		00010400		16
17									17
18					10-13-77 Rev		00010410		18
19		Nick Smilonich, Burroughs, CSG					00010500		19
20									20
21									21
22									22
23									23
24									24
25							00010540		25
26									26
27									27
28									28
29									29
30									30
31									31
32									32
33									33
34									34
35									35
36									36
37									37
38									38
39									39
40									40
41									41
42									42
43									43
44									44
45									45
46									46
47									47
48									48
49									49
50									50
51									51
52									52
53									53
54									54
55									55
56									56
57									57
58									58
59									59
60									60

1				1
2				2
3				3
4	<u>TABLE OF CONTENTS</u>			4
5				5
6	INTRODUCTION. . . . .	2	00010555	6
7	OPERATING SYSTEM INTERFACE. . . . .	2	00013400	7
8	GENERAL CODE FILE LAYOUT. . . . .	3	00017600	8
9	SEGMENT ZERO LAYOUT. . . . .	5	00021100	9
10	SEGMENT DICTIONARY (D1 STACK) . . . . .	13	00038900	10
11	FILE PARAMETER BLOCKS (FPBS). . . . .	15	00043100	11
12	COMPILER/INTRINSIC INTERFACE. . . . .	17	00052100	12
13	CODE FILE BINDER INFORMATION. . . . .	20	00060900	13
14	BINDER CONVENTIONS . . . . .	20	00062500	14
15	PROGRAM DESCRIPTION. . . . .	21	00065900	15
16	PROCEDURE DIRECTORY. . . . .	23	00075000	16
17	LOCAL DIRECTORY. . . . .	26	00084000	17
18	REFERENCES. . . . .	27	00087500	18
19	FPB/PPB RUN-TIME INFORMATION. . . . .	28	00089700	19
20			00094700	20
21	APPENDIX A Supplementary Binder Information. . . . .	30	00094700	21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1				1
2				2
3				3
4	<u>TABLE OF ILLUSTRATIONS</u>			4
5				5
6	GENERAL CODE FILE LAYOUT. . . . .	3	00017908	6
7	SEGMENT ZERO EXAMPLE. . . . .	10	00032153	7
8	COMPILERINFO EXAMPLE. . . . .	11	00034100	8
9	SEGMENT DICTIONARY EXAMPLE. . . . .	14	00040900	9
10	FPB EXAMPLE . . . . .	17	00050000	10
11	BEFORE FIRST CALL ON INTRINSIC. . . . .	18	00055300	11
12	AFTER FIRST CALL ON INTRINSIC . . . . .	18	00057700	12
13	PROGRAM DESCRIPTION EXAMPLE . . . . .	23	00073100	13
14	PROCEDURE DIRECTORY FORMAT. . . . .	24	00076100	14
15	TYPICAL ENTRY. . . . .	25	00080700	15
16	PROCEDURE DIRECTORY EXAMPLE . . . . .	26	00082110	16
17	LOCAL DIRECTORY EXAMPLE . . . . .	27	00095500	17
18	PPB EXAMPLE . . . . .	29	00092800	18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1	2	Feature - Large Systems Code File Format For 3.5.0 Release	00010504	1
2				2
3				3
4				4
5				5
6		INTRODUCTION	00010555	6
7		-----	-----	7
8				8
9		This document describes the format and content of the code file	00010800	9
10		objects, as emitted by the Large Systems compilers. The information	00010900	10
11		was derived from the WRAPUP procedures in the compilers.	00011000	11
12		SYMBOL/BINDER and the MCP global section. Where applicable,	00011100	12
13		references to pertinent sequence numbers and symbolics are noted.	00011300	13
14				14
15		The description is organized as follows:	00011500	15
16				16
17		OPERATING SYSTEM INTERFACE	00011600	17
18		CODE FILE DECLARATION	00011700	18
19		GENERAL CODE FILE LAYOUT	00011800	19
20		SEGMENT ZERO LAYOUT WORD AND COMPILERINFO	00011900	20
21		SEGMENT DICTIONARY	00012000	21
22		FILE PARAMETER BLOCKS (FPBS)	00012100	22
23		COMPILER/INTRINSIC INTERFACE	00012200	23
24		CODE FILE BINDER INFORMATION	00012300	24
25		BINDER CONVENTIONS	00012500	25
26		PROGRAM DESCRIPTION	00012600	26
27		PROCEDURE DIRECTORY	00012700	27
28		LOCAL DIRECTORY REFERENCES	00012800	28
29		FPB/PPB RUN-TIME INFORMATION	00013000	29
30				30
31		APPENDIX A - SUPPLEMENTARY BINDER INFORMATION	00013200	31
32				32
33				33
34		OPERATING SYSTEM INTERFACE	00013400	34
35		-----	-----	35
36				36
37		Large Systems compilers are declared as real procedures with one	00013700	37
38		array parameter.	00013800	38
39				39
40		REAL PROCEDURE ALGOLCOMPILER (SHEET); ARRAY SHEET [*];	00013900	40
41				41
42		The SHEET array is passed in from the MCP, and contains a skeleton	00014200	42
43		segment zero. This skeleton will be used by the compiler in	00014300	43
44		constructing the real segment zero for the compiled code file. The	00014400	44
45		SHEET passed contains values from WFL control cards (e.g.,	00014500	45
46		PRIORITY=75, SHEET [9]=75) and file-equation control information	00014600	46
47		(e.g., ?FILE LINE(KIND=REMOTE)). SHEET [0] contains the offset in	00014700	47
48		the SHEET to the variable (file-equation and program parameter)	00014800	48
49		information.	00014900	49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1		3		1
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3				3
4	CODE FILE DECLARATION		00015000	4
5				5
6	Code files are declared in the compilers as disk files with a		00015300	6
7	MAXRECSIZE of 30 words (1 segment).		00015400	7
8				8
9	A typical declaration is:		00015600	9
10				10
11	FILE CODE (MAXRECSIZE=30,		00015700	11
12	BLOCKSIZE=270,		00015900	12
13	EXTMODE=SINGLE,		00016000	13
14	FLEXIBLE=TRUE,		00016100	14
15	KIND=PACK, PACKNAME="DISK.",		00016200	15
16	MYUSE=OUT,		00016300	16
17	AREAS=40,		00016400	17
18	AREASIZE=504,		00016500	18
19	BUFFERS=2,		00016600	19
20	SAVEFACTOR=999);		00016700	20
21				21
22	In addition, these attributes and certain others are checked at		00016725	22
23	compiler initialization to prevent incorrect or mischievous		00016730	23
24	file-equation.		00016800	24
25				25
26	The AREASIZE attribute is defined as CHUNK in some of the compilers		00017000	26
27	and represents the disk row size. The compilers use this to ensure		00017100	27
28	code segments are kept within a single disk row (to avoid problems		00017200	28
29	with PRESENCEBIT).		00017500	29
30				30
31				31
32	GENERAL CODE FILE LAYOUT		00017600	32
33	-----		-----	33
34				34
35	The following chart illustrates the contents of the code		00017802	35
36				36
37	The order of the various parts of the code file is not fixed and		00017806	37
38	varies between compilers. Only segment zero is fixed. file, which		00017900	38
39	will be described herein.		00017905	39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	4	Feature - Large Systems Code File Format For 3.5.0 Release	00010504	.FLAG 52	1
2					2
3					3
4					4
5		GENERAL CODE FILE LAYOUT	00019601		5
6			00019602		6
7		SEGMENT ZERO	00019603		7
8		Directory of code file	00019604		8
9		and accumulator of	00019605		9
10		run-time information	00019606		10
11			00019607		11
12		FPB	00019608		12
13		(file parameter	00019609		13
14		block)	00019610		14
15			00019611		15
16			00019612		16
17		OBJECT CODE	00019613		17
18		Actual object code to be executed	00019614		18
19			00019615		19
20			00019616		20
21		OBJECT CODE	00019617		21
22		REFERENCES	00019618		22
23		Object code. The code segment	00019619		23
24		may also contain references	00019620		24
25		(code markers for binding)	00019621		25
26			00019622		26
27		LOCAL	00019623		27
28		DIRECTORY	00019624		28
29		Description of local variables	00019625		29
30		for binding	00019626		30
31			00019627		31
32		PROCEDURE	00019628		32
33		DIRECTORY	00019629		33
34		List of procedures	00019630		34
35		(blocks in the program)	00019631		35
36		used for binding	00019632		36
37			00019633		37
38			00019634		38
39			00019635		39
40		VALUE ARRAY	00019636		40
41		Preset data arrays, pools	00019637		41
42			00019638		42
43			00019639		43
44			00019640		44
45		PROGRAM	00019641		45
46		DESCRIPTION	00019642		46
47		Directory of binding information	00019643		47
48		in code file	00019644		48
49			00019645		49
50			00019646		50
51		D1 STACK	00019647		51
52		IMAGE	00019648		52
53		Segment dictionary or D[1]	00019649		53
54		stack image in 64-bit format	00019650		54
55			00019651		55
56			00019652		56
57					57
58					58
59					59
60					60

1					5					1
2	Feature - Large Systems Code File Format For 3.5.0 Release								000i0504	2
3										3
4	SEGMENT ZERO LAYOUT								00021100	4
5	-----								-----	5
6										6
7	This section outlines the content of each word in segment zero and								00021400	7
8	indicates the portion to be built by the compiler. The MCP defines								00021500	8
9	at 05043000 describe the segment words.								00022000	9
10										10
11										11
12			COMPILER						00022455	12
13	WORD	DEFINITION	SUPPLIED	FORMAT	COMMENTS				00022460	13
14		-----	-----	-----	-----				00022465	14
15	0				Displacement in the				00022600	15
16					SHEET of PPB/FPB				00022620	16
17					run-time information,				00022640	17
18					to be written by				00022660	18
19					the compiler into				00022680	19
20					the code file. This				00022700	20
21					location is zeroed				00022720	21
22					by the compiler.				00022740	22
23									00022900	23
24	1	LIBWORD	YES		Information concerning				00023000	24
25					the Library				00023020	25
26					SHARINGCLASS of the				00023040	26
27					codefile.				00023060	27
28									00023100	28
29	2	ENTRYPOINT	YES	Integer	Stack address of the				00023200	29
30					outer block PCW				00023300	30
31					in the segment				00023400	31
32					dictionary. Used				00023450	32
33					by the MCP to				00023500	33
34					construct a phony RCW				00023550	34
35					which allows NORMALBOJ				00023600	35
36					to exit into the JOB.				00023650	36
37									00023700	37
38	3	ROLLOUT	YES	Integer	Last code file record				00023800	38
39					(segment) number;				00023850	39
40					i.e., size of code				00023900	40
41					file. Used for				00023920	41
42					RESTART. See DOCTOR				00023940	42
43					at 35197810. VERSION				00024000	43
44					for MCP.				00024003	44
45									00024100	45
46	4	DATE	YES	Integer	Set to TIME(5) at				00024200	46
47					at compile wrapup.				00024300	47
48									00024400	48
49	5	TIME	YES	Integer	Set to TIME(11)				00024500	49
50					at compile wrapup.				00024520	50
51										51
52										52
53										53
54										54
55										55
56										56
57										57
58										58
59										59
60										60

*copy file*

1	6									1	
2		Feature - Large Systems Code File Format For 3.5.0 Release							00010504		2
3										3	
4										4	
5										5	
6			COMPILER					00024580		6	
7		WORD	DEFINITION	SUPPLIED	FORMAT		COMMENTS	00024600		7	
8		---	---	---	---		---	00024620		8	
9		6	SOEXECUTEINFO	YES	Word			00024800		9	
10			NONEXECUTABLEF		[47:1]		1 means this code file	00024804		10	
11							is not executable.	00024806		11	
12								00024808		12	
13			NONEXECUNSAFEF	YES	[45:1]		If NONEXECUTABLEF is	00024810		13	
14							1, then this field	00024812		14	
15							indicates why the code	00024814		15	
16							file is nonexecutable.	00024816		16	
17							A 1 means the file is	00024818		17	
18							not executable because	00024820		18	
19							it contains unsafe	00024822		19	
20							code. A 0 is used in	00024824		20	
21							all other cases.	00024826		21	
22								00024828		22	
23			NONEXECMACHINEF	YES	[44:19]		Families not supported	00024830		23	
24							by this code file.	00024832		24	
25							For each machine,	00024834		25	
26							0 (false) means that	00024836		26	
27							the code file will run	00024838		27	
28							on the machine, and	00024840		28	
29							1 (true) means it will	00024842		29	
30							not. The bits in the	00024844		30	
31							field are indexed	00024846		31	
32							with the following,	00024848		32	
33							where 0 implies	00024850		33	
34							(bit 26).	00024852		34	
35										35	
36			B6800ID	=	2			00024856		36	
37			B6900ID	=	3			00024858		37	
38			B5900ID	=	4			00024860		38	
39			A3ID	=	5			00024862		39	
40			A9ID	=	6			00024864		40	
41			A13ID	=	7			00024866		41	
42			B7700ID	=	11			00024868		42	
43			B7800ID	=	12			00024870		43	
44			B7000ID	=	13			00024872		44	
45			B7900ID	=	14			00024874		45	
46										46	
47										47	
48										48	
49										49	
50										50	
51										51	
52										52	
53										53	
54										54	
55										55	
56										56	
57										57	
58										58	
59										59	
60										60	

*Keeps heap programs from running!*

*MS*

WORD	DEFINITION	COMPILER SUPPLIED	FORMAT	COMMENTS	
Feature - Large Systems Code File Format For 3.5.0 Release					00010504
					00024880
					00024882
					00024884
	PARAMCOUNTF	YES	[19:20]	Number of formal parameters.	00024886
					00024888
			[46:1]	Unused.	00024890
					00024892
					00024894
	PARAMCOUNT	YES	Pars	Number of parameters to the main program.	00024900
					00025000
			Nonexec		00025100
			[47:1]	NONEXEC=1 means this code file is nonexecutable as a unit.	00025300
					00025400
					00025420
					00025440
					00025500
7	CORRESTIMATE	YES	Integer	If SHEET [7] is not zero, use supplied core estimate (from ?CORE card); else insert compiler generated core estimate.	00025600
					00025700
					00025720
					00025740
					00025800
					00025850
					00025900
					00025950
8	COMPILERINFO	YES		See section on COMPILERINFO	00026000
					00026050
					00026100
9	PRIORITY			MCP supplied from PRIORITY card	00026200
					00026250
					00026300
10	MAXPROCESSTIME			MCP supplied from PROCESSTIME card	00026400
					00026450
					00026500
11	MAXIOTIME			MCP supplied from IOTIME card	00026600
					00026650
					00026700
12	FPB	YES	Record number & length= [39:20]	If SHEET[0] NEQ 0, a file parameter or program parameter block is passed in the SHEET starting at [SHEET[0]] If it exists, the compiler writes it out to the code file. See DOCTOR at 35197600.	00026800
					00027000
					00027200
					00027220
					00027240
					00027260
					00027280
					00027300
					00027320
					00027340
					00027360
					00027500
					00027600

*30% code  
70% data*

*compiler's queue used for scheduling*

WORD	DEFINITION	COMPILER SUPPLIED	FORMAT	COMMENTS	
8 Feature - Large Systems Code File Format For 3.5.0 Release					00010504
13	OPTION			Standard MCP option word. See MCP at 01305000 for format.	00027840 00027860 00027880 00027884 00027886 00027888
14	SODEIMPLEMENTINFO	YES	Word	Miscellaneous deimplementation warnings will be assigned specific bits. The appropriate bits will either be set by the MCP or by a compiler.	00027900 00028000 00028100 00028102 00028104 00028106 00028108 00028110 00028112
This word will be used only by 3.7 compilers, 3.5, 3.6 and 3.7 MCP, and 3.5, 3.6 and 3.7 Binder.					00028130 00028140
This word will become available starting 4 releases after 3.7 (currently estimated to be 4.1).					00028150 00028160 00028170
15	STACKSIZE	YES	Integer	Stack estimate. If SHEET [15] EQL 0, use 512 else SHEET [15].	00028300 00028320 00028350 00028360 00028400
16	PROGRAM DESCRIPTION	YES	Record number & length [39:20]	Used with paramcount for parameter checking (see external reference of DOCTOR). Also used for binding (see section on binder information).	00028450 00028454 00028510 00028555 00028600 00028650 00028660 00028680 00028690
17	INTRINSICTABLE			For intrinsics code files. Points to "INTRINSICINFO" array.	00028700 00028702 00028705
18	D1DESC	YES	Record number & length [39:20]	Address in code file of 64-bit D1 stack (segment dictionary) image. Length is number of 64-bit tagged words (see section on segment dictionary).	00028750 00028800 00028860 00028900 00028950 00028960 00028980 00029000 00029002 00029004 00029006 00029010 00029150

WORD	DEFINITION	COMPILER SUPPLIED	FORMAT	COMMENTS	
Feature - Large Systems Code File Format For 3.5.0 Release					00010504
19	LOCALCOUNT	YES	Integer	Size of D2 stack allocated by compilers. FORTRAN and PL/I includes the D3 size too.	00029180 00029200 00029220 00029240 00029250 00029260 00029270 00029280 00029290
20	WSLIMIT			Working set core limit.	00029300 00029350
21	DIESTIMATE			Working set code estimate	00029500 00029600
22	RESTARTCOUNT			Checkpoint/restart use (?RESTART card)	00029700 00029800
23	EXECUTION COUNT				00029900
24	D1WSAVRG				00030000 00030100
25	RUNWSAVRG			Used for working set statistics, average amount of core and data. See UPDATECOREEST in MCP at 35370000.	00030200 00030300 00030400 00030600
26	LOGSTART			For restart. See DOCTOR at 35147910	00030700 00030800
27	SKELSTACK			For restart. See DOCTOR at 35197825	00030900 00031000
28	PROGRAMUNIT	YES	Record number & length [39:20]	Points to a Program Unit Directory if the code file contains one or more program units compiled under the LIBRARY compiler control option.	00031100 00031104 00031110 00031114 00031120 00031122 00031125 00031126 00031128 00031130 00031200



1	10	Feature - Large Systems Code File Format For 3.5.0 Release			00010504	1	
2						2	
3						3	
4		WORD	DEFINITION	COMPILER SUPPLIED	FORMAT	COMMENTS	4
5		----	-----	-----	-----	-----	5
6							6
7	29	SYMBOLICDESC	YES	Record number & length [39:20]		Points to	7
8						SYMBOLICINFO. Used	8
9						automatic separate	9
10						compilation and	10
11						binding (ALGOL	11
12						code files only).	12
13							13
14							14
15							15
16							16
17		SEGMENT ZERO EXAMPLE			00032153		17
18							18
19							19
20		SEGMENT NO. 000			00032410		20
21							21
22	0	000000000000	000000000000	000000000002*			22
23	3	00000000000C	0000050431C5	0004E38B7F8C			23
24	6						24
25	9	000000000000	0000000005A7	000000020600			25
26		000000000000	000000000000	000000000000			26
27							27
28	12	00000060000B	000000000000	000000000000			28
29	15	000000000200	000001200009**	000000000000			29
30							30
31	18	00000060000A***	000000000009	000000000000			31
32	21	000000000000	000000000000	000000000000			32
33							33
34	24	000000000000	000000000000	000000000000			34
35	27	000000000000	000000000000	000000000000			35
36							36
37	*	ENTRYPOINT AT D1[2]			00033700		37
38	**	PROGRAM DESCRIPTION AT RECORD # 9, 18 WORDS LONG			00033800		38
39	***	SEGMENT DICTIONARY AT RECORD # A, 6 WORDS LONG			00033900		39
40							40
41							41
42							42
43							43
44							44
45							45
46							46
47							47
48							48
49							49
50							50
51							51
52							52
53							53
54							54
55							55
56							56
57							57
58							58
59							59
60							60

1		11		
2	Feature - Large Systems Code File Format For 3.5.0 Release			00010504
3				
4	COMPILERINFO EXAMPLE			00034100
5	-----			
6				
7				00037301
8				00037302
9				00037303
10				00037304
11	47	46	45	44
12	43	42	41	40
13	31	23	15	9
14	LANGUAGEF		RELEASENUM	UNUSED
15			CYCLENUM	
16				00037342
17				00037344
18				00037346
19				00037348
20				00037360
21				00037380
22				00037390
23				00037400
24				00037420
25				00037440
26				00037460
27				00037480
28				00037500
29				00037520
30				00037540
31				00037560
32				00037580
33				00037600
34				
35	The version 31.234.567 would be store RELEASENUM = 31.			00037604
36	CYCLENUM = 234.			00037606
37				
38	The unused fields contain zeros.			00037620
39				
40				
41				
42				
43	LANGUAGEF			00037800
44				
45				
46	ALGOL	= 0	JOVIAL	= 5
47	COBOL	= 1	BETA	= 6
48	FORTRAN	= 2	ESPOL	= 7
49	XALGOL	= 3	DCALGOL	= 8
50	PL/1	= 4	BASIC	= 9
51			WFL	= 10
52				00038000
53				00038100
54				00038200
55				00038300
56				00038400
57				00038450
58				
59				
60				

-----  
 -These fields contain the  
 version of the compiler that  
 compiled the codefile.

-----Library capable

PP-----Privileged program ← avoid usacode security

-----Not used

-----DMS Capable

CP-----Control Program ← means program will always run  
 with ~~wait~~ in schedule

-----Not used

-----IPC Capable

1	12			1
2		Feature - Large Systems Code File Format For 3.5.0 Release	00010504	2
3				3
4		The defines for the COMPILERINFO word are at 05050500, 05034030 in	00038700	4
5		the MCP.	00038750	5
6				6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1		13		1
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3				3
4				4
5				5
6	SEGMENT DICTIONARY (D1 STACK)		00038900	6
7	-----		-----	7
8				8
9	The segment dictionary generated by the compilers is written to the		00039200	9
10	code file in a special format. Each stack entry is written as a		00039300	10
11	64-bit word. The purpose of this technique is to allow the MCP to		00039400	11
12	perform a "read with tags", when setting up the D1 stack. The hex		00039500	12
13	format of each entry is:		00039600	13
14				14
15	AT 00 DDDD DDDD DDDD		00039700	15
16	T = TAG		00039800	16
17	D = DESCRIPTOR BODY OR WORD		00039900	17
18	A = BINDER INFORMATION FIELD		00039950	18
19				19
20	The first two 64-bit entries in the stack image are zero, reserving		00040200	20
21	space for MSCW/RCW. Segment zero [18] identifies the starting record		00040300	21
22	number [19:20] and the length in 64-bit words [39:20]. The PCW (TAG		00040400	22
23	7) for the outer block (i.e., the starting point of the program) is		00040500	23
24	generally one of the latter entries in the segment dictionary. The		00040600	24
25	D1 stack address of this PCW is stored in segment zero [2].		00040705	25
26				26
27	The high order bit of field A is used as an indication bit for the		00040720	27
28	Binder. Normally, A = 4"0". However, the high order bit of field A		00040725	28
29	is set to 1 in the following two situations.		00040735	29
30				30
31	First, in a program that allows or requires an external procedure to		00040745	31
32	be bound in, the high order bit of field A is set to 1 in the entry		00040750	32
33	for the data descriptor (tag 5) that points to the description of how		00040755	33
34	the external procedure and its parameters, if any, were declared.		00040760	34
35				35
36	Second, in a separately-compiled procedure that requires access to		00040770	36
37	global variables, the high order bit of field A is set to 1 in the		00040775	37
38	entry for any segment descriptors (tag 3) for code segments that		00040780	38
39	contain references (NAMCs and VALCs) to the global variables.		00040790	39
40				40
41	The Binder may also preserve the value of field A after binding the		00040800	41
42	external procedures to the host so that replacement binding can be		00040805	42
43	used.		00040890	43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	14	Feature - Large Systems Code File Format For 3.5.0 Release	00010504	1
2				2
3				3
4		SEGMENT DICTIONARY EXAMPLE	00040900	4
5		-----	-----	5
6				6
7		SEGMENT NO. 00A	00040993	7
8			00040998	8
9			00041001	9
10		ENTRYPOINT	00041003	10
11		PCW	00041005	11
12		TAG 7	00041010	12
13			00041020	13
14		ZERO	00041100	14
15		ZERO	00041183	15
16		SEGMENT	00041184	16
17		DESCRIPTOR	00041185	17
18		TAG 3	00041188	18
19			00041190	19
20			00041200	20
21			00041300	21
22		READ ONLY	00041381	22
23		VALUE ARRAY	00041383	23
24		TAG 5	00041385	24
25			00041388	25
26			00041390	26
27		0000003005001 080000840007 000000100004 000000E00002	00041400	27
28		000A03600003 000803800003	00041500	28
29			00041600	29
30		000001900001 000203508003 000001200001 000004000003	00041700	30
31		000001500001 000000000009	00041800	31
32			00041900	32
33		B280ABB7A080 32402EBOOF50 03AFABB51005 B180875005B9	00042000	33
34		B327108BA180 00AE400AABA3	00042100	34
35			00042200	35
36		6000AF960DBE 270000840001 B20595B4BEFF 270000740002	00042300	36
37		B20595B4B0BE 000001E00000	00042400	37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

stack = 0  
 PSI = 0  
 PWI = 36  
 L = 2  
 SDU = 1  
 SDF = 3

Leftover  
 trash

1		15		1
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3				3
4				4
5				5
6				6
7				7
8				8
9	FILE PARAMETER BLOCKS (FPBS)		00043100	9
10	-----		-----	10
11				11
12	FPBS are collections of attributes describing files declared in the		00043400	12
13	user program. The FPB will eventually be converted to a FIB when a		00043500	13
14	file is opened.		00043600	14
15				15
16	FPBS are generated and written to the code file in the following		00043800	16
17	manner:		00043900	17
18				18
19	WORD 0	LANGUAGENUMBER (see COMPILERINFO)	00044000	19
20		& FILELEVEL [47:8] (software)	00044100	20
21			00044200	21
22	WORD 1-N	INTNAME of the file in the form (HEX):	00044300	22
23		TT 01 01 LL NN ... NN 00 ... 00	00044400	23
24			00044500	24
25			00044600	25
26		TT = Total length (bytes)	00044700	26
27		LL = Length of name	00044800	27
28		NN = Name	00044900	28
29		00 = Padded to word boundary	00045000	29
30			00045100	30
31	WORD N+1	List of attributes in hex form:	00045200	31
32		Numeric/Boolean attributes	00045300	32
33		LL AA VV .. VV	00045400	33
34			00045500	34
35		LL = Length in bytes	00045600	35
36		AA = Attribute number	00045700	36
37		VV = Attribute value	00045800	37
38			00045900	38
39		String attributes	00046000	39
40		02 AA LL VV .. VV	00046100	40
41			00046200	41
42		AA = ATTRIBUTE NUMBER	00046300	42
43		LL = Length	00046400	43
44		VV = Attribute value	00046500	44
45			00046515	45
46		Padding, used to adjust to word boundary	00046525	46
47		for FPB's with variable entries filled	00046530	47
48		in at run time.	00046535	48
49		01	00046540	49
50			00046545	50
51		Stopper, the last valid byte in the FPB.	00046550	51
52		00	00046555	52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	16				1
2		Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3					3
4		Reference to the FPB in the code file is via a special descriptor		00047200	4
5		allocated for the file in the user stack (e.g., at D2 or higher).		00047300	5
6		This stack descriptor is allocated at the level of the file		00047400	6
7		declaration, and is constructed within the stack building code. The		00047500	7
8		stack location will eventually be changed to a descriptor pointing to		00047600	8
9		the FIB in memory. The format of the FPB descriptor is:		00047800	9
10					10
11		[45:1] = 1 Indexed bit		00047900	11
12		[42:3] = 7 Invalid SZF field		00048000	12
13		[35:16] Length of FPB (words)		00048100	13
14		[18:1] = 1 MCP value array bit		00048200	14
15		[17:18] Record number of FPB		00048300	15
16		[TAG] = 5		00048310	16
17					17
18		A typical descriptor in HEX would look like the following:		00048800	18
19					19
20		5 270000440001		00048900	20
21					21
22		The indexed/byte (SZF) fields are invalid and will cause a hardware		00049200	22
23		interrupt when the descriptor is accessed. The interrupt will force		00049300	23
24		a FIB to be constructed (GETFPB, ATTRIBUTEHANDLER in the MCP), and		00049400	24
25		the descriptor will be overwritten by a legitimate one pointing to		00049500	25
26		the FIB memory space.		00049900	26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1		17										1
2			Feature - Large Systems Code File Format For 3.5.0 Release							00010504		2
3												3
4			FPB EXAMPLE							00050000		4
5			-----							-----		5
6												6
7			SEGMENT NO. 002							00050050		7
8										00050055		8
9										00050065		9
10			LANGUAGE							00050070		10
11			(O=ALGOL)							00050075		11
12			FILELEVEL=1	INTNAME="CODE"		INTMODE=EBCDIC				00050080		12
13										00050085		13
14										00050090		14
15			010000000000	08010104C3D6	C4C500000000	031D04030801				00050100		15
16										00050175		16
17			TITLE="HEXER."							00050180		17
18										00050185		18
19										00050190		19
20			030F1E020006	C8C5E7C5D94B						00050200		20
21										00050300		21
22			000000000000	000000000000	000000000000	000000000000				00050400		22
23			000000000000	000000000000						00050500		23
24										00050600		24
25												25
26												26
27			COMPILER/INTRINSIC INTERFACE							00052100		27
28			-----							-----		28
29												29
30			The compilers interface code files with system intrinsics by							00052400		30
31			generating special descriptors (TAG 5) in the segment dictionary for							00052500		31
32			each intrinsic referenced. The descriptor format is:							00052700		32
33												33
34			TAG 5							00052800		34
35			[42:3] Invalid SZF=7							00053000		35
36			[23:11] Installation number							00053100		36
37			[12:13] Intrinsic number							00053200		37
38												38
39			The INTRINSICINFO table (0,17) contains all intrinsic/installation							00053500		39
40			numbers for the current intrinsic file. The compilers use this table							00053600		40
41			for intrinsic calls. For standard intrinsics (e.g., SIN), the							00053700		41
42			intrinsic numbers are predefined (see SYMBOL/BINDER at 10670000) and							00053800		42
43			the installation number is ZERO. For installation intrinsics, these							00053900		43
44			numbers originate from the BIND cards used to create the intrinsic							00054000		44
45			file.							00054100		45
46												46
47			Any call on an intrinsic is generated as a normal MKST, NAMC, ENTR							00054300		47
48			sequence, where the NAMC address couple is the D1 location of the							00054400		48
49			special TAG 5 descriptor. A special hardware interrupt will occur on							00054500		49
50			the first call on the intrinsic, due to the invalid SZF field. The							00054600		50
51			descriptor will be overwritten with an SIRW to the intrinsic's PCW in							00054700		51
52			the running intrinsic stack and subsequent calls will function as							00054800		52
53			procedure calls.							00054900		53
54												54
55												55
56												56
57												57
58												58
59												59
60												60





1				19		
2		Feature - Large Systems Code File Format For 3.5.0 Release			00010504	
3						
4		SUBSEQUENT CALLS ON INTRINSIC			00060401	
5					00060402	
6					00060403	
7					00060404	
8					00060405	
9					00060406	
10		IRW			00060407	
11					00060408	
12					00060410	
13		RCW			00060420	
14					00060430	
15				3	SEG DESCRIPTOR	
16		MSCW			00060450	
17					00060460	
18					00060470	
19					00060480	
20					00060490	
21					00060500	
22					00060510	
23					00060520	
24		USER STACK			00060530	
25					00060540	
26					00060550	
27		SEGMENT DICTIONARY (USER)			00060560	
28					00060570	
29					00060580	
30					00060590	
31					00060600	
32					00060610	
33					00060620	
34					00060630	
35					00060700	
36						
37						
38						
39						
40						
41						
42						
43						
44						
45						
46						
47						
48						
49						
50						
51						
52						
53						
54						
55						
56						
57						
58						
59						
60						

1	20				
2		Feature - Large Systems Code File Format For 3.5.0 Release		00010504	
3					
4					
5					
6		CODE FILE BINDER INFORMATION		00060900	
7		-----		-----	
8					
9		Extensive information is maintained in the code file which allows		00061200	
10		binding of program objects.		00061300	
11					
12		This information is comprised of:		00061500	
13					
14		PROGRAM DIRECTORY		00061600	
15		PROCEDURE DIRECTORY		00061700	
16		LOCAL DIRECTORIES		00061800	
17		REFERENCES		00061900	
18		SEGMENT DICTIONARY		00061940	
19					
20		Although a thorough understanding of these structures is not		00062200	
21		essential for analyzing a Large Systems Code file, the ability to		00062300	
22		identify the above components in the code file necessary.		00062320	
23					
24		For a description of the information kept for the Binder in the		00062380	
25		segment dictionary, see the section SEGMENT DICTIONARY (D1 STACK).		00062420	
26					
27					
28		BINDER CONVENTIONS		00062500	
29		-----		-----	
30					
31		To describe the content of the binder information, the following		00062800	
32		conventions will be used. The binder directories/descriptions are		00062900	
33		keyed by special words in the code file in the following form:		00063100	
34					
35		UC = [47:8] = Universal class		00063200	
36		SC = [39:20] = Subclass		00063300	
37		N = [19:20] = Length		00063400	
38					
39		Example:		00063700	
40					
41		02 0000 0004			
42				00063900	
43		Universal class = 2 = Single precision operand		00064000	
44		Sub class = 0 = Integer type		00064100	
45		Length = 4 Words		00064200	
46					
47		Keywords may be denoted as (UC, SC, N, ENTRY). Appendix A contains a		00064500	
48		complete list of universal/sub classes.		00064600	
49					
50		Identifiers are constructed as:		00064800	
51					
52					
53		4" TT 01 01 CC "8" NN . . . N"		00064900	
54				00065000	
55		TT = Total length in bytes of this construct		00065100	
56		CC = Number of characters in identifier		00065200	
57					
58					
59					
60					

1		21		1
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3	NN = The identifier		00065300	3
4	Individual stack items are described by ITEM DESCRIPTIONS. These		00065600	4
5	descriptions usually include an identifier entry (0,1,N) (IDENTIFIER)		00065700	5
6	and an address couple entry (0,2,N) (ADDRESS COUPLE).		00065800	6
7				7
8				8
9				9
10	PROGRAM DESCRIPTION		00065900	10
11	-----		-----	11
12				12
13				13
14	The program description (PD) is located by the code file segment		00066100	14
15	number at segment ZERO [16].[19:20]. It describes the outer		00066200	15
16	procedure (if this is a procedure program), global items (GLOBALS),		00066300	16
17	points at the procedure/block directory (called the PROCEDURE		00066450	17
18	DIRECTORY) and supplies miscellaneous code information (code address		00066500	18
19	of stack building code, etc.). The PD is also used for run-time		00066600	19
20	parameter checking of procedure programs (e.g., SYSTEM/DUMPALL). See		00066800	20
21	EXTERNALREFERENCE of DOCTOR for this usage.		00066900	21
22				22
23	Program descriptions are structured differently depending on whether		00067100	23
24	the program is a HOST/BLOCK program or a separate procedure program.		00067190	24
25				25
26	PROCEDURE PROGRAM	BLOCK PROGRAM	00067300	26
27	-----	-----	00067400	27
28				28
29	SEGO[16]-----	SEGO[16]-----	00067600	29
30			00067620	30
31			00067625	31
32	Item description of<-		00067700	32
33	outer procedure		00067800	33
34	including parameters		00067900	34
35			00068000	35
36	Procedure directory	Procedure directory<-	00068100	36
37	pointer	pointer	00068200	37
38			00068300	38
39	External directory	External directory	00068400	39
40			00068500	40
41	MISCELLANEOUS	MISCELLANEOUS	00068600	41
42				42
43	The procedure directory is a list of blocks/procedures in the program		00069200	43
44	and resides apart from the PD. The procedure directory pointer in		00069300	44
45	the PD is HEX encoded (0,22,L). The next word contains the disk		00069400	45
46	segment number of the procedure directory [19:20] and length [39:20].		00070000	46
47				47
48	Note: In the following discussion and examples, all program		00070015	48
49	description subclasses are in base 16.		00070020	49
50				50
51	After finding the PD from SEGO[16], a quick scan through the PD for		00070200	51
52	an entry like the following example will lead to the procedure		00070300	52
53	directory.		00070400	53
54				54
55	000002200005	000000500009	00070500	55
56			00070600	56
57				57
58				58
59				59
60				60

1	22									1
2		Feature - Large Systems Code File Format For 3.5.0 Release							00010504	2
3										3
4		UC=00, SC=22	PROCEDURE DIRECTORY AT						00070700	4
5			SEGMENT 9, LENGTH=5						00070800	5
6										6
7		Other elements of the program description are:							00071200	7
8										8
9		GLOBAL DIRECTORY (0,4,L)							00071300	9
10									00071400	10
11		Contains descriptions of all GLOBALS/EXTERNALS							00071500	11
12		for this program. Separate intrinsics are							00071600	12
13		entered here; e.g., (24,0,L).							00071700	13
14									00071800	14
15		MISCELLANEOUS INFORMATION							00071900	15
16									00072000	16
17		BLOCKEXIT POINTERS	(0,19,1) (PCW)						00072100	17
18		END D2 STACK BUILDING CODE	(0,12,1) (PCW)						00072200	18
19		D2 STACK SIZE	(0,15,1) (SIZE)						00072300	19
20									00072400	20
21		LT48 POINTERS FOR VALUE							00072500	21
22		ARRAYS (SUCH AS FPBS)	(0,E,1) (PCW)						00072600	22
23									00072603	23
24		LEVEL 2 SOFTWARE CONTROL							00072605	24
25		WORD IMAGE	(0,2A,1)						00072608	25
26									00072700	26
27		where PCW is a skeleton PCW (no tag) pointing							00072800	27
28		to the appropriate syllable of code.							00072900	28
29										29
30										30
31										31
32										32
33										33
34										34
35										35
36										36
37										37
38										38
39										39
40										40
41										41
42										42
43										43
44										44
45										45
46										46
47										47
48										48
49										49
50										50
51										51
52										52
53										53
54										54
55										55
56										56
57										57
58										58
59										59
60										60

1					23					
2	Feature - Large Systems Code File Format For 3.5.0 Release							00010504		
3										
4	PROGRAM DESCRIPTION EXAMPLE							00073100		
5	-----							-----		
6										
7	SEGMENT NO. 009							00073300		
8								00073400		
9								00073475		
10			EXTERNAL		DIRECTORY		00073477			
11	PROCEDURE DIRECTORY		(0,4,L)		INTRINSIC		00073479			
12	POINTER (0,22,L)		6 WORDS		(24,2,L)		00073481			
13	DIRECTORY AT RECORD #8				5 WORDS		00073482			
14							00073483			
15							00073485			
16	-----		-----		-----		00073500			
17	000002200003	000000300008	000000400006	240000200005			00073600			
18	000000100002	07010103F0F6					00073700			
19								00073780		
20								00073782		
21								00073784		
22								00073785		
23								00073790		
24	F135402EB60F	000000200001	000000100004	-----			00073800			
25								00073885		
26								00073890		
27	000A03600003	000803800008					00073900			
28								00073920		
29	BLOCKEXIT POINTER		END OF D2 STACK				00073940			
30	(0,19,L)		(0,12,L)				00073960			
31								00074000		
32	-----		-----		-----		00074050			
33	000001900001	000203508003	000001200001	000004000003			00074100			
34								00074110		
35	D2 STACK SIZE							00074120		
36	(0,15,L)							00074130		
37								00074140		
38								00074150		
39	-----						00074200			
40	000001500001	000000000009					00074300			
41	B280ABB7A060	32402EB6CF50	03AFABB51005	B180875005B9		00074400				
42	B327108BA180	00AE400AABA3					00074500			
43								00074600		
44	6000AF960DBE	270000840001	B20595B4BEEF	270000740002		00074700				
45	B20595B4B0BE	000001E00000					00074800			
46										
47										
48	PROCEDURE DIRECTORY							00075000		
49	-----							-----		
50										
51	The procedure directory contains information about all of the blocks							00075200		
52	and procedures in the program.							00075400		
53										
54										
55										
56										
57										
58										
59										
60										

1	24								1
2		Feature - Large Systems Code File Format For 3.5.0 Release						00010504	2
3									3
4		The procedure directory is located from the entry in the program						00075600	4
5		description of the form (0,22,L1) (L2,ADDRESS). L1 is the length of						00075600	5
6		the entire procedure directory.						00075705	6
7									7
8		If L1 = L2, then the entire procedure directory is in disk segment						00075715	8
9		ADDRESS, beginning in word 1. Word 0 of that segment is zero.						00075720	9
10									10
11		If L1 > L2, then the procedure directory is in several pieces. The						00075730	11
12		first piece is of length L2 and begins in word 1 of the disk segment						00075735	12
13		specified by ADDRESS. Word 0 of that segment is of the form						00075740	13
14		(L3,ADDRESS) and gives the length and disk segments number of the						00075740	14
15		next piece of procedure directory. This chain of disk segments						00075745	15
16		continue until a disk segment is reached that contains a zero in word						00075750	16
17		0; this disk segment contains the last piece of the procedure						00075755	17
18		directory. In each disk segment in the chain, the procedure						00075760	18
19		directory information begins in word 1.						00075800	19
20									20
21		The format of the procedure directory is as follows:						00076000	21
22									22
23		WORD 1						00076300	23
24									24
25		47:1 BIT47 1 if procedure is external.						00076600	25
26									26
27		46:1 BIT46 1 if local directory of procedure						00076700	27
28		contains owns or segment dictionary						00076800	28
29		items.						00076900	29
30									30
31		45:18 DISKSEGF disk segment number of local directory						00077050	31
32		for this procedure or block.						00077100	32
33									33
34		27:1 NEWPROCF 1 if this is a new procedure added to the						00077200	34
35		host because of an unresolved call on it						00077300	35
36		by a procedure bound in.						00077400	36
37									37
38		26:1 VALARYBIT 1 if this is not a procedure but a D1						00077500	38
39		value array.						00077700	39
40									40
41		25:5 LDWORDF Local directory of this procedure may not						00077850	41
42		start on segment boundary, but may start						00077900	42
43		at this word (0-29) of segment indicated						00078000	43
44		by DISKSEGF.						00078100	44
45									45
46		20:5 NESTLVLF Lexical level of this procedure.						00078400	46
47									47
48		15:3						00078600	48
49		12:13 SEGDIF Code 0 segment (see SEGSBIT) number which						00078750	49
50		is stack building code for this						00078800	50
51		procedure. All non-own and non-segment						00078850	51
52		directory items in the local directory						00078900	52
53		are built in this segment (code also						00078900	53
54		exists there).						00079050	54
55									55
56									56
57									57
58									58
59									59
60									60

1	Feature - Large Systems Code File Format For 3.5.0 Release			00010504	1
2					2
3					3
4	WORD 2			00079200	4
5					5
6					6
7	47:8	Number of characters in identifier.		00079450	7
8		blocks; e.g. BLOCK#6.		00079500	8
9	XXXXXXX	identifier (in EBCDIC). Compilers assign		00079600	9
10		names to blocks; e.g. BLOCK#6. As many		00079700	10
11		words as necessary.		00079800	11
12					12
13	WORDS N-M			00079900	13
14					14
15	15:1	SEGSBIT	Implies that this word contains the	00080100	15
16			number of all the code segments that are	00080200	16
17			unique to this procedure.	00080300	17
18					18
19	12:13, 28:13, 44:13	Up to 3 code segment numbers per word are		00080450	19
20		placed in these fields in the order list		00080500	20
21		at only if SEGSBIT is on.		00080600	21
22					22
23	TYPICAL ENTRY			00080700	23
24	-----			-----	24
25					25
26					26
27					27
28	SEGSBIT			00080900	28
29	ON			00081000	29
30				00081100	30
31				00081190	31
32	ID,			00081193	32
33	LENGTH			00081200	33
34	-----	-----	-----	00081300	34
35	WORD 1	4 PROC	1 1 WORD 1 of next entry	00081400	35
36	-----	-----	-----	00081500	36
37	EXTRA SEGMENTS			00081600	37
38	other than stack building segment			00081650	38
39	indicated in WORD 1				39
40	Note: If no extra segments are contained in a procedure, the word			00081900	40
41	following the IDENTIFIER will have 15:1 = 0 since it will be			00081900	41
42	WORD 1 of the next entry.			00082100	42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60





1	Feature - Large Systems Code File Format For 3.5.0 Release			00010504	1
2					2
3	LOCAL DIRECTORY EXAMPLE			00085500	3
4	-----			-----	4
5					5
6					6
7	LOCAL DIRECTORY	FILE	IDENTIFIER	00085800	7
8	(0,21,18)	(16,1,5)	(0,1,L)	00085875	8
9				00085880	9
10				00085885	10
11				00085890	11
12	000002100018	160000100005	000000100002	00085900	12
13			ADDRESS COUPLE	00085975	13
14	"LINE"		(0,2,L)	00085980	14
15				00085985	15
16				00085990	16
17	08010104D3C9	D5C500000000	000000200001	00086000	17
18				00086100	18
19	(2,3)			00086120	19
20				00086140	20
21				00086160	21
22	000000200003	160000100005	000000100002	00086200	22
23	08010104C3D6	C4C500000000	000000200001	00086300	23
24				00086400	24
25	000000200004	020000000004	000000100001	00086500	25
26	05010101C900	000000200001	000000200005	00086600	26
27				00086700	27
28	060000100006	000000100001	05010101C100	00086800	28
29	000000200001	000000200006	000000F00001	00086900	29
30				00087000	30
31	000000000001	000000000000	000000000000	00087100	31
32	000000000000	000000000000	000000000000	00087200	32
33				00087300	33
34					34
35					35
36					36
37					37
38					38
39	REFERENCES			00087500	39
40	-----			-----	40
41					41
42	References are pointers into the code file that mark code to be fixed			00087800	42
43	up should binding occur.			00087900	43
44					44
45	In the segment dictionary, if the TAG field of a segment descriptor			00088100	45
46	is 4"8300", there are references to code in that segment. The word			00088200	46
47	immediately following the object code in the disk segment (the length			00088300	47
48	of the code is given in the segment discriptor) has the following			00088400	48
49	format:			00088500	49
50					50
51	[47:12] REFNUMF	Number of references following this word (the		00088600	51
52		references may cross the segment boundary		00088620	52
53		into the following disk segment).		00088650	53
54					54
55	[35:20] REFDISKADF	If zero, all of the references follow		00088700	55
56		this word.If nonzero, it is the disk		00088720	56
57					57
58					58
59					59
60					60

1	28			1
2		Feature - Large Systems Code File Format For 3.5.0 Release	00010504	2
3				3
4		segment number of the next group	00088740	4
5		of references.	00088760	5
6				6
7		If REFDISKADF is nonzero, then the references appear in more than	00088900	7
8		group, and the next group is in the disk segment addressed by	00089000	8
9		REFDISKADF. In that segment, word 0 has the following format:	00089020	9
10				10
11		[47:12] REFNUMF The number of references following this word.	00089030	11
12				12
13		[35:20] REFDISKADF If zero, this is the last group of	00089050	13
14		references. If nonzero, it is the disk	00089055	14
15		segment number of the next group of	00089060	15
16		references.	00089065	16
17			00089080	17
18				18
19		The references following this word are in the form of a PCW with SDI,	00089300	19
20		word index, and syllable index fields.	00089400	20
21				21
22		References refer to operators that contain address couples, such as	00089520	22
23		NAMC and VALC, where the address couple addresses D1 or D2 stack	00089540	23
24		locations (since these addresses may change upon binding), or to	00089560	24
25		LT48/MPCW operators in D2 stack building code (the literal or the PCW	00089580	25
26		may require modification upon binding).	00089600	26
27				27
28				28
29				29
30				30
31				31
32		FPB/PPB RUN-TIME INFORMATION	00089700	32
33		-----	-----	33
34				34
35		SHEET[0] indicates to the compiler whether run-time file-equation	00090000	35
36		information (FPB) or program parameter information (PPB) is to be	00090100	36
37		written to the code file.	00090300	37
38				38
39		This information is usually the last segment of the code file. The	00090500	39
40		format of the FPB was discussed in the FPB section. The PPB is a	00090600	40
41		list of task attributes to be applied at task initiation.	00090800	41
42				42
43		The PPB (program parameter block) is formatted as follows:	00091000	43
44				44
45		each task	00091100	45
46		PTASKLINKF = [47:28]	00091200	46
47		PATLINKF = [19:20]	00091300	47
48			00091400	48
49		for each attribute	00091500	49
50		PATNOF = [47:08] attribute number	00091600	50
51		PATTYPEF = [39:08] attribute type (i.e., STRING ...)	00091700	51
52		PATMISCF = [31:08]	00091800	52
53		PATLENF = [23:24] length of entry	00091900	53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1		29		
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	
3				
4	Word 0 of the segment for an FPB or PPB contains the following:		00092400	
5				
6	[47:1] = 1 for a PPB, 0 for an FPB		00092500	
7	[19:20] = length of PPB/FPB		00092600	
8				
9				
10	PPB EXAMPLE		00092800	
11	-----		-----	
12				
13				
14			00093100	
15		ATTRIBUTE NUMBER=0	00093175	
16	[47:1]=1	=NYNAME	00093180	
17			00093184	
18			00093190	
19	800000000006	000000400000	000200000003	
20	COMPILED PROGRAM NAME		00093200	
21			00093280	
22			00093285	
23			00093290	
24	09010105C8C5	E7C5D9000000	000000000000	
25			00093300	
26	0000003D0500	080000840007	000000100004	
27	000000E00002	000A03600003	000803800003	
28			00093700	
29	000001900001	000203508003	000001200001	
30	000004000003	000001500001	000000000009	
31			00093800	
32	B280ABB7A060	32402EB60F50	03AFABB51005	
33	B18087500589	B327108BA180	00AE400AABA3	
34			00094100	
35	6000AF960DBE	270000840001	B20595B4BEFF	
36	270000740002	B20595B4B0BE	000001E00000	
37			00094200	
38			00094300	
39			00094400	
40			00094500	
41				
42				
43				
44				
45				
46				
47				
48				
49				
50				
51				
52				
53				
54				
55				
56				
57				
58				
59				
60				

1	30								1
2		Feature - Large Systems Code File Format For 3.5.0 Release						00010504	2
3									3
4									4
5									5
6									6
7									7
8									8
9		APPENDIX A						00094900	9
10		-----						-----	10
11									11
12									12
13		Supplementary Binder Information						00095100	13
14		-----						-----	14
15									15
16		This appendix outlines supplementary information needed for the						00095400	16
17		section on code file binder data.						00095500	17
18									18
19									19
20									20
21									21
22									22
23		UNIVERSAL INFORMATION						00095600	23
24		-----						-----	24
25									25
26		A typical entry has the following form:						00096696	26
27									27
28								00096701	28
29								00096702	29
30		: UC : SC : N :						00096703	30
31		: : : :						00096704	31
32		-----						00096705	32
33		: : :						00096706	33
34		: STUFF : :						00096707	34
35		: : : > N WORDS						00096708	35
36		-----						00096709	36
37								00096710	37
38								00096711	38
39								00097000	39
40									40
41		where UC FIELD = [47:8]						00097200	41
42		SC FIELD = [39:20]						00097400	42
43		N FIELD = [19:20]						00097500	43
44									44
45		Such an entry may be denoted by (UC,SC,N,STUFF).						00097800	45
46									46
47									47
48									48
49									49
50									50
51									51
52									52
53									53
54									54
55									55
56									56
57									57
58									58
59									59
60									60





1	Feature - Large Systems Code File Format For 3.5.0 Release			00010504	1
2					2
3					3
4	UC	SC	DESCRIPTION	00101600	4
5	--	--	-----	00101700	5
6					6
7		46	LIST OF FORMATS AND SUBFORMATS INVOKED FOR TRANSACTION BASE	00106160	7
8				00106165	8
9		47	TRANSACTION BASE FORMAT	00106170	9
10		48	TRANSACTION BASE SUBFORMATILOG PCW BELOW SCW	00106180	10
11		49	TRANSACTION BASE ASSOCIATED WITH TR RECORD OR RECORD ARRAY	00106190	11
12		50	PROCEDURE/SUBROUTINE COMPILERINFO WORD	00106200	12
13		51	LIBRARY TEMPLATE ATTRIBUTES	00106210	13
14		52	DATA DICTIONARY INVOCATION DIRECTORY	00106220	14
15		53	DATA DICTIONARY USAGE DIRECTORY	00106230	15
16		54	ENTRY POINT DESCRIPTION OF A LIBRARY ENTRY POINT	00106240	16
17		52	LIBRARY TO WHICH AN ENTRY POINT BELONGS	00106250	17
18					18
19	2	0	SP OPERAND (INTEGER)	00106400	19
20		1	SP OPERAND (REAL)	00106500	20
21		2	SP OPERAND (BOOLEAN)	00106600	21
22		3	WORD VARIABLE	00106700	22
23		4	REFERENCE VARIABLE	00106800	23
24	4	1	EVENT	00106900	24
25		3	DOUBLE PRECISION OPERAND	00107000	25
26		4	COMPLEX OPERAND	00107100	26
27	6	0	SP ARRAY (INTEGER)	00107200	27
28		1	SP ARRAY (REAL)	00107300	28
29		2	SP ARRAY (BOOLEAN)	00107400	29
30		3	WORD ARRAY	00107500	30
31		4	REFERENCE ARRAY	00107600	31
32		5	DIRECT INTEGER ARRAY	00107700	32
33		6	DIRECT REAL ARRAY	00107800	33
34		7	DIRECT BOOLEAN ARRAY	00107900	34
35		8	COBOL SEGMENTED ARRAY	00108000	35
36		9	FORTRAN VARFILE ARRAY	00108100	36
37		10	NEW FORMAT ARRAY	00108300	37
38		11	TRANSACTION RECORD OR RECORD ARRAY	00108350	38
39	8	0	SP SUBSCRIPTED VARIABLE (INTEGER)	00108500	39
40		1	SP SUBSCRIPTED VARIABLE (REAL)	00108600	40
41		2	SP SUBSCRIPTED VARIABLE (BOOLEAN)	00108700	41
42		3	FORMAT (ALGOL)	00108800	42
43		4	PICTURE (ALGOL)	00108900	43
44	10	2	EVENT ARRAY	00109100	44
45		3	DOUBLE PRECISION ARRAY	00109200	45
46		4	COMPLEX ARRAY	00109300	46
47		5	DIRECT DOUBLE ARRAY	00109400	47
48	12	3	DOUBLE PRECISION SUBSCRIPTED VARIABLE	00109500	48
49		4	COMPLEX SUBSCRIPTED VARIABLE	00109600	49
50	14	1	CHARACTER ARRAY:CHARACTER SIZE=4	00109700	50
51		2	CHARACTER ARRAY:CHARACTER SIZE=6	00109800	51
52		3	CHARACTER ARRAY:CHARACTER SIZE=8	00109900	52
53		4	NON REENTRANT FORMAT ARRAY (FORTRAN)	00110000	53
54		5	DIRECT CHARACTER ARRAY:CHARACTER SIZE=4	00110100	54
55		6	DIRECT CHARACTER ARRAY:CHARACTER SIZE=6	00110200	55
56		7	DIRECT CHARACTER ARRAY:CHARACTER SIZE=8	00110300	56
57					57
58					58
59					59
60					60



1	34		Feature - Large Systems Code File Format For 3.5.0 Release	00010504	1
2					2
3					3
4	UC	SC	DESCRIPTION	00101600	4
5	--	--	-----	00101700	5
6					6
7	16	1	POINTER	00110400	7
8	18	0	SUBROUTINE	00110500	8
9		1	INITIALIZATION SUBROUTINE (FORTRAN)	00110600	9
10		2	LABEL	00110700	10
11		3	SWITCH (LABEL)	00110800	11
12		4	LABEL FOR STOP IN SUBROUTINE (FORTRAN)	00110900	12
13		5	SWITCH FORMAT	00111000	13
14		6	SWITCH LIST	00111100	14
15		7	SWITCH FILE	00111200	15
16		8	DIRECT SWITCH FILE	00111300	16
17		9	MCP SAVE CODE (SEGMENT 1 AND/OR 5)	00111400	17
18		10	FORMAL LABEL	00111600	18
19		11	EPILOG PROCEDURE (DCALGOL,ESPOL)	00111615	19
20	20	0	FUNCTION (INTEGER)	00111800	20
21		1	FUNCTION (REAL)	00111900	21
22		2	FUNCTION (BOOLEAN)	00112000	22
23		3	FUNCTION (DOUBLE PRECISION)	00112100	23
24		4	FUNCTION (COMPLEX)	00112200	24
25		5	FORTRAN EXTERNAL SUBPROGRAM EXTERNAL INTRINSIC	00112300	25
26			OR (NOTHING UNKNOWN)	00112400	26
27		6	I/O LIST (ALGOL)	00112500	27
28		7	POINTER PROCEDURE	00112600	28
29		8	WORD PROCEDURE	00112700	29
30		9	REFERENCE PROCEDURE	00112800	30
31		10	& UP - PL/I FUNCTIONS	00113000	31
32	22	1	FILE	00113200	32
33		2	DIRECT FILE	00113300	33
34		3	COBOL FILE	00113400	34
35		4	DM RESTART FILE (COBOL)	00113500	35
36	24	1	TASK	00113600	36
37		2	TASK ARRAY	00113700	37
38	26	1	FORTRAN COMMON BLOCK	00113800	38
39		2	FORTRAN EQUIVALENCE ARRAY	00113900	39
40	28	1	INTERRUPT	00114000	40
41	30	1	MESSAGE	00114100	41
42		2	MESSAGE ARRAY	00114200	42
43	32	1	QUEUE	00114300	43
44		2	QUEUE ARRAY	00114400	44
45	34	0	INTRINSIC (SUBROUTINE)	00114500	45
46	36	0	INTRINSIC (INTEGER)	00114600	46
47		1	INTRINSIC (REAL)	00114700	47
48		2	INTRINSIC (BOOLEAN)	00114800	48
49		3	INTRINSIC (DOUBLE PRECISION)	00114900	49
50		4	INTRINSIC (COMPLEX)	00115000	50
51	38		PL/I VARIABLES	00115100	51
52	40	0	DATA BASE	00115115	52
53	42	0	EBCDIC STRING VARIABLE (ALGOL)	00115120	53
54		1	ASCII STRING VARIABLE (ALGOL)	00115125	54
55		2	HEX STRING VARIABLE (ALGOL)	00115130	55
56	44	0	EBCDIC STRING ARRAY (ALGOL)	00115135	56
57					57
58					58
59					59
60					60

1					35			
2			Feature - Large Systems Code File Format For 3.5.0 Release				00010504	
3								
4	UC	SC	DESCRIPTION				00101600	
5	--	--	-----				00101700	
6								
7		1	ASCII STRING ARRAY (ALGOL)				00115140	
8		2	HEX STRING ARRAY (ALGOL)				00115145	
9	46	0	EBCDIC STRING PROCEDURE (ALGOL)				00115150	
10		1	ASCII STRING PROCEDURE (ALGOL)				00115155	
11		2	HEX STRING PROCEDURE (ALGOL)				00115160	
12	48	0	ALTERNATIVE ITEM GROUP (NEWP)				00115165	
13	50	1	DATA DICTIONARY DESCRIPTION OF DATABASE				00115170	
14		2	DATA DICTIONARY DESCRIPTION OF TRANSACTION BASE				00115175	
15		3	DATA DICTIONARY DESCRIPTION OF DATA SET				00115180	
16		4	DATA DICTIONARY DESCRIPTION OF TRANSACTION FORMAT				00115185	
17		5	DATA DICTIONARY DESCRIPTION OF EXPLICITLY-INVOKED DISJOINT SET				00115190	
18							00115192	
19		6	DATA DICTIONARY DESCRIPTION OF GLOBAL DATA SET				00115195	
20	52	0	TRANSACTION BASE				00115200	
21	54	0	LIBRARY				00115205	
22	56	0	LIBRARY ENTRY POINT				00115210	
23	58	0	EXPORTED SUBROUTINE				00115215	
24		1	EXPORTED INTEGER FUNCTION				00115220	
25		2	EXPORTED REAL FUNCTION				00115225	
26		3	EXPORTED BOOLEAN FUNCTION				00115230	
27		4	EXPORTED DOUBLE FUNCTION				00115235	
28		5	EXPORTED COMPLEX FUNCTION				00115240	
29		6	EXPORTED EBCDIC STRING PROCEDURE				00115245	
30		7	EXPORTED ASCII STRING PROCEDURE				00115250	
31		8	EXPORTED HEX STRING PROCEDURE				00115255	
32	60	0	BY-CALLING SUBROUTINE OF FUNCTION				00115260	
33							00115270	
34			NOTES:				00115300	
35								
36			(1) AN ODD UC MEANS THE ITEM IS A CALL-BY-NAME OR				00115340	
37			CALL-BY-REFERENCE				00115360	
38			PARAMETER.				00115400	
39								
40			(2) SUBROUTINE=UNTYPED PROCEDURE,				00115600	
41			FUNCTION=TYPED PROCEDURE				00115990	
42								
43								
44								
45								
46								
47								
48								
49								
50								
51								
52								
53								
54								
55								
56								
57								
58								
59								
60								

1	36			1
2		Feature - Large Systems Code File Format For 3.5.0 Release	00010504	2
3				3
4				4
5				5
6				6
7				7
8				8
9		KEYWORD DESCRIPTIONS	00116100	9
10		-----		10
11		(0,1,N,IDENTIFIER)	00116300	11
12				12
13		IDENTIFIER has the form:	00116500	13
14				14
15		4"TT0101CC"8"XX...X"	00116700	15
16				16
17		where:	00116900	17
18				18
19		CC is the number of characters in the IDENTIFIER string	00117100	19
20		TT=CC+4 (total number of bytes in string)	00117200	20
21		N=(TT+5) DIV 6	00117300	21
22				22
23		The string is left justified and the last word contains undefined	00117600	23
24		filler.	00117700	24
25				25
26		(0,2,1,ADDRESSCOUPLE)	00117800	26
27				27
28		ADDRESSCOUPLE has the form:	00118000	28
29				29
30		:////: LEVEL : DISP :	00118200	30
31				31
32		where:	00118400	32
33				33
34		LEVEL=[39:20]	00118600	34
35		DISP =[19:20]	00118700	35
36				36
37		(0,3,N,PARAMETERS)	00118900	37
38				38
39		PARAMETERS has the form:	00119100	39
40				40
41		(0,9,1,P) :	00119300	41
42				42
43		(UC1,SC1,M1,ITEM1) :	00119400	43
44				44
45		(UC2,SC2,M2,ITEM2) > N WORDS	00119600	45
46				46
47		:	00119700	47
48		:	00119800	48
49		(UCP,SCP,MP,ITEMP) :	00119900	49
50				50
51		where:	00120000	51
52				52
53		P is the number of parameters (P GEQ 0)	00120400	53
54		N is the total number of words describing parameters (N GEQ 0)	00120500	54
55				55
56				56
57				57
58				58
59				59
60				60

1		37		1
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3				3
4	Each item is a description of a parameter. The items must appear		00120800	4
5	in the order of the required parameters.		00120900	5
6				6
7	(0,4,N,GLOBALDIRECTORY)		00121000	7
8				8
9	GLOBALDIRECTORY has the form:		00121200	9
10				10
11	(UC1,SC1,M1,ITEM1) :		00121400	11
12	:		00121500	12
13	:	> N WORDS	00121600	13
14	:		00121700	14
15	(UCP,SCP,MP,ITEMP) :		00121800	15
16				16
17	where:		00122000	17
18				18
19	N is the total number of words in the global directory		00122200	19
20				20
21	Each item is a description of something global to an external		00122400	21
22	procedure.		00122500	22
23				23
24	For a block program, each item must contain:		00122700	24
25				25
26	(0,1,N1,IDENTIFIER)		00122800	26
27	(0,2,N2,ADDRESSCOUPLE)		00122900	27
28	(0,15,1,NUMBERDIMENSIONS) IF ARRAY		00123000	28
29	(0,7,N7,MPCWPOINTER) IF EXTERNAL PROCEDURE		00123100	29
30	(0,14,1,LENGTH) IF FORTRAN COMMON BLOCK		00123200	30
31	(0,3,N3,PARAMETERS) IF PROCEDURE WITH PARAMETERS.		00123300	31
32				32
33	For a procedure program, each item description must contain the		00123600	33
34	above plus:		00123700	34
35				35
36	(0,6,N6,GLOBALREFERENCES)		00123800	36
37				37
38	The description for an external procedure is contained in a value		00124100	38
39	array whose descriptor is in the segment dictionary. In this		00124100	39
40	case, the global directory entry may contain:		00124300	40
41				41
42	(UC,SC,2)		00124400	42
43	(0,17,1,SDI)		00124500	43
44				44
45	where SDI is the segment dictionary index for the descriptor, or		00124800	45
46	it may be a normal entry.		00124900	46
47				47
48	(0,5,N,D1REFERENCES)		00125000	48
49				49
50	D1REFERENCES has the form:		00125200	50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	38			1
2		Feature - Large Systems Code File Format For 3.5.0 Release	00010504	2
3				3
4		SP1 :	00125400	4
5		:	00125500	5
6		SP2 > N WORDS	00125600	6
7		:	00125700	7
8		SPN :	00125800	8
9				9
10		where each SP is a "SYLLABLE POINTER" to an operator referring to	00126100	10
11		a level 1 item. The syllable pointer has the form of a PCW with a	00126200	11
12		TAG of 0 as follows:	00126300	12
13				13
14		://///:PSR: PIR : :LL: SDI :	00126400	14
15				15
16		where:	00126600	16
17				17
18		PSR=[35:3] = syllable index for start of the operator	00126800	18
19		PIR=[32:13] = word index for start of the operator	00126900	19
20		LL =[18:5] = lex level at which the operator is executed	00127000	20
21		SDI=[13:14] = segment dictionary index for code segment	00127100	21
22		containing the operator	00127200	22
23				23
24		(0,6,N,GLOBALREFERENCES)	00127400	24
25				25
26		GLOBALREFERENCES has the same form as DIREFERENCEs. The operator	00127700	26
27		VALC may reference any level global to the procedure in which it	00127800	27
28		occurs.	00127900	28
29				29
30		(0,7,N,MPCWPOINTERS)	00128000	30
31				31
32		MPCWPOINTER is a syllable pointer to a LT48 operator which is used	00128300	32
33		to make a literal containing the SDI of the value array descriptor	00128400	33
34		for the external procedure.	00128500	34
35				35
36		(0,9,1,NUMBERPARAMETERS)	00128600	36
37				37
38		NUMBERPARAMETERS is the number of parameters required by this	00128800	38
39		program, procedure or entry.	00129000	39
40				40
41		(0,10,N,EXTERNALID)	00129100	41
42				42
43		EXTERNALID is a explicit file identifier in KUNKERFORM.	00129400	43
44				44
45		(0,11,N,OWNDIRECTORY)	00129500	45
46				46
47		OWNDIRECTORY is a list of own variables which were declared in a	00129800	47
48		procedure program. These are items which must be added to the	00129900	48
49		host D2 stack. The format is the same as that in the global	00130000	49
50		directory.	00130100	50
51				51
52		(0,12,N,OWNREFERENCES)	00130200	52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1		39		1
2	Feature - Large Systems Code File Format For 3.5.0 Release		00010504	2
3				3
4	OWNREFERENCES is a list of syllable pointers to all code		00130450	4
5	references to an own variable.		00130600	5
6				6
7	(0,13,N,MAKEFIBPOINTERS)		00130700	7
8				8
9	MAKEFIBPOINTERS is a list of syllable pointers to LT48 operators		00131000	9
10	for making FIB descriptors. There must be one of these for each		00131100	10
11	file. It may be in the global directory or directly in the		00131200	11
12	program description.		00131300	12
13				13
14	(0,14,N,MAKEARRAYDESCPNTRS)		00131400	14
15				15
16	MAKEARRAYDESCPNTRS is a list of syllable pointers to LT48		00131700	16
17	operators for making value array descriptors. One of these is		00131800	17
18	required for each value array (such as format strings) that is not		00131900	18
19	a file and does not have a D1 descriptor.		00132000	19
20				20
21	(0,15,1,NUMBERDIMENSIONS)		00132100	21
22				22
23	NUMBERDIMENSIONS is the number of dimensions (in the ALGOL or		00132400	23
24	hardware sense) required by an array.		00132500	24
25				25
26	(0,16,1,LENGTH)		00132600	26
27				27
28	LENGTH is the number length in elements of an array.		00132900	28
29				29
30	(0,17,1,SDI)		00133000	30
31				31
32	SDI is the segment dictionary index of the value array descriptor		00133300	32
33	for an external procedure.		00133400	33
34				34
35	(0,18,1,ENDOFD2CODE)		00133500	35
36				36
37	ENDOFD2CODE is a syllable pointer to the syllable immediately		00133800	37
38	after the D2 stack building code.		00133900	38
39				39
40	(0,19,1,FIRSTEX)		00134000	40
41				41
42	FIRSTEX is a syllable pointer to the first executable code		00134300	42
43	syllable of a separately-compiled procedure. (This refers to the		00134400	43
44	procedure itself, not the pseudo outer block.)		00134500	44
45				45
46	(18,SC,N,PROCEDUREITEM)		00134600	46
47	or		00134700	47
48	(20,SC,N,PROCEDUREITEM)		00134800	48
49				49
50	PROCEDUREITEM is a description of a procedure. This may be found:		00135100	50
51				51
52	(1) As the first entry in the program description for a procedure		00135300	52
53	program.		00135400	53
54	(2) As an external procedure description.		00135600	54
55	(3) As an entry in the global directory in either the program		00135800	55
56	description or an external procedure description.		00135900	56
57				57
58				58
59				59
60				60

1	40			1
2		Feature - Large Systems Code File Format For 3.5.0 Release	00010504	2
3				3
4		A PROCEDUREITEM must include:	00136100	4
5				5
6		(0,1,N,IDENTIFIER)	00136200	6
7		(0,2,1,ADDRESSCOUPLE)	00136300	7
8		(0,3,N,PARAMETERS) IF IT HAS PARAMETERS	00136400	8
9		(0,7,1,MPCWPOINTER) IF EXTERNAL PROCEDURE DESCRIPTION	00136500	9
10			.FINISH	10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60

1	.FLAG 52	00017810	.FLAG 52	1
2				2
3				3
4				4
5				5
6				6
7				7
8				8
9				9
10				10
11				11
12				12
13				13
14				14
15				15
16				16
17				17
18				18
19				19
20				20
21				21
22				22
23				23
24				24
25				25
26				26
27				27
28				28
29				29
30				30
31				31
32				32
33				33
34				34
35				35
36				36
37				37
38				38
39				39
40				40
41				41
42				42
43				43
44				44
45				45
46				46
47				47
48				48
49				49
50				50
51				51
52				52
53				53
54				54
55				55
56				56
57				57
58				58
59				59
60				60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	E V E N T S	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	A P P E N D I X C	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1	WORKFILE [NOT NAMED]		5:03 PM TUESDAY, APRIL 9, 1985		1	
2					2	
3					3	
4	12260000	DEFINE	% ESPOL FIELDS	12260000	31.061.240	4
5	12260200	% FIELDS IN FIRST WORD OF EVENT		12260200	31.055.217	5
6	12260400	EVENTUSAGEF=[47:2]#	% USAGE OF SECOND WORD OF EVENT	12260400	32.044.309	6
7	12260800	, PROCUREDATF=[36:35]#	% TRANSFERRED FROM PROCURER LINKWORD	12260800	351022452	7
8	12261000	, PROCURERF=[36:SNSIZE]#	% STACK WHICH FIXED OR PROCURED	12261000	351022452	8
9	12261400	, EVENTQHEADF=[24:21]#	% LINKAGE TO WAITING STACK:	12261400	351022452	9
10	12261600	, EVENTNUMF=[24:9]#	% RELATIVE INDEX (0,2,4...)	12261600	351022452	10
11	12261800	, EVENTSNRF=[15:SNSIZE]#	% STACK NUMBER	12261800	35.026.736	11
12	12262000	, UNAVAILABLEF=[2:1]#	% NOT AVAILABLE(EVENT)	12262000	31.061.240	12
13	12262200	, HAPPENEDBITF=[0:1]#	% HAPPENED(EVENT)	12262200	31.061.240	13
14	12262240	% VALUES OF THE EVENTUSAGE FIELD		12262240	32.044.309	14
15	12262260	, EVUNUSED = 0 #	% SECOND WORD NOT IN USE	12262260	32.044.309	15
16	12262280	, EVRWTLOCK = 1 #	% EVENT USED AS READ/WRITER/THINKER	12262280	32.044.309	16
17	12262300	, EVINTATTACHED = 2 #	% SOFTWAREINTERRUPT ATTACHED	12262300	32.044.309	17
18	12262320	, EVINSCRIBED = 3 #	% EVENT INSCRIBED WITH SIGNATURE	12262320	32.044.309	18
19	12262340	% FIELDS IN 2ND WORD OF EVENT WHEN INSCRIBED WITH A SIGNATURE		12262340	32.044.309	19
20	12262360	% THE WHOLE WORD IS THE INDEX INTO THE GETAREA POOL OF THE MEMO		12262360	32.044.309	20
21	12262400	% FIELDS IN 2ND WORD OF EVENT WHEN ATTACHED TO SOFTWARE INTERRUPT		12262400	31.055.217	21
22	12262600	% THE WHOLE WORD IS AN ABSOLUTE ADDRESS TO A LINKAGE AREA.		12262600	31.082.217	22
23	12262800	% FIELDS IN 2ND WORD OF EVENT WHEN USED AS READER/WRITER/THINKER LOCK		12262800	31.055.217	23
24	12263000	, READERCOUNTF=[23:12]#	% NUMBER OF READERS	12263000	31.061.240	24
25	12263200	, THINKERSNRF=[35:SNSIZE]#	% STACK NUMBER OF THE THINKER, OR 0	12263200	35.026.736	25
26	12263400	, WRITINGF=[0:1]#	% THE THINKER STACK IS WRITER	12263400	31.061.240	26
27	12263410	% FIELDS IN A LINKWORD (FOR AN EVENT) IN THE SIDECAR		12263410	351022452	27
28	12263420	, EVLINK STATEF=[45:02]#	% STATE OF THE LINKWORD	12263420	351022452	28
29	12263430	, EVLINK_NORMALV = 1'00' #	% LINKED TO AN EVENT	12263430	351022452	29
30	12263440	, EVLINK_CAUSEDV = 1'01' #	% MARKS EVENT THAT WOKE UP STACK	12263440	351022452	30
31	12263450	, EVLINK_DISCARDV = 1'11' #	% NOT LINKED TO AN EVENT	12263450	351022452	31
32	12263460	, EVLINK_TIMEV = 1'10' #	% LINKWORD IS LINKED IN CALENDAR	12263460	351022452	32
33	12263470	% PROCUREDATF=[36:35]#	% USED FOR PROCURING EVENT	12263470	351022452	33
34	12263480	% PROCURERF =	% USED FOR PROCURING EVENTS	12263480	351022452	34
35	12263490	% EVENTQHEADF =	% USED FOR PROCURING AND WAITING	12263490	351022452	35
36	12263500	% UNAVAILABLEF=[ 2: 1]#	% USED FOR PROCURING EVENTS	12263500	351022452	36
37	12263600	% FIELDS IN CALENDAR LINKWORD (1ST OF PAIR)		12263600	31.055.217	37
38	12263800	, CALTYPEF=[47:2]#	% WHAT TO DO AT THE SPECIFIED TIME	12263800	35.008.265	38
39	12263850	% EVLINK STATEF=[45:02]#	% SET TO EVLINK TIMEV WHEN LINKED IN	12263850	351022452	39
40	12263900		% CALENDAR, SET TO EVLINK_CAUSEDV	12263900	351022452	40
41	12263950		% ONCE ITS DELINKED.	12263950	351022452	41
42	12264000	, QBACKF=[39:20]#	% BACKWARD LINK (SEGMENT NUMBER)	12264000	351022452	42
43	12264200	, QNEXTF=[19:20]	% FORWARD LINK "	12264200	31.061.240	43
44	12264400	% FIELDS IN CALENDAR INFO WORD (2ND OF PAIR; SEE ALSO WAIT LINK WORD)		12264400	31.055.217	44
45	12264600	, CALSNRF=[47:SNSIZE]#	% STACK NUMBER TO BE AWAKENED	12264600	35.026.736	45
46	12264800	, CALTIMEF=[35:32]#	% WAKE-UP VALUE FOR CLOCK/255	12264800	31.061.240	46
47	12265000	% FIELDS IN 2ND PARAMETER TO WAITP,TIMETUNNEL (THIS INFORMATION IS		12265000	351022452	47
48	12265100	%	COPIED TO THE BEDWORD)	12265100	351022452	48
49	12265200	, NOTDSABLEF=[47:1]#	% DSABLE OPTION (FROM COMPILER)	12265200	31.153.521	49
50	12265400	, DSABLEF=[46:1]#	% NOTDSABLE OPTION (FROM COMPILER)	12265400	31.153.521	50
51	12266200	, SWAPF=[3:2] #	% SWAP OPTIONS (TEMP) (FROM COMPILER)	12266200	31.153.521	51
52	12266400	, SWAPNOWV = 1'11' #	% SWAP IMMEDIATELY	12266400	32.048.390	52
53	12266600	, DELAYSWAPV = 1'01' #	% SWAP IF STILL ASLEEP AFTER WHILE	12266600	32.048.390	53
54	12266800	, NOSWAPV = 1'10' #	% DO NOT SWAP	12266800	32.048.390	54
55	12267000	%	NO RECOMMENDATION	12267000	32.048.390	55
56	12267200	, RESETF=[0:1]#	% RESET EVENT (FROM COMPILER)	12267200	31.153.521	56
57	12267400	% FIELDS IN BEDWORD OF PIB		12267400	31.170.674	57
58	12267500	% NOTDSABLEF=[47:01]#		12267500	351022452	58
59	12267600	, DSABLEF=[46:01]#		12267600	351022452	59
60	12267800	, EVENTCOUNTF=[44:09]#	% NUMBER OF EVENTS AWAITED	12267800	351022452	60

1	12268000	, EVENTOFFSETF = [35:16]# %	OFFSET TO EVENTS IN STACK	12268000	351022452	1
2	12268005	% SWAPF = [ 3: 2]#		12268005	351022452	2
3	12268010	% RESETF = [ 0: 1]#		12268010	351022452	3
4	12268012	%		12268012	351022452	4
5	12268014	%		12268014	351022452	5
6	12268016	%	<<< ON THE INTERPRETATION OF THE FIELDS OF THE BEDWORD .... >>>	12268016	351022452	6
7	12268018	%		12268018	351022452	7
8	12268020	%	THE BEDWORD OF THE PIB IS SET UP EITHER WHEN A STACK IS WAITING	12268020	351022452	8
9	12268024	%	FOR AN EVENT OR IT IS LINKED IN THE CALENDAR QUEUE. THE CALENDAR	12268024	350711819	9
10	12268026	%	QUEUE CONSISTS OF STACKS THAT NEED TO BE AWAKENED AFTER A SPECIFIED	12268026	350711819	10
11	12268028	%	AMOUNT OF TIME HAS ELAPSED.	12268028	350711819	11
12	12268030	%		12268030	350711819	12
13	12268032	%	WAITING FOR A SINGLE EVENT CAN OCCUR EITHER FROM MULTIPLEWAIT	12268032	350711819	13
14	12268034	%	OR FROM ANY ONE OF OTHER PROCEDURES LIKE WAITP, PROCUREP ETC.	12268034	350711819	14
15	12268036	%	A DISTINCTION NEEDS TO BE MADE BETWEEN THE TWO CASES. IN THE	12268036	350711819	15
16	12268038	%	MULTIPLEWAIT CASE, WHEN THE EVENT IS CAUSED, MULTIPLEDELINK NEEDS TO	12268038	350711819	16
17	12268040	%	BE CALLED TO RETURN THE RESULT OF THE WAIT FUNCTION. NO SUCH CALL	12268040	350711819	17
18	12268042	%	NEEDS TO BE MADE IN THE OTHER INSTANCES OF WAITING FOR A SINGLE	12268042	350711819	18
19	12268044	%	EVENT. ON THE OTHER HAND, IF SUCH A WAITING STACK HAS TO BE DS-ED,	12268044	350711819	19
20	12268046	%	MULTIPLEDELINK HAS TO BE CALLED IN ALL THE ABOVE INSTANCES, TO	12268046	350711819	20
21	12268048	%	DELINK THE SINGLE EVENT. MULTIPLEDELINK KNOWS THAT AN EVENT NEEDS TO	12268048	350711819	21
22	12268050	%	BE DELINKED EITHER IF THE EVENTCOUNTF FIELD IS NON-ZERO OR IF IT	12268050	350711819	22
23	12268052	%	DETERMINES THAT TIME DID NOT HAVE TO BE DELINKED ( THE ASSUMPTION	12268052	350711819	23
24	12268054	%	BEING THAT MULTIPLEDELINK IS CALLED TO DELINK AT LEAST ONE OF EITHER	12268054	350711819	24
25	12268056	%	TIME OR EVENTS).	12268056	350711819	25
26	12268058	%		12268058	350711819	26
27	12268060	%	WAITING FOR ONLY TIME (NO EVENTS) CAN OCCUR EITHER FROM	12268060	350711819	27
28	12268062	%	MULTIPLEWAIT, OR FROM DMSWAIT. EVEN HERE, A DISTINCTION NEEDS TO BE	12268062	350711819	28
29	12268064	%	MADE. IN THE MULTIPLEWAIT CASE WHEN THE SPECIFIED TIME HAS ELAPSED	12268064	350711819	29
30	12268066	%	MULTIPLEDELINK MUST BE CALLED. IN THE OTHER CASE NO SUCH CALL NEEDS	12268066	350711819	30
31	12268068	%	TO BE MADE. ON THE OTHER HAND WE NEED TO MAINTAIN INFORMATION IN	12268068	350711819	31
32	12268070	%	BOTH THE ABOVE CASES THAT THE STACKS HAVE BEEN LINKED IN THE	12268070	350711819	32
33	12268072	%	CALENDAR. THIS IS NECESSARY BECAUSE, DELINKING NEEDS TO BE DONE IN	12268072	350711819	33
34	12268074	%	BOTH CASES IF THE STACKS ARE BEING DS-ED. THE ABOVE CONSTRAINTS	12268074	350711819	34
35	12268076	%	ARE SATISFIED BY SETTING THE EVENTCOUNTF TO ZERO FOR THE DMSWAIT	12268076	351022452	35
36	12268078	%	CASE, SINCE THE EVENTCOUNTF IS ALWAYS GREATER THAN ZERO FOR THE	12268078	351022452	36
37	12268080	%	MULTIPLEWAIT CASE.	12268080	351022452	37
38	12268090	%		12268090	350711819	38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	A T T R I B U T E S	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	A P P E N D I X D	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1	SYMBOL/ATTABLEGEN (03/20/85)	11:59 AM TUESDAY, APRIL 9, 1985	1
2			2
3			3
4	177000 FILE ATTRIBUTE SECTION.	00177000	4
5	177500	00177500 32.123.013	5
6	178000 TITLE 0 <NAME> PRIORITY HIGH WFL COMPILERS	00178000 29.012.008	6
7	179000 PARAMETERS 10PT,	00179000	7
8	180000 FILESECTION 1 <INTEGER> WFL	00180000 29.012.008	8
9	181000 COMPILERS ,	00181000 29.012.008	9
10	182000 REEL = FILESECTION,	00182000	10
11	183000 CREATIONDATE 2 <INTEGER> WFL	00183000 29.012.008	11
12	184000 COMPILERS READONLY ,	00184000 29.012.008	12
13	185000 DATE = CREATIONDATE,	00185000	13
14	186000 CYCLE 3 <INTEGER> WFL COMPILERS	00186000 29.012.008	14
15	187000 CATALOG LIBMAINT ,	00187000 29.012.008	15
16	188000 VERSION 4 <INTEGER> WFL COMPILERS	00188000 29.012.008	16
17	189000 CATALOG LIBMAINT ,	00189000 29.012.008	17
18	190000 SAVEFACTOR 5 <INTEGER> WFL	00190000 29.012.008	18
19	191000 COMPILERS ,	00191000 29.012.008	19
20	192000 DENSITY 6 <INTEGER> MNEMONIC BPI200/BPI556/BPI800/BPI1600/BPI6250/	00192000 31.157.015	20
21	193000 LOW/MEDIUM/HIGH/SUPER WFL COMPILERS ,	00193000 31.157.015	21
22	194000 PARITY 7 <INTEGER> MNEMONIC STANDARD/NONSTANDARD/ODD/EVEN	00194000 35.028.008	22
23	195000 WFL COMPILERS ,	00195000 29.012.008	23
24	196000 KIND CASE 8 <KIND> PRIORITY HIGH MNEMONIC	00196000 29.012.008	24
25	196100 DONTCARE SA/NOUNIT SA/DISK SA/ODT SA/	00196100 35.039.013	25
26	197000 REMOTE SA/PAPERREADER SA/PAPERPUNCH SA/PRINTER SA/	00197000 35.151.028	26
27	197050 HY SA/READER SA/	00197050 35.151.028	27
28	197100 PUNCH SA/DISKETTE SA/	00197100 30.031.008	28
29	197200 PORT SA/HC SA/	00197200 34.058.012	29
30	197500 ASID SA/FPSAP SA/	00197500 35.079.023	30
31	197550 CSMA SA/	00197550 35.074.020	31
32	197600 SPC SA/	00197600 35.038.011	32
33	198000 TAPE7 SA/TAPE9 SA/TAPEPE SA/TAPE SA/PACK SA	00198000 34.043.010	33
34	198010 WFL	00198010 29.012.008	34
35	199000 COMPILERS CATALOG VOLUME LIBMAINT ,	00199000 29.012.008	35
36	200000 LABEL 9 <INTEGER> MNEMONIC OMITTED/STANDARD/	00200000 29.012.008	36
37	201000 OMITTEDEF WFL COMPILERS ,	00201000 29.012.008	37
38	202000 LABELTYPE = LABEL,	00202000	38
39	203000 EXTMODE 10 <INTEGER> PRIORITY HIGH MNEMONIC SINGLE/HEX/	00203000 29.012.008	39
40	204000 BCL/EBCDIC/ASCII/BINARY WFL COMPILERS ,	00204000 29.012.008	40
41	205000 OPTIONAL 11 <BOOLEAN> MNEMONIC TRUE/FALSE/	00205000 29.012.008	41
42	206000 <NUL TRUE> WFL COMPILERS ,	00206000 29.012.008	42
43	207000 PROTECTION 12 <INTEGER> MNEMONIC TEMPORARY/	00207000 29.012.008	43
44	208000 SAVE/PROTECTED WFL COMPILERS ,	00208000 29.012.008	44
45	209000 FILETYPE 13 <INTEGER> PRIORITY HIGH WFL	00209000 29.012.008	45
46	210000 COMPILERS ,	00210000 29.012.008	46
47	211000 BLOCKSIZE 14 <INTEGER> PRIORITY HIGH WFL	00211000 29.012.008	47
48	212000 COMPILERS ,	00212000 35.001.001	48
49	213000 MAXRECSIZE 15 <INTEGER> PRIORITY HIGH WFL	00213000 29.012.008	49
50	214000 COMPILERS PARAMETERS 10PT ,	00214000 32.052.007	50
51	215000 MINRECSIZE 16 <INTEGER> WFL	00215000 29.012.008	51
52	216000 COMPILERS ,	00216000 29.012.008	52
53	217000 AREASIZE 17 <INTEGER> PRIORITY HIGH WFL	00217000 29.012.008	53
54	218000 COMPILERS ,	00218000 29.012.008	54
55	219000 AREAS 18 <INTEGER> PRIORITY HIGH WFL COMPILERS,	00219000 29.012.008	55
56	220000 APL 19 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> COMPILERS,	00220000 33.046.012	56
57	221000 MYUSE 20 <INTEGER> PRIORITY HIGH MNEMONIC CLOSED/IN/OUT/IO	00221000 29.012.008	57
58	222000 WFL COMPILERS ,	00222000 29.012.008	58
59	223000 OTHERUSE 21 <INTEGER> MNEMONIC SECURED/IN/	00223000 29.012.008	59
60	224000 OUT/IO WFL COMPILERS ,	00224000 29.012.008	60

1	225000	FLEXIBLE 22 <BOOLEAN> PRIORITY HIGH MNEMONIC TRUE/FALSE/	00225000	29.012.008	1
2	226000	<NUL TRUE> WFL COMPILERS ,	00226000	29.012.008	2
3	227000	IAD 23 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> COMPILERS READONLY,	00227000	29.012.008	3
4	230000	BUFFERS 26 <INTEGER> WFL COMPILERS,	00230000		4
5	231000	DIRECTION 27 <INTEGER> MNEMONIC REVERSE/FORWARD WFL COMPILERS,	00231000		5
6	232000	CURRENTBLOCK 28 <INTEGER> COMPILERS READONLY,	00232000		6
7	233000	INTMODE 29 <INTEGER> MNEMONIC SINGLE/HEX/BCL/EBCDIC/ASCII WFL	00233000		7
8	234000	COMPILERS,	00234000		8
9	235000	OPEN 30 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> COMPILERS,	00235000	33.046.012	9
10	236000	PRESENT 31 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00236000		10
11	237000	SIZEMODE 32 <INTEGER> MNEMONIC SINGLE/HEX/BCL/EBCDIC/ASCII WFL	00237000		11
12	238000	COMPILERS,	00238000		12
13	239000	SIZEOFFSET 33 <INTEGER> WFL COMPILERS,	00239000		13
14	240000	SIZE2 34 <INTEGER> WFL COMPILERS,	00240000		14
15	241000	STATE 35 <REAL> COMPILERS READONLY PARAMETERS 10PT,	00241000	32.052.007	15
16	242000	EOF 36 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00242000		16
17	242700	DMAREALOCK 38 <INTEGER> COMPILERS PARAMETERS 1REQ10PT,	00242700	32.141.014	17
18	242800	DMROWLOCK = DMAREALOCK,	00242800	32.141.014	18
19	243000	FAMILYNAME 39 <IDENTIFIER> PRIORITY HIGH WFL COMPILERS CATALOG,	00243000	003	19
20	244000	PACKNAME = FAMILYNAME,	00244000		20
21	245000	SINGLEUNIT 40 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00245000		21
22	246000	LIBMAINT,	00246000	29.012.008	22
23	247000	SINGLEPACK = SINGLEUNIT,	00247000		23
24	248000	CYLINDERMODE 41 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00248000		24
25	249000	RESIDENT 42 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00249000		25
26	250000	BLOCK 44 <INTEGER> COMPILERS READONLY,	00250000		26
27	251000	CARRIAGECONTROL 45 <INTEGER> MNEMONIC STANDARD/CTLASA/CTL360 WFL	00251000		27
28	252000	COMPILERS,	00252000		28
29	253000	RECORD 46 <INTEGER> COMPILERS READONLY,	00253000		29
30	254000	UPDATEFILE 47 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00254000		30
31	255000	AVAILABLE 48 <INTEGER> COMPILERS READONLY,	00255000		31
32	256000	LASTRECORD 49 <INTEGER> COMPILERS,	00256000		32
33	257000	PAGESIZE 50 <INTEGER> WFL COMPILERS PARAMETERS 10PT,	00257000		33
34	258000	PAGE 51 <INTEGER> COMPILERS,	00258000		34
35	259000	LINENUM 52 <INTEGER> COMPILERS,	00259000		35
36	260000	FORMID 54 <STRING> WFL COMPILERS,	00260000	29.053.020	36
37	261000	FORMMESSAGE = FORMID,	00261000		37
38	262000	FORMESSAGE =FORMMESSAGE,	00262000		38
39	263000	UPDATED 57 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00263000		39
40	264000	FILEKIND 58 <INTEGER> MNEMONIC NULLFILE/DIRECTORY/SYSTEMDIRECTORY/	00264000	29.023.006	40
41	264100	VERSIONDIRECTORY/	00264100	29.023.006	41
42	265000	CATALOG/BACKUPDISK/RECONSTRUCTIONFILE/SYSTEMDIRFILE/	00265000	29.023.006	42
43	265500	ARCHIVELOG/BOOTCODE/MDLCODE/MDLSYMBOL/	00265500	35.073.019	43
44	265600	REMOTEAUDIT/REMOTEBACKUP/	00265600	31.143.013	44
45	266000	JOBDESCFILE/XDISKFILE/COMPILERCODEFILE/CHECKPOINTFILE/CPJOBFILE/	00266000		45
46	267000	DCPCODE/NDLCODE/LIBRARYCODE/INTRINSICFILE/MCPCODEFILE/ALGOLCODE/	00267000		46
47	268000	COBOLCODE/FORTRANCODE/XALGOLCODE/PLICODE/JOVIALCODE/ESPOLCODE/	00268000		47
48	269000	DCALGOLCODE/BASICCODE/XFORTRANCODE/JOBCODE/BOUNDCODE/CODEFILE/	00269000		48
49	270000	ALGOLSYMBOL/COBOLSYMBOL/FORTRANSYMBOL/XALGOLSYMBOL/PLISYMBOL/	00270000		49
50	271000	JOVIALSYMBOL/ESPOLSYMBOL/DCALGOLSYMBOL/BASICSYMBOL/XFORTRANSYMBOL/	00271000		50
51	272000	BINDERSYMBOL/DASDLSYMBOL/DMALGOLSYMBOL/DCPSYMBOL/NDLSYMBOL/DATA/	00272000		51
52	273000	SEQDATA/GUARDFILE/APLDATA/APLWORKSOURCE/CDATA/CSEQDATA/	00273000		52
53	273500	RECOVERYFILE/DMALGOLCODE/JOBSYMBOL/	00273500	29.004.001	53
54	273600	BACKUPPRINTER/INFOFILE/	00273600	34.043.010	54
55	273700	VFORTRANSYMBOL/VFORTRANCODE/VLINKEDCODE/VMCPCODE/	00273700	29.051.016	55
56	273710	LCOBOLSYMBOL/LCOBOLSL3CODE/LCOBOLSL5CODE/	00273710	33.003.002	56
57	273720	SFORTRANSYMBOL/ SFORTRANCODE/ SLINKEDCODE/ SMCPCODE/	00273720	33.003.002	57
58	273730	FIRMWARE/	00273730	35.073.019	58
59	273740	NEWPSYMBOL/NEWPCODE/	00273740	33.003.002	59
60	273745	PAGEBACKUP/FONT/FORM/GRAPHICS/	00273745	34.042.008	60

1	273750	SANSSYMBOL/SANSCODE/	00273750	33.003.002	1
2	273760	PASCALSYMBOL/PASCALCODE/	00273760	33.003.002	2
3	273770	RPGSYMBOL/RPGCODE/	00273770	33.003.002	3
4	273780	NDLIISYMBOL/NDLIICODE/	00273780	33.003.002	4
5	273790	FORTRAN77SYMBOL/FORTRAN77CODE/	00273790	33.003.002	5
6	273800	SORTSYMBOL/SORTCODE/	00273800	34.005.001	6
7	273847	MDTTEST/MDTDUMP/PROMBURNERDATA/CONFIGURATIONDATA/CONFIDENCECODE/	00273847	32.7X.001	7
8	273850	BDDATA/MP1FIRMWARE/MP1ALGOLCODE/	00273850	33.101.013	8
9	273900	COBOL74SYMBOL/COBOL74CODE/	00273900	34.017.003	9
10	273950	OHNESYMBOL/TEXTDATA/PRINTERCONTROLFILE/	00273950	35.015.006	10
11	274000	DBRESTARTSET/DBDATA/BACKUPPUNCH/SCHEDULEFILE COMPILERS WFL,	00274000	29.017.004	11
12	275000	CRUNCHED 59 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS	00275000		12
13	276000	READONLY,	00276000		13
14	277000	ROWSINUSE 60 <INTEGER> COMPILERS READONLY,	00277000		14
15	278000	USEDATE 61 <INTEGER> COMPILERS READONLY,	00278000		15
16	279000	ROWADDRESS 64 <INTEGER> COMPILERS READONLY PARAMETERS 1REQ1OPT,	00279000	29.012.008	16
17	280000	INTERCHANGE 66 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00280000		17
18	281000	LIBMAINT,	00281000	29.012.008	18
19	282000	IC = INTERCHANGE,	00282000		19
20	283000	DUPLICATED 67 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00283000		20
21	284000	COPIES 69 <INTEGER> WFL COMPILERS,	00284000		21
22	285000	INTNAME 72 <IDENTIFIER> COMPILERS,	00285000	003	22
23	286000	UNITS 73 <INTEGER> MNEMONIC CHARACTERS/WORDS/TRUE/FALSE/<NUL CHARACTERS>	00286000		23
24	287000	WFL COMPILERS,	00287000		24
25	288000	ATTERR 74 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00288000		25
26	289000	ATTTYPE 75 <INTEGER> MNEMONIC * COMPILERS READONLY,	00289000	29.023.007	26
27	290000	ATTVALUE 76 <REAL> COMPILERS READONLY,	00290000		27
28	291000	IOCLOCKS 77 <REAL> COMPILERS READONLY,	00291000		28
29	292000	UNITNO 78 <INTEGER> WFL COMPILERS CATALOG LIBMAINT,	00292000	29.012.008	29
30	293000	SECURITYGUARD 79 <NAME> WFL COMPILERS,	00293000		30
31	294000	SECURITYTYPE 80 <INTEGER> MNEMONIC PRIVATE/PUBLIC/GUARDED/CONTROLLED	00294000	31.041.003	31
32	295000	WFL COMPILERS,	00295000	31.041.003	32
33	296000	SECURITYUSE 81 <INTEGER> MNEMONIC SECURED/IN/OUT/IO WFL COMPILERS,	00296000		33
34	299000	READCHECK 83 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00299000		34
35	300000	RECORDINERROR 84 <INTEGER> COMPILERS READONLY,	00300000		35
36	301000	ERRORTYPE 85 <INTEGER> MNEMONIC NOERROR/SUNOTREADY/READPARITYERROR/	00301000		36
37	302000	READCHECKFAILURE COMPILERS READONLY PARAMETERS 1OPT,	00302000		37
38	303000	IOINERROR 87 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00303000		38
39	304000	TAPEREELRECORD 88 <INTEGER> COMPILERS READONLY,	00304000		39
40	304500	DMTIMESTAMP 89 <REAL> COMPILERS,	00304500	29.041.015	40
41	305000	EXCLUSIVE 90 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00305000		41
42	306000	TRANSLATE 91 <INTEGER> MNEMONIC DEFAULTTRANS/FULLTRANS/SOFTONLY/	00306000		42
43	307000	FORCESOFT/NOSOFT/NOTRANS WFL COMPILERS,	00307000		43
44	308000	TRANSLATING 92 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY,	00308000		44
45	309000	INPUTTABLE 93 <TRANSLATIONTABLE> COMPILERS WRITEONLY,	00309000	29.012.008	45
46	310000	OUTPUTTABLE 94 <TRANSLATIONTABLE> COMPILERS WRITEONLY,	00310000	29.012.008	46
47	311000	USECATALOG 95 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00311000		47
48	312000	GENERATION 96 <INTEGER> WFL CATALOG LIBMAINT COMPILERS,	00312000	29.012.008	48
49	313000	SENSITIVEDATA 97 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL	00313000		49
50	314000	LIBMAINT COMPILERS,	00314000	29.012.008	50
51	315000	FAMILYINDEX 98 <INTEGER> WFL LIBMAINT COMPILERS PARAMETERS 2OPT,	00315000	29.053.020	51
52	317000	STATIONCOUNT 99 <INTEGER> COMPILERS READONLY,	00317000		52
53	318000	FAMILYSIZE = STATIONCOUNT,	00318000		53
54	319000	POPULATION 100 <INTEGER> COMPILERS READONLY,	00319000		54
55	320000	STATIONLIST 101 <SIMPLENAME> COMPILERS WRITEONLY,	00320000	31.215.029	55
56	321000	FAMILY = STATIONLIST,	00321000		56
57	322000	ENABLEINPUT 102 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS	00322000		57
58	323000	READONLY PARAMETERS 1OPT,	00323000		58
59	324000	STATIONSDENIED 103 <INTEGER> COMPILERS READONLY,	00324000		59
60	325000	TRAINID 104 <INTEGER> MNEMONIC ASCII72/BCL64/EBCDIC18/EBCDIC48/	00325000		60

1	326000	EBCDIC72/EBCDIC96/NOID WFL COMPILERS,	00326000		1
2	327000	TIMELIMIT 105 <REAL> WFL COMPILERS,	00327000		2
3	328000	LASTSUBFILE 106 <INTEGER> COMPILERS,	00328000	32.052.007	3
4	328500	LASTSTATION = LASTSUBFILE,	00328500	32.052.007	4
5	329000	RECEPTIONS 107 <INTEGER> COMPILERS READONLY PARAMETERS 10PT,	00329000		5
6	330000	CENSUS 108 <INTEGER> COMPILERS READONLY PARAMETERS 10PT,	00330000	29.051.019	6
7	330100	NORESOURCEWAIT 109 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE>	00330100	004	7
8	330200	WFL COMPILERS,	00330200	004	8
9	331000	DISPOSITION 111 <INTEGER> COMPILERS READONLY PARAMETERS 10PT,	00331000	32.052.007	9
10	332000	TRANSMISSIONS 112 <INTEGER> COMPILERS READONLY PARAMETERS 10PT,	00332000		10
11	333000	ASSIGNTIME 113 <INTEGER> COMPILERS READONLY PARAMETERS 1REQ,	00333000		11
12	333100	TANKING 114 <INTEGER> MNEMONIC UNSPECIFIED/NONE/SYNC/ASYNC	00333100	29.038.014	12
13	333120	COMPILERS WFL,	00333120	29.035.014	13
14	334000	SCREEN 116 <BOOLEAN> MNEMONIC TRUE/FALSE COMPILERS READONLY	00334000		14
15	335000	PARAMETERS 1REQ,	00335000		15
16	336000	TRANSMISSIONNO 117 <INTEGER> COMPILERS READONLY PARAMETERS 1REQ,	00336000	31.215.029	16
17	337000	TRANSMISSIONO = TRANSMISSIONNO,	00337000	31.215.029	17
18	338000	WIDTH 118 <INTEGER> COMPILERS READONLY PARAMETERS 1REQ,	00338000		18
19	339000	SERIALNO 119 <SERIALNO> COMPILERS WFL LIBMAINT CATALOG VOLUME	00339000	29.053.020	19
20	339010	PARAMETERS 10PT,	00339010	29.053.020	20
21	357020	BACKUPKIND 135 <INTEGER> MNEMONIC	00357020	29.051.017	21
22	357040	DONTCARE/ DISK/ PACK/ TAPE/ TAPE7/ TAPE9/ TAPEPE	00357040	31.215.029	22
23	357060	WFL COMPILERS,	00357060	30.021.006	23
24	357100	NEWFILE 136 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00357100	31.057.007	24
25	357110	EXTENT 137 <INTEGER> WFL COMPILERS,	00357110	30.031.008	25
26	357120	WRITECHECK 138 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS,	00357120	30.031.008	26
27	357130	CURRENTTEXT 139 <INTEGER> WFL COMPILERS READONLY,	00357130	30.031.008	27
28	357140	MYNAME 140 <NAME> WFL COMPILERS,	00357140	30.031.007	28
29	357150	YOURNAME 141 <NAME> WFL COMPILERS PARAMETERS 10PT,	00357150	32.052.007	29
30	357160	FILEUSE 142 <INTEGER> PRIORITY HIGH MNEMONIC IN/OUT/IO	00357160	31.092.008	30
31	357170	WFL COMPILERS,	00357170		31
32	357180	HOSTNAME 143 <IDENTIFIER> WFL COMPILERS LIBMAINT PARAMETERS 10PT,	00357180	32.052.007	32
33	357190	FILEORGANIZATION 144 <INTEGER> MNEMONIC NOTRESTRICTED/RELATIVE/INDEXED/	00357190	33.121.017	33
34	357195	INDEXEDNOTRESTRICTED/PLIISAM	00357195	35.039.012	34
35	357200	WFL COMPILERS,	00357200	31.167.017	35
36	357210	HSDIALOGNO 145 <INTEGER> COMPILERS,	00357210	31.186.025	36
37	357220	TIMESTAMP 146 <REAL> COMPILERS READONLY,	00357220	32.141.014	37
38	357230	HSFILECOPY 147 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> COMPILERS	00357230	32.141.014	38
39	357240	READONLY,	00357240	32.141.014	39
40	357250	KEYEDIOKEYS 148 <REAL> COMPILERS WRITEONLY	00357250	35.047.014	40
41	357260	,ACCESSMODE 149 <INTEGER> MNEMONIC SEQUENTIALACCESS/	00357260	32.141.014	41
42	357270	RANDOMACCESS/DYNAMICACCESS COMPILERS	00357270	32.141.014	42
43	357280	WRITEONLY	00357280	32.141.014	43
44	357290	,MAXSUBFILES 150 <INTEGER> WFL COMPILERS	00357290	32.141.014	44
45	357300	,MYHOSTNAME 151 <IDENTIFIER> WFL COMPILERS READONLY	00357300	32.141.014	45
46	357310	,INPUTEVENT 152 <EVENT> COMPILERS READONLY PARAMETERS 10PT	00357310	32.141.014	46
47	357320	,OUTPUTEVENT 153 <EVENT> COMPILERS READONLY PARAMETERS 10PT	00357320	32.141.014	47
48	357330	,CHANGEVENT 154 <EVENT> COMPILERS READONLY PARAMETERS 10PT	00357330	32.143.015	48
49	357340	,YOURUSERCODE 155 <IDENTIFIER> WFL COMPILERS PARAMETERS 10PT	00357340	32.141.014	49
50	357350	,COMPRESSION 156 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357350	32.141.014	50
51	357360	PARAMETERS 10PT	00357360	32.141.014	51
52	357370	,MAXCENSUS 157 <INTEGER> WFL COMPILERS PARAMETERS 10PT	00357370	32.141.014	52
53	357380	,CHANGEDSUBFILE 158 <INTEGER> COMPILERS READONLY	00357380	32.141.014	53
54	357390	,AREALength 159 <INTEGER> PRIORITY HIGH WFL COMPILERS	00357390	32.141.014	54
55	357400	,AREAALLOCATED 160 <BOOLEAN> MNEMONIC TRUE/FALSE WFL COMPILERS READONLY	00357400	32.141.014	55
56	357410	PARAMETERS 1REQ10PT	00357410	32.141.014	56
57	357420	,FRAME SIZE 161 <INTEGER> PRIORITY HIGH WFL COMPILERS	00357420	32.141.014	57
58	357430	,FILENAME 162 <NAME> PRIORITY HIGH WFL COMPILERS	00357430	32.141.014	58
59	357440	,COPYNAME 163 <NAME> COMPILERS READONLY PARAMETERS 1REQ	00357440	32.141.014	59
60	357450	,STATIONNAME 164 <NAME> COMPILERS READONLY PARAMETERS 1REQ	00357450	32.141.014	60



1	357460	,BLOCKSTRUCTURE 165 <INTEGER> PRIORITY HIGH	00357460	32.141.014	1
2	357470	MNEMONIC FIXED/EXTERNAL/VARIABLE/LINKED WFL COMPILERS	00357470	32.141.014	2
3	357480	,DEPENDENTSPECS 166 <BOOLEAN> PRIORITY HIGH	00357480	32.141.014	3
4	357490	MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357490	32.141.014	4
5	357500	,SCREENSIZE 167 <INTEGER> COMPILERS READONLY PARAMETERS 1REQ	00357500	32.141.014	5
6	357510	,CURRENTRECORD 168 <INTEGER> COMPILERS READONLY	00357510	32.143.015	6
7	357520	,SUBFILEERROR 169 <INTEGER> MNEMONIC NOERROR/DISCONNECTED/DATALOST/	00357520	32.143.015	7
8	357530	NOBUFFER/NOFILEFOUND/UNREACHABLEHOST/UNSUPPORTEDFUNCTION	00357530	33.111.016	8
9	357535	COMPILERS READONLY PARAMETERS 1OPT	00357535	33.111.016	9
10	357540	,FILESTATE 170 <INTEGER> MNEMONIC CLOSED/AWAITINGHOST/OFFERED/OPENED/	00357540	32.143.015	10
11	357550	SHUTTINGDOWN/BLOCKED/CLOSEPENDING/DEACTIVATIONPENDING/DEACTIVATED/	00357550	32.143.015	11
12	357560	DENIED/POSTPONED/DENIEDILLEGALUSE	00357560	32.143.015	12
13	357570	COMPILERS READONLY PARAMETERS 1OPT	00357570	32.143.015	13
14	357580	,NEXTRECORD 171 <INTEGER> COMPILERS READONLY	00357580	33.027.005	14
15	357590	,LABELKIND 172 <INTEGER> COMPILERS READONLY	00357590	33.027.005	15
16	357600	,STATIONSALLOWED 173 <INTEGER> COMPILERS READONLY	00357600	33.027.005	16
17	357610	,COPYINERROR 174 <INTEGER> COMPILERS READONLY	00357610	33.027.005	17
18	357615	,USETIME 175 <INTEGER> COMPILERS READONLY	00357615	33.030.007	18
19	357620	,USERINFO 176 <REAL> COMPILERS	00357620	33.030.007	19
20	357625	,ALTERDATE 177 <INTEGER> COMPILERS READONLY	00357625	33.030.007	20
21	357630	,ALERTIME 178 <INTEGER> COMPILERS READONLY	00357630	33.030.007	21
22	357635	,CREATIONTIME 179 <INTEGER> COMPILERS READONLY	00357635	33.030.007	22
23	357640	,BLANK 180 <INTEGER> MNEMONIC NULL/ZERO COMPILERS	00357640	34.005.002	23
24	357645	,PRINTERKIND 181 <INTEGER> MNEMONIC DONTCARE/IMAGEPRINTER/LINEPRINTER	00357645	35.015.006	24
25	357650	WFL COMPILERS	00357650	35.015.006	25
26	357655	,PRINTERCONTROL 182 <NAME> WFL COMPILERS	00357655	35.015.006	26
27	357660	,SIZEVISIBLE 183 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE>	00357660	35.035.009	27
28	357665	WFL COMPILERS	00357665	35.035.009	28
29	357670	,AFTER 184 <STRING> WFL COMPILERS	00357670	35.076.021	29
30	357675	,ALIGNFILE 185 <NAME> WFL COMPILERS	00357675	35.076.021	30
31	357680	,ALIGNMENT 186 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357680	35.076.021	31
32	357685	,BANNER 187 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357685	35.076.021	32
33	357690	,PRINTCHARGE 188 <SIMPLENAME> WFL COMPILERS	00357690	35.151.026	33
34	357695	,CHARSET 189 <SIMPLENAME> WFL COMPILERS	00357695	35.076.021	34
35	357700	,CHECKPOINT 190 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357700	35.076.021	35
36	357705	,DESTINATION 191 <STRING> WFL COMPILERS	00357705	35.076.021	36
37	357710	,NOTE 192 <STRING> WFL COMPILERS	00357710	35.076.021	37
38	357715	,PRINTCOPIES 193 <INTEGER> WFL COMPILERS	00357715	35.076.021	38
39	357720	,PRINTDISPOSITION 194 <INTEGER>	00357720	35.076.021	39
40	357725	MNEMONIC DONTPRINT/DIRECT/CLOSE/EOT/EOJ	00357725	35.076.021	40
41	357730	WFL COMPILERS	00357730	35.076.021	41
42	357735	,SAVEBACKUPFILE 195 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE>	00357735	35.113.025	42
43	357736	WFL COMPILERS	00357736	35.113.025	43
44	357740	,TRANSFORM 196 <STRING> WFL COMPILERS	00357740	35.076.021	44
45	357745	,USERBACKUPNAME 197 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE>	00357745	35.076.021	45
46	357750	WFL COMPILERS	00357750	35.076.021	46
47	357755	,DUMMYFILE 198 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357755	35.076.021	47
48	357760	,TRIMBLANKS 199 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00357760	35.077.022	48
49	357765	,OLDYOURUSERCODE 200 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE>	00357765	35.093.024	49
50	357770	WFL COMPILERS	00357770	35.093.024	50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1	SYMBOL/ATTABLEGEN (03/20/85)		11:59 AM TUESDAY, APRIL 9, 1985		1
2					2
3					3
4	358000	TASK ATTRIBUTE SECTION.		00358000	4
5	358500			00358500	5
6	359000	NAME 0 <NAME> WFL COMPILERS,		00359000	6
7	361000	MIXNUMBER 1 <INTEGER> WFL COMPILERS READONLY,		00361000	7
8	362000	STACKNO = MIXNUMBER,		00362000	8
9	363000	CORE 2 <CORE> PRIORITY HIGH WFL COMPILERS,		00363000	9
10	365000	COREESTIMATE = CORE,		00365000	10
11	366000	PRIORITY 3 <INTEGER> PRIORITY HIGH WFL COMPILERS,		00366000	11
12	368000	DECLAREDPRIORITY = PRIORITY,		00368000	12
13	369000	MAXPROCTIME 4 <INTEGER> PRIORITY HIGH WFL		00369000	13
14	370000	COMPILERS,		00370000	14
15	371000	MAXIOTIME 5 <INTEGER> WFL COMPILERS,		00371000	15
16	374000	IOTIME = MAXIOTIME,		00374000	16
17	375000	TARGET 6 <INTEGER> WFL COMPILERS,		00375000	17
18	377000	TARGETTIME = TARGET,		00377000	18
19	378000	STACK 7 <INTEGER> WFL COMPILERS,		00378000	19
20	380000	STACKSIZE = STACK,		00380000	20
21	381000	USERCODE 8 <SIMPLENAME> COMPILERS,		00381000	21
22	383000	TASKVALUE 9 <REAL> PRIORITY HIGH WFL COMPILERS,		00383000	22
23	385000	VALUE = TASKVALUE,		00385000	23
24	386000	HISTORY 10 <REAL> WFL COMPILERS READONLY,		00386000	24
25	391000	TYPE 11 <REAL> MNEMONIC PROCESS/CALL/RUN/JOBSTACK WFL COMPILERS		00391000	25
26	392000	READONLY,		00392000	26
27	393000	STATUS 12 <INTEGER> MNEMONIC		00393000	27
28	393500	NEVERUSED		00393500	28
29	393600	/SCHEDULED		00393600	29
30	393700	/ACTIVE		00393700	30
31	393800	/SUSPENDED		00393800	31
32	393900	/FROZEN		00393900	32
33	394000	/GOINGAWAY		00394000	33
34	394100	/UNINITIATED		00394100	34
35	394200	/TERMINATED		00394200	35
36	394300	/BADINITIATE		00394300	36
37	394400	WFL COMPILERS,		00394400	37
38	395000	ACCUMPROCTIME 13 <INTEGER> WFL COMPILERS READONLY,		00395000	38
39	395100	PROCESSTIME = ACCUMPROCTIME,		00395100	39
40	396000	ACCUMIOTIME 14 <INTEGER> WFL COMPILERS READONLY,		00396000	40
41	396100	PROCESSIONTIME = ACCUMIOTIME,		00396100	41
42	397000	ELAPSEDTIME 15 <INTEGER> WFL COMPILERS READONLY,		00397000	42
43	398000	EXCEPTIONTASK 16 <TASK> COMPILERS,		00398000	43
44	399000	LOCKED 17 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> COMPILERS,		00399000	44
45	400000	STOPPOINT 18 <REAL> WFL COMPILERS READONLY,		00400000	45
46	401000	PARTNER 19 <TASK> COMPILERS,		00401000	46
47	402000	STATION 20 <REAL> WFL COMPILERS,		00402000	47
48	403000	INITIATOR = STATION,		00403000	48
49	404000	EXCEPTIONEVENT 21 <EVENT> COMPILERS,		00404000	49
50	405000	OPTION 22 <OPTION> MNEMONIC */LONG/FAULT DP/DSED DP/CCLIST/BACKUP/		00405000	50
51	405100	BDBASE/SIBS DP/LIBRARIES DP/PRIVATELIBRARIES DP/		00405100	51
52	405200	DEBUG/		00405200	52
53	406000	BASE DP/ARRAYS DP/CODE DP/FILES DP/NOSUMMARY/HARDDUMP/DBS DP/		00406000	53
54	406500	AUTORM/IPCOVERRIDE		00406500	54
55	407000	WFL COMPILERS,		00407000	55
56	408000	OPTIONS = OPTION,		00408000	56
57	409000	VALIDITYBITS 23 <BOOLEAN> MNEMONIC * COMPILERS		00409000	57
58	410000	READONLY,		00410000	58
59	411000	FILECARDS 24 <ATTRIBUTELIST> WFL COMPILERS,		00411000	59
60	412000	FILE = FILECARDS,		00412000	60

1	413000	ERROR 25 <REAL> MNEMONIC * WFL COMPILERS,	00413000	29.023.007	1
2	414000	TASKATTERR = ERROR,	00414000		2
3	414500	PARTNEREXISTS 27 <BOOLEAN> COMPILERS READONLY,	00414500	34.062.015	3
4	415000	RESTART 28 <INTEGER> WFL COMPILERS,	00415000		4
5	416000	BDNAME 29 <SIMPLENAME> WFL COMPILERS,	00416000	003	5
6	417000	BACKUPPREFIX = BDNAME,	00417000		6
7	418000	STACKHISTORY 30 <STRING> COMPILERS READONLY,	00418000		7
8	419000	SUBSPACES 31 <INTEGER> WFL COMPILERS,	00419000		8
9	420000	TASKFILE 32 <TASKFILE> COMPILERS,	00420000		9
10	421000	DECKGROUPNO 33 <INTEGER> COMPILERS READONLY,	00421000		10
11	422000	CLASS 34 <INTEGER> WFL COMPILERS READONLY,	00422000		11
12	423000	QUEUE = CLASS,	00423000		12
13	424000	COMPILETYPE 35 <INTEGER> MNEMONIC COMPILEANDGO/SYNTAX/LIBRARYANDGO/	00424000	29.017.003	13
14	425000	LIBRARY COMPILERS,	00425000	29.035.013	14
15	426000	TASK 37 <TASK> WFL,	00426000		15
16	427000	ORGUNIT 38 <INTEGER> COMPILERS READONLY,	00427000		16
17	428000	MAXCARDS 39 <INTEGER> WFL COMPILERS,	00428000	29.012.008	17
18	429000	PUNCHLIMIT = MAXCARDS,	00429000	29.012.008	18
19	430000	MAXLINES 40 <INTEGER> WFL COMPILERS,	00430000	29.012.008	19
20	431000	PRINTLIMIT = MAXLINES,	00431000	29.012.008	20
21	432000	JOBNUMBER 41 <INTEGER> COMPILERS,	00432000		21
22	433000	CHARGE 42 <SIMPLENAME> WFL COMPILERS,	00433000	29.035.013	22
23	434000	CHARGECODE = CHARGE,	00434000		23
24	435000	DESTNAME 44 <DESTINATION> WFL COMPILERS,	00435000		24
25	436000	BACKUPDESTINATION = DESTNAME,	00436000		25
26	437000	SOURCESTATION 45 <REAL> COMPILERS,	00437000		26
27	438000	DESTSTATION 46 <REAL> COMPILERS,	00438000		27
28	439000	SOURCEKIND 47 <REAL> COMPILERS READONLY,	00439000		28
29	440000	RESTARTED 48 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> COMPILERS,	00440000	31.056.007	29
30	441000	MAXWAIT 49 <REAL> WFL COMPILERS,	00441000	29.017.004	30
31	442000	STACKLIMIT 50 <INTEGER> WFL COMPILERS,	00442000	29.035.013	31
32	443000	FETCH 52 <FETCH> WFL,	00443000		32
33	444000	RESOURCE 53 <RESOURCE> WFL,	00444000		33
34	445000	FAMILY 55 <FAMILY> WFL COMPILERS,	00445000	29.017.004	34
35	446000	WAITLIMIT 56 <REAL> WFL COMPILERS,	00446000		35
36	447000	ELAPSEDLIMIT 57 <REAL> WFL COMPILERS,	00447000		36
37	448000	TASKLIMIT 58 <REAL> WFL COMPILERS, %QUEUE ATTRIBUTE ONLY	00448000	29.059.020	37
38	449000	DISKLIMIT 59 <INTEGER> WFL COMPILERS,	00449000	29.059.020	38
39	449100	TANKING 60 <INTEGER> MNEMONIC UNSPECIFIED/NONE/SYNC/ASYNC	00449100	29.038.014	39
40	449120	COMPILERS WFL,	00449120	31.041.003	40
41	449130	ACCESSCODE 61 <SIMPLENAME> WFL COMPILERS,	00449130	31.215.029	41
42	449140	SUBSYSTEM 62 <IDENTIFIER> WFL COMPILERS,	00449140	31.215.029	42
43	449150	BACKUPFAMILY 63 <IDENTIFIER> WFL COMPILERS,	00449150	31.158.016	43
44	449525	HOSTNAME 64 <IDENTIFIER> WFL COMPILERS,	00449525	31.215.029	44
45	449600	HISTORYTYPE 66 <INTEGER> MNEMONIC	00449600		45
46	449650	NORMALV/	00449650	31.218.031	46
47	449655	DUMPINGV/	00449655	31.218.031	47
48	449660	QTEDV/	00449660	31.218.031	48
49	449665	STEDV/	00449665	31.218.031	49
50	449670	DSEDV/	00449670	31.218.031	50
51	449680	NORMALEOTV/	00449680	31.218.031	51
52	449685	SYNTAXERRORV/	00449685	31.218.031	52
53	449690	UNKNOWNEOTV/	00449690	31.218.031	53
54	449695	UNINITIATEDV	00449695	31.218.031	54
55	449700	WFL COMPILERS READONLY,	00449700	31.218.031	55
56	449750	HISTORYCAUSE 67 <INTEGER> MNEMONIC	00449750		56
57	449800	OPERATORCAUSEV/PROGRAMCAUSEV/RESOURCECAUSEV/	00449800		57
58	449820	FAULTCAUSEV/SYSTEMCAUSEV/	00449820		58
59	449850	DCERRCAUSEV/IOERRCAUSEV/	00449850		59
60	449880	SOFTIOERRCAUSEV/NEWIOERRCAUSEV/	00449880		60

1	449900	UNIMPLEMENTEDCAUSEV/UNSPECIFIEDCAUSEV/	00449900	1
2	449920	EBDMSERRCAUSEV/NETWORKCAUSEV	00449920	2
3	449950	WFL COMPILERS READONLY,	00449950	3
4	450010	HISTORYREASON 68 <INTEGER> MNEMONIC	00450010	4
5	450020	RSVPV/	00450020	5
6	450030	CLEARUNITV/	00450030	6
7	450040	JUSTDSEDV/	00450040	7
8	450050		00450050	8
9	450060	MISSINGCODEFILENAMEV/	00450060	9
10	450070	MISSINGCODEFILEV/	00450070	10
11	450080	INITACTIVETASKV/	00450080	11
12	450090	NOEXTERNALRUNV/	00450090	12
13	450100	INVALIDACCESSCODEV/	00450100	13
14	450110	DEATHINFAMILYV/	00450110	14
15	450120	CRITICALBLOCKV/	00450120	15
16	450130	BADGOTOV/	00450130	16
17	450140	NOTEXECUTABLEV/	00450140	17
18	450150	UNMATCHEDPARAMSV/	00450150	18
19	450160	INVCOMPILERVV/	00450160	19
20	450170	SECURITYERRORV/	00450170	20
21	450180	LIBMAINTV/	00450180	21
22	450190	BADRESIZEDEALLOCV/	00450190	22
23	450200	MISSINGINTRINSICV/	00450200	23
24	450210	INCOMPATIBLELEVELV/	00450210	24
25	450220	INFANTICIDEV/	00450220	25
26	450230	NOTBOUNDV/	00450230	26
27	450240	ILLEGALOWNARRAYV/	00450240	27
28	450250	DIMSIZEERRORV/	00450250	28
29	450260	UPLEVELATTACHV/	00450260	29
30	450270	ILLEGALSWAPV/	00450270	30
31	450280	SWAPDOESNTALLOWV/	00450280	31
32	450290	BADTASKATTRIBUTEV/	00450290	32
33	450300	MISSINGCARDDECKV/	00450300	33
34	450310	BADRESTARTV/	00450310	34
35	450320	BADEVENTUSAGEV/	00450320	35
36	450330	BADGIVELOCKV/	00450330	36
37	450340	BADGETLOCKV/	00450340	37
38	450350	COMPILERONLYV/	00450350	38
39	450360	TASKLIMITEXCEEDEDV/	00450360	39
40	450370	AXBADARRAYV/	00450370	40
41	450380	RUNTIMEWFLV/	00450380	41
42	450390	COMPILERERRORV/	00450390	42
43	450400	LIBMISSINGNAMEV/	00450400	43
44	450410	LIBTYPEMISMATCHV/	00450410	44
45	450420	LIBDIRECTORYNOTFOUNDV/	00450420	45
46	450430	CYCLICPROVISIONV/	00450430	46
47	450440	PREVIOUSLYFROZENLIBV/	00450440	47
48	450450	LIBIMPLEMENTATIONERRORV/	00450450	48
49	450460	BADPTRLIBV/	00450460	49
50	450470	NONUNQLIBV/	00450470	50
51	450480	NOINITPARAMV/	00450480	51
52	450490	SWAPJOBcantBELIBV/	00450490	52
53	450500	GLOBALLIBONLYV/	00450500	53
54	450510	INVALIDLIBREFV/	00450510	54
55	450520	LIBFEATURENOTIMPLEMENTEDV/	00450520	55
56	450530	BADCOMPILERINDEXV/	00450530	56
57	450540	LIBNOTPROCESSEDORRUNV/	00450540	57
58	450550	LIBMUSTBESEPARATESTACKV/	00450550	58
59	450560		00450560	59
60	450570	PROCESSEXCEEDEDV/	00450570	60

1	450580	IOEXCEEDV/	00450580		1
2	450590	STACKEXCEEDV/	00450590		2
3	450600	PRINTXCEEDV/	00450600		3
4	450610	PUNCHXCEEDV/	00450610		4
5	450620	CARDREADXCEEDV/	00450620	33.008.003	5
6	450630	MEMORYXCEEDV/	00450630		6
7	450640	DIRECTORYXCEEDV/	00450640		7
8	450650	TAPEXCEEDV/	00450650		8
9	450660	WAITXCEEDV/	00450660		9
10	450670	ELAPSEDEXCEEDV/	00450670		10
11	450680	DISKLIMITXCEEDV/	00450680		11
12	450690	STRINGPOOLEXCEEDV/	00450690		12
13	450700		00450700		13
14	450710	DIVIDEBYZEROV/	00450710		14
15	450720	EXPOVERFLOWV/	00450720		15
16	450730	EXPUNDERFLOWV/	00450730		16
17	450740	INVALIDINDEXV/	00450740		17
18	450750	INTEGEROVERFLOWV/	00450750		18
19	450760	INACTIVEQV/	00450760		19
20	450770	MEMORYPROTECTV/	00450770		20
21	450780	INVALIDOPV/	00450780		21
22	450790	LOOPV/	00450790		22
23	450800	MEMORYPARITYV/	00450800		23
24	450810	SCANPARITYV/	00450810		24
25	450820	INVALIDADDRESSV/	00450820		25
26	450830	STACKOVERFLOWV/	00450830		26
27	450840	STRINGPROTECTV/	00450840		27
28	450850	PROGRAMMEDOPV/	00450850		28
29	450860	BOTTOMOFSTACKV/	00450860		29
30	450870	SEQUENCEERRORV/	00450870		30
31	450880	INVALIDPCWV/	00450880		31
32	450890	STACKUNDERFLOWV/	00450890		32
33	450900	ZAPPEDV/	00450900		33
34	450910	DISKPARITYV/	00450910		34
35	450936		00450936		35
36	450938	NOMEMV/	00450938		36
37	450940	PARITYONPBITV/	00450940		37
38	450942	ARRAYTOOLARGEV/	00450942		38
39	450944		00450944		39
40	450946	WSSIZEEXCEEDV/	00450946		40
41	450948		00450948		41
42	450950	RESPONSEREQUIREDV/	00450950		42
43	450952		00450952		43
44	450954	DYNAMICOWNARRAYV/	00450954		44
45	450956		00450956		45
46	450958	NOHOSTV/	00450958		46
47	450960	HOSTNOTREACHABLEV/	00450960		47
48	450962	HOSTHALTLOADEDV/	00450962	35.181.029	48
49	450964	TASKPROTOCOLERRORV/	00450964	35.181.029	49
50	450966		00450966	35.181.029	50
51	450968	DISCONNECTEDV/	00450968	35.181.029	51
52	450970	SUSPENDEDV	00450970	35.181.029	52
53	450972	WFL COMPILERS READONLY,	00450972	35.181.029	53
54	450980	HSINFO 69 <STRING> WFL COMPILERS READONLY,	00450980	31.186.025	54
55	450982	HSPARAMSIZE 70 <INTEGER> WFL COMPILERS READONLY, % TEMP FOR 31&32&33	00450982	33.105.015	55
56	450984	HSPARAMARRAY 71 <STRING> WFL COMPILERS READONLY, % TEMP FOR 31&32&33	00450984	33.105.015	56
57	450985	ITINERARY 72 <STRING> COMPILERS READONLY,	00450985	32.048.003	57
58	450986	DATABASE 73 <ATTRIBUTELIST> WFL,	00450986	32.046.005	58
59	450987	LIBRARY 74 <ATTRIBUTELIST> WFL,	00450987	32.049.006	59
60	450988	VISIBILITY 75 <INTEGER> MNEMONIC UNSPECIFIED/	00450988	32.049.006	60



1	450989	MINIMAL/	00450989	32.049.006	1
2	450990	SUBSYSTEM/	00450990	32.049.006	2
3	450991	GLOBAL WFL COMPILERS	00450991	32.123.013	3
4	450995	,TASKPARAMETERS 76 <STRING> COMPILERS READONLY %PRELIMINARY FOR 32 IMP	00450995	32.123.013	4
5	451050	,STACKNUMBER 77 <INTEGER> COMPILERS READONLY	00451050	33.002.001	5
6	451100	,TIMESTARTED 78 <REAL> COMPILERS READONLY	00451100	33.002.001	6
7	451102	,STARTTIME 79 <STRING> WFL READONLY	00451102	33.043.008	7
8	451104	,STARTTIMESTAMP 80 <REAL> WFL READONLY	00451104	33.043.008	8
9	451106	,JOBSUMMARY 81 <INTEGER> MNEMONIC DEFAULT/	00451106	33.043.009	9
10	451108	CONDITIONAL/	00451108	33.043.009	10
11	451110	SUPPRESSED/	00451110	33.043.009	11
12	451112	UNCONDITIONAL WFL COMPILERS	00451112	33.043.009	12
13	451115	,CHECKPOINTABLE 82 <BOOLEAN> MNEMONIC TRUE/FALSE WFL COMPILERS READONLY	00451115	33.045.011	13
14	451120	,BRCLASS 83 <INTEGER> MNEMONIC NOBR/ONCEONLY/MULTIPLE WFL COMPILERS	00451120	33.045.011	14
15	451122	,SW1 84 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451122	33.104.014	15
16	451124	,SW2 85 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451124	33.104.014	16
17	451126	,SW3 86 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451126	33.104.014	17
18	451128	,SW4 87 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451128	33.104.014	18
19	451130	,SW5 88 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451130	33.104.014	19
20	451132	,SW6 89 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451132	33.104.014	20
21	451134	,SW7 90 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451134	33.104.014	21
22	451136	,SW8 91 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451136	33.104.014	22
23	451140	,INHERITMCSSTATUS 92 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL	00451140	33.105.015	23
24	451145	COMPILERS	00451145	33.105.015	24
25	451148	,ACTUALDESTNAME 93 <STRING> WFL COMPILERS	00451148	35.190.031	25
26	451150	,TADS 94 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE> WFL COMPILERS	00451150	35.190.031	26
27	451152	,LANGUAGE 95 <SIMPLENAME> WFL COMPILERS	00451152	35.190.031	27
28	451154	,CODEVISIBILITY 96 <INTEGER> MNEMONIC UNSPECIFIED/	00451154	35.190.031	28
29	451156	MINIMAL/	00451156	35.190.031	29
30	451158	SUBSYSTEM/	00451158	35.190.031	30
31	451160	GLOBAL WFL COMPILERS	00451160	35.190.031	31
32	451162	,JOBSUMMARYTITLE 97 <NAME> WFL COMPILERS	00451162	35.190.031	32
33	451164	,NOJOBSUMMARYIO 98 <BOOLEAN> MNEMONIC TRUE/FALSE/<NUL TRUE>	00451164	35.190.031	33
34	451166	WFL COMPILERS	00451166	35.190.031	34
35	451168	,PRINTDEFAULTS 99 <STRING> WFL COMPILERS	00451168	35.190.031	35
36	451170	,ACCEPTEVENT 100 <EVENT> COMPILERS	00451170	35.190.031	36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	PROGRAM INFORMATION BLOCK	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	APPENDIX E	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		WORKFILE [NOT NAMED]	5:05 PM TUESDAY, APRIL 9, 1985	1
2				2
3				3
4	15371200	%	15371200 31.062.240	4
5	15371220	% BOTH PROCESS TASKS AND PROGRAM SEGMENT DICTIONARIES ARE IMPLEMENTED	15371220 31.170.674	5
6	15371240	% AS STACKS - ARRAYS IN THE STACKVECTOR. THE FIXED BASE OF THE STACK	15371240 31.170.674	6
7	15371260	% CONTAINS SOME CONTROL INFORMATION REQUIRED BY HARDWARE OR SOFTWARE	15371260 31.170.674	7
8	15371280	% CONVENTION, AND SOME DATA THAT ARE NEEDED ONLY WHEN THE STACK IS	15371280 31.170.674	8
9	15371300	% PRESENT AND AVAILABLE. THE REST OF THE CONTROL INFORMATION IS	15371300 31.170.674	9
10	15371320	% KEPT IN SOME ARRAYS INDEXED IN PARALLEL WITH THE STACKVECTOR:	15371320 34.005.122	10
11	15371340	% STACKINFO IS A SCALAR ARRAY CONTAINING DATA THAT	15371340 34.009.256	11
12	15371360	% CHARACTERIZE EACH STACK NUMBER AS TASK, SEGDICT, ETC.	15371360 34.005.122	12
13	15371370	% STACKINFO FIELDS ARE ACCESSIBLE AT ALL TIMES FOR EVERY	15371370 34.009.256	13
14	15371380	% STACK INDEX IN THE ALLOWABLE RANGE, AND IS ARRANGED FOR	15371380 34.009.256	14
15	15371390	% MASKSEARCHING.	15371390 34.005.122	15
16	15371420	% STACKSTATUS IS A SCALAR ARRAY CONTAINING PRIORITY AND PROCESSING	15371420 31.170.674	16
17	15371440	% STATE.	15371440 34.005.142	17
18	15371480	% SPIBVECTOR IS A DOPE VECTOR TO STRUCTURES CONTAINING ALL THE REST	15371480 31.170.674	18
19	15371500	% OF THE CONTROL INFORMATION FOR A TASK OR SEGDICT. EACH SUCH STRUCTURE	15371500 31.170.674	19
20	15371520	% IS CALLED A "PROGRAM INFORMATION BLOCK" (PIB) AND IS BASICALLY A	15371520 31.170.674	20
21	15371540	% CONTAINER FOR ALL SORTS OF DATA REQUIRED BY MANY DIFFERENT MODULES.	15371540 31.170.674	21
22	15371560	%	15371560 31.170.674	22
23	15371580	% THE "TASK" VARIABLE DECLARED IN USER LANGUAGES IS IMPLEMENTED AS A	15371580 31.170.674	23
24	15371600	% PIB. ALL PIBS BELONG (IN THE MEMORY-MANAGEMENT SENSE) TO THE MCP	15371600 31.170.674	24
25	15371620	% AND RESIDE IN GLOBAL MEMORY, NO MATTER WHERE DECLARED. THOSE PIBS	15371620 31.170.674	25
26	15371640	% WHICH ARE ACTIVE HAVE DESCRIPTORS IN THE SYSTEM PIB VECTOR AND ARE	15371640 31.170.674	26
27	15371660	% ADDRESSABLE AS SPIB[<STACK NUMBER>,*]. (TASKS BEING RUN VIA HOST	15371660 31.170.674	27
28	15371680	% SERVICES AT AN EXTERNAL HOST HAVE PIBS WITH DESCRIPTORS IN THE	15371680 31.170.674	28
29	15371700	% EXTSPIBVECTOR.)	15371700 31.170.674	29
30	15371720	%	15371720 31.170.674	30
31	15371740	CONSTANT % PIB STRUCTURE: COMPLETE PIBS ARE USED FOR PROCESS STACKS	15371740 31.170.674	31
32	15371760	% AND FOR DBS AND THE MCP STACK. SHORT PIBS ARE USED	15371760 31.170.674	32
33	15371800	% FOR ORDINARY SEGDICTS AND INTRINSICS.	15371800 31.062.240	33
34	15372000	% ORDERED ALPHABETICALLY EXCEPT WHEN OTHERWISE CONSTRAINED.	15372000 31.062.240	34
35	15372200	%%%%%%%%% ONLY IN PROCESS PIBS:	15372200 31.170.674	35
36	15372400	PIBMSCW =0, % D1 MSCW FOR SOPHIA ROUTINES	15372400 31.170.674	36
37	15373200	%%%%%%%%% ONLY IN SEGDICT PIBS:	15373200 31.170.674	37
38	15373400	CODEHEADERINDEX =0, % INDEX OF CODE-FILE HEADER	15373400 31.166.240	38
39	15373800	%%%%%%%%% IN ALL PIBS:	15373800 31.170.674	39
40	15374000	ZOTBOT1, % CLEARED BY DOCTOR BEFORE BOT	15374000 31.162.599	40
41	15374100	ACTUALBOXES =ZOTBOT1, % BOXMASK FOR LOGGING - ACTUALLY USED BOXES	15374100 34.004.078	41
42	15374200	CODELINKS, % SEGDICT LINKAGES: CF SEGDICTF, ETC	15374200 312261018	42
43	15374400	COMPILERINFO, % FROM SEGO: CF LANGUAGEYPEF, ETC	15374400 31.162.599	43
44	15374600	COREINTEGRAL, % WORKING-SET ACCOUNTING	15374600 31.162.599	44
45	15374800	COREINUSE, % TOTAL WORDS OF CORE IN USE	15374800 31.162.599	45
46	15375000	COREINUSESAVED, % WORDS OF CORE IN USE THAT ARE MARKED SAVE	15375000 31.162.599	46
47	15375200	FORCEDOLAYTIME, % OVERLAY ACCOUNTING	15375200 31.162.599	47
48	15375300	GRAPHHEADWORD, % HEADWORD FOR GRAPH ROUTINES	15375300 32.034.203	48
49	15375400	OLAYCNTL, % OLAYFREQ=47:24 OLAYAMT=23:24	15375400 31.162.599	49
50	15375450	ORIGBOXES, % BOXMASK FOR LOGGING - ORIGINAL BOXES	15375450 34.004.078	50
51	15375500	SEARCHINFO, % INFO NEEDED TO SEARCH A STACK	15375500 32.056.510	51
52	15375600	SUBSPACEID, % NAME OF OWNER OF SUBSPACE	15375600 31.162.599	52
53	15375800	SWAPHOLD, % SWAPJOB COUNT OF I/O OR SONS ACTIVE	15375800 31.162.599	53
54	15376000	TIMESTARTED, % -CLOCK AT INITIATION; ELAPSED TIME AT EOT	15376000 31.162.599	54
55	15376200	ZOTBOT1L =TIMESTARTED,	15376200 31.166.240	55
56	15376500	RUNNINGCOUNT, % USERS OF SEGDICT/DBS (& SIBS FOR DM)	15376500 32.033.179	56
57	15376600	D1PIBLENGTH, %===== END OF SEGDICT PIB =====	15376600 31.170.674	57
58	15376800	%%%%%%%%% REST OF PROCESS (& DBS & MCP) PIBS:	15376800 31.170.674	58
59	15377000	%%%%%%%%% SELF-VALIDATING ARRAY ITEMS:	15377000 31.062.240	59
60	15377200	FIRSTINITARRAY =D1PIBLENGTH, % TRANSCRIBED FOR RUN	15377200 31.170.674	60

1	15377400	ACCESSCODE =FIRSTINITARRAY,	% ARRAY: STANDARDFORM NAME	15377400	31.166.240	1
2	15377600	CHARGECODE,	% ARRAY: STANDARDFORM NAME	15377600	31.162.599	2
3	15377650	DBEQTNBLK,	% DATABASE LABEL EQUATION BLOCK	15377650	32.052.429	3
4	15377700	DBSINFO,	% TOP OF FROZEN DBS, ETC. (SEE LAYOUT)	15377700	32.033.179	4
5	15377750	DESTNAME,	% STD. FORM STATION NAME OF DESTINATION	15377750	35.027.777	5
6	15377800	HOSTNAME,	% STD. FORM HOST NAME	15377800	31.162.599	6
7	15378200	FAMILYSUB,	% INTERNALFORM FAMILY SPECIFICATIONS	15378200	31.162.599	7
8	15378500	JSTITLE,	% DISPLAYFORM JOBSUMMARYTITLE	15378500	351122543	8
9	15378600	MYBDNAME,	% STANDARDFORM PREFIX FOR BD FILES	15378600	31.162.599	9
10	15378800	MYFPB,	% FILE PARAMETER BLOCK	15378800	31.162.599	10
11	15379000	MYNAME,	% STANDARDFORM TASK NAME	15379000	35.011.319	11
12	15379400	MYPPB,	% PROGRAM PARAMETER BLOCK	15379400	31.162.599	12
13	15379600	SUBSYSTEMID,	% SUBSYSTEM NAME (SUBSTANDARDFORM NAME)	15379600	31.162.599	13
14	15379800	YOURNAME,	% STANDARDFORM COMPILEE NAME	15379800	31.162.599	14
15	15380000	BACKUPFAMILY,	% PACKNAME IN SUBSTANDARDFORM	15380000	31.162.599	15
16	15380200		LASTINITARRAY =BACKUPFAMILY,	15380200	31.166.240	16
17	15380400		%%%%%%%%%% ITEMS NOT REPORTABLE AFTER EOT:	15380400	31.062.240	17
18	15380600	ZOTEOT,	% CLEARED BY TERMINATE AFTER EOT	15380600	31.162.599	18
19	15381000	BDINFO =ZOTEOT,	% BACKUP OUTPUT CONTROL: CF BNUMF, ETC	15381000	35.023.641	19
20	15381200	EVENTCAR,	% DESCRIPTOR TO SIDECAR ARRAY FOR EVENTS	15381200	35.008.265	20
21	15381300	BEDWORD,	% INFO FOR EVENT LINKAGE (CF MULTI-WAIT)	15381300	35.008.265	21
22	15381400	BLOCK,	% SIRW TO (SCW OF) CRITICAL BLOCK	15381400	31.162.599	22
23	15381600	CARDSPUNCHED,	% DECREMENTED PUNCH-CARD OUTPUT LIMIT	15381600	31.162.599	23
24	15381640	ALARMTIMES,	% MEASURES FREQUENCY OF ALARM INTERRUPTS	15381640	35.032.873	24
25	15381660		% (TWO 24 BIT FIELDS, TIMES IN SECONDS)	15381660	35.032.873	25
26	15381700	BOTTIMESTAMP,	% TIMESTAMP AT BOT	15381700	34.018.467	26
27	15382000	CPINFO,	% CHECKPOINT DATA: CF CPCOUNTF, ETC	15382000	31.162.599	27
28	15382200	DISKUSED,	% DECREMENTED USER-DISK LIMIT	15382200	31.162.599	28
29	15382600	ENTRYPOINT,	% PCW OR SIRW FOR TASK ENTRY POINT	15382600	31.162.599	29
30	15382800	EXCEPTIONEVENT,	% EVENT CAUSED BY IPC STATUS CHANGE	15382800	31.162.599	30
31	15383000	EXCEPTIONEVENT2,	% (SECOND WORD)	15383000	31.162.599	31
32	15383200	EXTERNALFAMILYLINK,	% EXTERNAL FAMILY LINKAGE VIA TAGS	15383200	31.163.603	32
33	15383400	FREEPBITS,	% WORDS OVERLAYED WHILE BATCH JOB SUSPENDED	15383400	31.162.599	33
34	15383450	HEADERACCESS,	% JOB AND CODE FILE HEADER INDICIES	15383450	35.013.388	34
35	15383500	INSTRETRYINFO,	% B7700 INSTRUCTION RETRY INFORMATION	15383500	31.7X.026	35
36	15383600	INTERCEPTRCW,	% RCW DISPLACED BY INTERLOOPER'S	15383600	31.162.599	36
37	15383800	INTERCESSION,	% INTERCEDE DRIVER	15383800	31.162.599	37
38	15383900	ITINERARY,	% HOSTSERVICES ITINERARY INFORMATION	15383900	32.044.304	38
39	15383920	JOBDECKINFO,	% JOB DECK CONTROL ARRAY	15383920	32.054.458	39
40	15383940	JOBINFO,	% ABOUT THE JOB OF THIS TASK: CF JOBCLASSF	15383940	32.054.458	40
41	15384000	JOBMSGS,	% 2-D ARRAY: DISPLAY RSVP ACCEPT ETC.	15384000	31.162.599	41
42	15384200	PIBMEMLOC =JOBMSGS,	% TEMP MEMLOC WORD (SWAPPER TO INITIATE)	15384200	31.170.674	42
43	15384600	LIBRARYDIR,	% LIBRARY DIRECTORY DESCRIPTOR	15384600	31.163.577	43
44	15384700	LIBINITIATOR = LIBRARYDIR,	% LIB INITIATION INFO (LIBACCESS, ETC.)	15384700	32.147.947	44
45	15384800	LINESPRINTED,	% DECREMENTED PRINTER OUTPUT LIMIT	15384800	31.162.599	45
46	15385000	OLAYFILEDESC,	% HEADER FOR OVERLAY FILE. SHARED WITH:	15385000	31.162.599	46
47	15385200	D1STACKINFO =OLAYFILEDESC,	% CODE-FILE DESC FOR D1STACK (BEFORE BOJ)	15385200	31.166.240	47
48	15385550	PORTLIB PIB,	% NO OF PORTS DECLARED AND TOUCHED, ETC.	15385550	35.005.118	48
49	15385600	PROCESSFAMILYLINK,	% CF PARENTF, ETC	15385600	31.162.599	49
50	15386400	REMPRINTMASK,	% WHICH MCS'S TO NOTIFY OF PRINTING AT EOJ	15386400	31.162.599	50
51	15386600	REMPUNCHMASK,	% WHICH MCS'S TO NOTIFY OF PUNCHING AT EOJ	15386600	31.162.599	51
52	15386800	REPLY,	% RSVP COMMUNICATION: CF REPVALF, ETC	15386800	31.162.599	52
53	15387000	REPLYEVENTX,	% RSVP EVENT	15387000	31.222.003	53
54	15387200	REPLYEVENT2,	% (SECOND WORD)	15387200	31.222.003	54
55	15387600	ROLLOUTINFO,	% WFL-SUPPLIED JOBSTACK ROLLOUT DATA	15387600	31.162.599	55
56	15388000	SOFTINTQ,	% SOFTWARE INTERRUPT QUEUE HEADWORD	15388000	31.162.599	56
57	15388400	SWAPDATA,	% SIDE CAR FOR SWAPPER STUFF	15388400	31.162.599	57
58	15388500	TASKBOXES,	% BOXES TASK IS ALLOWED IN	15388500	34.004.078	58
59	15388600	TASKINFO,	% MISCELLANEOUS: CF INUSEBYSYSTEMF, ETC	15388600	31.162.599	59
60	15388650	TASKPARAMS,	% BEFORE INITIATE: ARRAY OF PARAMETERS	15388650	32.033.179	60



1	15388700		% AFTER INITIATE: NUMBER OF PARAMETERS	15388700	32.033.179	1
2	15388800	PIBIOCB,	% ARRAY: IOCB FOR DISKWAIT, SWAPPER ETC.	15388800	31.170.674	2
3	15389000	TASKTYPE,	% SEE PROCESSTYPE DEFINE ET SEQ.	15389000	31.162.599	3
4	15389700	HSINFO,	% HOSTSERVICES TAG INFORMATION	15389700	31.185.744	4
5	15389800	SURROGATEINFO,	% FOR CONTROLLING INTER-HOST TASKING	15389800	31.162.599	5
6	15389830	WAITSTART,	% CLOCK WHEN AN EVENT-WAIT STARTED, IF > 0.	15389830	35.023.641	6
7	15389840		% ALSO USED (<0) TO LINK STACKS TO SWAP IN	15389840	35.023.641	7
8	15389842	DEIMPRUNTIMESET,	% THE BITS CORRESPONDING TO DEIMPLEMATION	15389842	350772055	8
9	15389844		% WARNIG'S BITS ARE SET HERE TO REPORT THE	15389844	350772055	9
10	15389846		% DISPLAYED WARNINGS FOR A SOFTWARE.	15389846	350772055	10
11	15389850		%%%%%%%%% BELOW ARE ITEMS USED BY GEORGE, ORDERED &	15389850	35.023.641	11
12	15389860		%%%%%%%%% GROUPED TOGETHER FOR B7000 PERFORMANCE BOOST	15389860	35.023.641	12
13	15389870	ACTIONQ,	% THINGS THIS STACK MUST DO: CF ACTIONCASEF	15389870	35.023.641	13
14	15389880	RECENTPROC,	% PROC TIME AT LAST RESURRECT	15389880	35.023.641	14
15	15389890	PALACE,	% PCW TO GEORGE OF SOPHIA	15389890	35.023.641	15
16	15389900	PALACEREF,	% PSEUDOSTACK SIRW TO PALACE IN THIS PIB	15389900	35.023.641	16
17	15389910	USAGE,	% MISCELLANEOUS: SEE ARP, ETC	15389910	35.023.641	17
18	15389920	DMSBED,	% DM EVENT LINKAGE	15389920	35.023.641	18
19	15389930	READYON,	% CLOCK WHEN ENTERED INTO READYQ OR GEORGED	15389930	35.023.641	19
20	15389940	SCHEDDATA =READYON,	% SCHEDULING INFORMATION	15389940	35.023.641	20
21	15389950	PROCESSOR,	% PROCESSOR CURRENTLY EXECUTING THIS STACK	15389950	35.023.641	21
22	15389960	CLOCKONTIME,	% CLOCK WHEN STACK GOT PROCESSOR	15389960	35.023.641	22
23	15389970	LETGEORGEDOIT,	% SIGNAL FOR GEORGE (FULL WORD, NOT LOCKED)	15389970	35.023.641	23
24	15390000	ZOTEOTL =LETGEORGEDOIT,		15390000	35.023.641	24
25	15390200		%%%%%%%%% ITEMS REPORTABLE AFTER EOT:	15390200	31.062.240	25
26	15390400	ZOTBOT2,	% CLEARED BY DOCTOR BEFORE BOT	15390400	31.162.599	26
27	15390450	READYTIME = ZOTBOT2,	% TIME-I-WANTED-PROCESSOR ACCUMULATOR	15390450	35.023.641	27
28	15390500	PROCESSTIME,	% PROCESS TIME ACCUMULATOR (CLOCK UNITS)	15390500	35.023.641	28
29	15390540	\$ SET OMIT = NOT ACTIVETIME		15390540	340431074	29
30	15390560	% ACTCOUNT & ACTTIME ONLY TAKE UP PIB SPACE IF ACTIVETIME OPTION IS SET		15390560	340431074	30
31	15390580	ACTCOUNT,	% COUNTER TO CONTROL ACTTIME	15390580	35.023.641	31
32	15390600	ACTTIME,	% ACCUMULATES TIME DURING PROCESS OR I/O	15390600	340431074	32
33	15390620	\$ POP OMIT SET OMIT = ACTIVETIME		15390620	340431074	33
34	15390640	ACTCOUNT = 4'FFF',	% INVALID WITHOUT THE ACTIVETIME OPTION	15390640	340431074	34
35	15390660	ACTTIME = 4'FFF',	% INVALID WITHOUT THE ACTIVETIME OPTION	15390660	340431074	35
36	15390680	ACTTIMEFUDGE = PROCESSTIME,	% RESTORE CORRECT COUNT	15390680	35.023.641	36
37	15390700	\$ POP OMIT		15390700	340431074	37
38	15390800		%%%%%%%%% ABOVE ARE ITEMS USED BY GEORGE	15390800	35.023.641	38
39	15390850	IOTIME,	% I/O TIME ACCUMULATOR (CLOCK UNITS)	15390850	34.018.467	39
40	15390960	INITPBITCOUNT,	% COUNT OF INIT PBITS	15390960	34.018.467	40
41	15390970	INITPBITTYME,	% PROC TIME DOING INIT PBITS	15390970	34.018.467	41
42	15390980	OTHERPBITCOUNT,	% COUNT OF OTHER PBITS	15390980	34.018.467	42
43	15390990	OTHERPBITTYME,	% PROC TIME DOING PBITS EXCEPT INIT PBITS	15390990	34.018.467	43
44	15391000	HISTORY,	% CF HISTYPEF, ETC, & HISTORYTYPE ET SEQ.	15391000	31.162.599	44
45	15391200	IOCOUNT1,	% CARDS READ=47:24 CARDS PUNCHED=23:24	15391200	31.162.599	45
46	15391300	COMPTIME = IOCOUNT1,	% TEMP - TIME CODE COMPILED (DOCTOR TO BOJ)	15391300	33.001.013	46
47	15391400	IOCOUNT2,	% LINES PRINTED	15391400	31.162.599	47
48	15391500	SCHEDTIME =IOCOUNT2,	% HOLD SCHEDULE TIME BEFORE BOT	15391500	31.201.856	48
49	15391800	MYSTACKHISTORY,	% ARRAY: TEXT DISPLAY OF PROCEDURE HISTORY	15391800	31.162.599	49
50	15392400	STOPPOINT,	% RCW AND FAULT NUMBER OF LAST FAULT	15392400	31.162.599	50
51	15392600	ZOTBOT2L =STOPPOINT,		15392600	31.166.240	51
52	15392800		%%%%%%%%% ITEMS NEVER CLEARED:	15392800	31.062.240	52
53	15393000	LOCKEDEVENT,	% EVENT FOR "LOCKED" ATTRIBUTE	15393000	31.162.599	53
54	15393200	LOCKEDEVENT2,	% (SECOND WORD)	15393200	31.162.599	54
55	15393400	USERSTUFF,	% REST OF PIB TRANSCRIBED FOR RUN	15393400	31.170.674	55
56	15393600	VALIDITYBITS =USERSTUFF,	% VALIDITYBITS.VALBIT(X) MARKS PIB ITEM X	15393600	31.170.674	56
57	15393800		%%%%%%%%% ITEMS WITH VALIDITY BITS (LIMIT 48):	15393800	31.062.240	57
58	15394000	VALBIAS,	% SEE VALBIT DEFINE	15394000	31.162.599	58
59	15394200	AVALUE =VALBIAS,	% "TASKVALUE" ATTRIBUTE	15394200	31.166.240	59
60	15394300	MAXPROCESSTIME,	% LIMIT ON PROCESSOR TIME	15394300	35.023.641	60

1	15394350	SWAPSPEX,	% SWAPPER SPEX & CONTROLS: CF LOCATIONF ETC	15394350	35.023.641	1
2	15394400	CODECORE,	% D1 MEMORY SPEX: LIMITF & ESTIMATEF	15394400	31.162.599	2
3	15394600	DATACORE,	% D2 MEMORY SPEX: LIMITF & ESTIMATEF	15394600	31.162.599	3
4	15394800	DISKLIMIT,	% LIMIT ON USER DISK	15394800	31.162.599	4
5	15395000	ELAPSEDLIMIT,	% LIMIT ON ELAPSED TIME	15395000	34.018.467	5
6	15395020	MAXIOTIME,	% LIMIT ON I/O TIME	15395020	34.018.467	6
7	15395060	MAXWAIT,	% LIMIT ON DM LOCK WAIT	15395060	34.018.467	7
8	15395080	WAITLIMIT,	% LIMIT ON EACH EVENT WAIT	15395080	34.018.467	8
9	15395200	ERROR,	% -HOW IF ERROR ON TASK ATTRIBUTE # HOW	15395200	31.162.599	9
10	15395400	EXCEPTIONTASK,	% REF TO PIB TO KICK AT OUR STATUS CHANGE	15395400	31.170.674	10
11	15396200	OPTION,	% THE "OPTION" ATTRIBUTE: CF FAULTCLSF, ETC	15396200	31.162.599	11
12	15396400	PARTNER,	% REF TO PIB TO WAKE IF I CONTINUE OR DIE	15396400	31.170.674	12
13	15396600	PATHCONTROL,	% SOURCE ID OUTPUT ROUTE: CF DESTCONTROLF	15396600	31.162.599	13
14	15396800	PRINTLIMIT,	% LIMIT ON PRINTED LINES	15396800	31.162.599	14
15	15397000	PRIORITY,	% "DECLAREDPRIORITY" ATTRIBUTE	15397000	31.162.599	15
16	15397200	PUNCLIMIT,	% LIMIT ON PUNCH-CARD OUTPUT	15397200	31.162.599	16
17	15397400	RESTARTCOUNT,	% NUMBER OF TIMES RESTARTED	15397400	31.162.599	17
18	15397600	SERIAL,	% STK/JOB/PIB ID:SERIALSTKNRF,JOBNRF,TSKNRF	15397600	350741952	18
19	15397800	STACKLIMIT,	% LIMIT ON STACK SIZE	15397800	31.162.599	19
20	15398000	STACKSIZE,	% INITIAL STACK SIZE	15398000	31.162.599	20
21	15398200	STATIONINFO,	% "STATION"&"TANKING": STATIONF & TASKTANKF	15398200	31.162.599	21
22	15398600	TARGETTIME,	% "TARGETTIME" ATTRIBUTE (NOT YET IMPL.)	15398600	31.162.599	22
23	15398800	TAPECOUNT,	% RESOURCE ACCTG: CF PETAPEF, ETC	15398800	31.162.599	23
24	15399000	TAPEPOOL,	% DITTO	15399000	31.162.599	24
25	15399200	TASKLIMIT,	% DECREMENTED LIMIT ON TASK INITIATIONS	15399200	31.162.599	25
26	15399400	USERCODE,	% STANDARDFORM USERCODE (4 WORDS IN PIB)	15399400	31.170.674	26
27	15399600	USERCODEPRIV = USERCODE+3,	% 4TH WORD OF USERCODE PLUS:	15399600	31.166.240	27
28	15400000		% SEE USERCODEPRIVF DEFINE ETC.	15400000	32.055.474	28
29	15400400	PIBLENGTH,	%===== END FIXED PROCESS PIB =====	15400400	31.170.674	29
30	15400600	TASKNAME1 = PIBLENGTH,	% FIRST WORD OF MYNAME IF APPENDED TO PIB	15400600	31.170.674	30
31	15400800	PIBNAMESPACE = 8;	% DEFAULT LENGTH FOR MYNAME APPENDIX	15400800	31.170.674	31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

		WORKFILE [NOT NAMED]	5:05 PM TUESDAY, APRIL 9, 1985	
1				
2				
3				
4	15506400	% FIELDS IN STACKINFO	15506400	31.053.002
5	15506600	, BOXTHRUVISF = [47:10]# % PUN FOR GEORGE: BOX & 3 BITS & VIS	15506600	34.005.132
6	15506800	, BOXNOF = [47:6]# % BOX NUMBER IN WHICH STACK RESIDES	15506800	34.005.132
7	15507000	, URSVPWAITINGF = [41:1]# % WAITING ON A UNIT RSVP	15507000	34.005.132
8	15507200	, TIMELIMITSF = [40:1]# % ETERNALIR MUST MONITOR TIME LIMITS	15507200	34.005.132
9	15507400	, NEEDAROWF = [39:1]# % OLAYSCOUT SERVICES ARE NEEDED	15507400	34.005.132
10	15507600	, VISIBLEF = [38:1]# % DISPLAY AND LOG THIS TASK (NOTE PUN)	15507600	34.005.132
11	15507800	, CANTSUSPENDF = [37:1]# % MAY NOT BE SUSPENDED	15507800	34.005.132
12	15508000	, SUSPF = [36:1]# % SUSPENDED BY SYSTEM	15508000	34.005.132
13	15508600	, PRIVILEGEDUSERF = [35:1]# % THIS PROCESS IS PRIVILEGED	15508600	31.138.003
14	15508800	, COMPILERF = [34:1]# % THIS IS A COMPILER	15508800	31.138.003
15	15509000	, ACTIVEBSF = [33:1]# % DBS IS (STILL) ACTIVE	15509000	31.138.003
16	15509100	, SUSPREQUESTF = [32:1]# % ST BY SYS REQUESTED	15509100	33.011.314
17	15509200	, STACKKINDF = [31:8]# % NATURE OF STACK:	15509200	31.138.003
18	15509400	, MCPSTACKF = [31:1]# % THE MCP (D[O]) STACK 10010000	15509400	31.138.003
19	15509500	, FROZEN (D[O]) MCP STACK 10011000	15509500	32.048.380
20	15509600	, DBSF = [30:1]# % OLD STYLE DATABASE STACK 01010000	15509600	32.033.179
21	15509700	, NEW STYLE DATABASE STACK 01000001	15509700	32.033.179
22	15509800	, INTRINSICSF = [29:1]# % AN INTRINSICS STACK 00110000	15509800	31.138.003
23	15510000	, SEGDICTIONARY = [28:1]# % A SEGMENT DICTIONARY XXX1X000	15510000	31.138.003
24	15510200	, LIBRARYSTACKF = [27:1]# % A USABLE (FROZEN) LIBRARY 000X100X	15510200	31.183.733
25	15510400	, JOBSTACKF = [26:1]# % A JOB STACK 00000101	15510400	31.138.003
26	15510600	, IRINDICATORF = [25:1]# % AN INDEPENDENT RUNNER 0000X011	15510600	321741152
27	15510800	, TASKSTACKF = [24:1]# % A TASK (PROCESSING) STACK 0000XXX1	15510800	31.138.003
28	15510900	, INTERCEPTABLEF = [23:1]# % IR STACK CAN BE INTERCEPTED	15510900	340621601
29	15511000	, VIEWABLEF = [22:1]# % OK TO LOOK AT THE ASSOCIATED PIB	15511000	31.170.674
30	15511200	, LOGGABLEF = [21:1]# % OK TO WRITE TO JOBLOG	15511200	31.062.240
31	15511400	, STACKPRESENTF = [20:1]# % STACK HAS BEEN BUILT AND IS IN CORE	15511400	31.144.496
32	15511600	, SEARCHABLEF = [19:1]# % STACK (OR PARAMS) SUBJECT TO SEARCH	15511600	31.144.496
33	15511800	, SEARCHVIAQF = [18:1]# % SEARCH MUST BE DONE VIA SEARCHQ	15511800	31.144.496
34	15512000	, BLOCKEDF = [17:1]# % STACK IS BLOCKED - NO PROCS ALLOWED	15512000	32.056.510
35	15512200	, STATSTACKF = [16:1]# % STAT STACK NEEDED IF STATISTICS	15512200	31.201.870
36	15512300	, STRETCHINGF = [15:1]# % STACK IS BEING STRETCHED	15512300	321781141
37	15512350	, SIWSGOINGF = [14:1]# % WSSHERRIFF IS INVOLVED WITH THIS	15512350	34.012.256
38	15512400	, STACK'S MEMORY: EOJ MUST WAIT.	15512400	34.012.256
39	15512450	, SIEOJF = [13:1]# % EOJ IS DESTROYING THIS STACK, SO	15512450	34.012.256
40	15512500	, WSSHERRIFF MUST LAY OFF.	15512500	34.012.256
41	15512520	, DONTUPDATEF = [12:1]# % DONT UPDATE SEGO STACKSIZE	15512520	340581495
42	15512540	, NOTRESUMEABLEF = [11:1]# % THIS LIBRARY MUST NOT BE RESUMED AT	15512540	350571529
43	15512560	, THIS STAGE	15512560	350571529
44	15512570	, DONTKANGAROOF = [10:1]# % DO NOT KANGAROO ANY CODE THAT HAS	15512570	350761985
45	15512580	, THIS STACK AS A SEG DICT (DMS,HS)	15512580	350761985
46	15512590	, EOJINUSEONESF = [ 9:1]# % IF GOTOSOLVER FINDS THAT THE WORD	15512590	350762001
47	15512592	, VIA SIRW ISNT EOJRCW THEN	15512592	350762001
48	15512594	, IF THERE ISNT A GO-PAST SCW	15512594	350762001
49	15512596	, OR THIS BIT IS OFF THEN	15512596	350762001
50	15512598	, ITS TIME GET A MEMDUMP	15512598	350762001
51	15512600	, OTHERWISE	15512600	350762001
52	15512602	, TURN OFF THIS BIT AND USE THE	15512602	350762001
53	15512604	, GO-PAST	15512604	350762001
54	15512650	, LIBRESUMINGF = [ 8:1]# % LIB BEING RESUMED BUT NOT STRETCHED	15512650	351122539
55	15512700	, YET	15512700	351122539
56	15512780	, SWAPJOBFB = [ 0:1]# % THIS IS A SWAPPED TASK	15512780	350762001
57	15512800	, SWAPANDBOXF = [ 0:7]# % COMBINATION: = BOXNO IF NOT SWAPPED	15512800	34.005.132
58	15512960	% FIELDS IN PIB[SEARCHINFO]	15512960	32.056.510
59	15512980	, BLOCKINGNUMBERF = [ 7: 8]# % NUMBER OF OUTSTANDING BLOCKS	15512980	32.056.510
60	15512982	, NSTACKSEARCHEDGESF = [23:16]#	15512982	350711823

1	15512990	, WAITINGFORUNBLOCKF	= [47: 1]#		15512990	34.035.855	1
2	15513000	% FIELDS IN PIB[CODELINKS]		% CHANGED UNDER PROCESSCHANGELOCK	15513000	350531393	2
3	15513200	, TADSBPF	= [47:1]#	% A BP IS PENDING FOR THIS SESSION	15513200	350531393	3
4	15513250	, TADSPRIVATEF	= [46:1]#	% NON SHARABLE SEGDICTIONARY BECAUSE OF TADS	15513250	350531393	4
5	15513300	, BOJINTRINSICSF	= [45:1]#	% ORIGINAL BOJ MUST LINK INTRINSICS	15513300	32.060.553	5
6	15513400	, SEGDICTIONARYLINKF	= [35: SNSIZE]#	% LINK TO NEXT SEGDICTIONARY	15513400	35.026.736	6
7	15513500			% FOR SAME HEADER	15513500	35.026.736	7
8	15513600	% AVAILABLE	= [21:10]#	% FORMERLY - INTRINSICS STACK NUMBER	15513600	32.174.947	8
9	15513800	, SEGDICTIONARY	= [11: SNSIZE]#	% SEGMENT-DICTIONARY (D1) STACK	15513800	35.026.736	9
10	15513900			% NUMBER	15513900	35.026.736	10
11	15514000	% FIELDS IN PIB[COMPILERINFO] AND CODEFILE SEGO[SOCOMPILERINFO]			15514000	312241005	11
12	15515000	, IPCCAPABLEF	= [47:1]#		15515000	31.062.240	12
13	15515200	, SORTCAPABLEF	= [46:1]#		15515200	31.062.240	13
14	15515400	, CONTROLPROGRAMF	= [45:1]#	% SET BY CP COMMAND IN SETSTATUS	15515400	31.157.559	14
15	15515600	, DMSCAPABLEF	= [44:1]#		15515600	31.062.240	15
16	15515800	% NOBADPTRF	= [43:1]#	% NO "UPLEVEL" POINTER ASSIGNMENT	15515800	31.140.002	16
17	15516000	, PRIVILEGEDPROGRAMF	= [42:1]#	% SET BY PP COMMAND IN SETSTATUS	15516000	31.157.559	17
18	15516200	, LIBRARYCAPABLEF	= [41:1]#	% COULD "FREEZE" AS A LIBRARY	15516200	31.092.375	18
19	15516210	, NOGLOBALEQUATEF	= [40:1]#	% WFL: NO GLOBAL FILE EQUATION	15516210	32.054.458	19
20	15516215	, TRANSPARENTPRIVF	= [39:1]#	% TRANSPARENT PRIVILEGE	15516215	321741148	20
21	15516218	, AUTOSUPPRESSF	= [38:1]#	% SET BY SUPPRESS COMAND IN SETSTATUS	15516218	35.023.638	21
22	15516219	, RESIDENTPROGRAMF	= [37:1]#	% SET BY RP COMMAND IN SETSTATUS	15516219	350711854	22
23	15516220	, LANGUAGEPEF	= [31:8]#		15516220	312241005	23
24	15516240	% IN COMPILERINFO AND IN SOCOMPILERINFO OF NEW CODE FILES:			15516240	312241005	24
25	15516260	, MARKLEVELF	= [23:8]#	% MARK*10+LEVEL LEQ 99; GEQ 10 (31)	15516260	312241005	25
26	15516270	% U N U S E D	[ 15:6]		15516270	312241005	26
27	15516280	, CYCLENUMBERF	= [9:10]#	% CYCLE LEQ 999	15516280	312241005	27
28	15516300	% IN SOCOMPILERINFO OF OLD CODE FILES:			15516300	312241005	28
29	15516320	, OLDMARKF	= [23:8]#	% MARK LEQ 9 (3)	15516320	312241005	29
30	15516340	, OLDLEVELF	= [15:8]#	% LEVEL LEQ 9	15516340	312241005	30
31	15516360	% OLDCYCLEF	= [ 7:8]#	% CYCLE LEQ 255	15516360	312241005	31
32	15516400	% FIELDS IN PIB[CODECORE] AND PIB[DATACORE]			15516400	31.170.674	32
33	15516600	, LIMITF	= [39:20]#		15516600	31.062.240	33
34	15516800	, ESTIMATEF	= [19:20]#		15516800	31.062.240	34
35	15517000	% FIELDS IN PIB[CPINFO]			15517000	31.170.674	35
36	15517200	, CPCOUNTF	= [ 9:10]#		15517200	31.062.240	36
37	15517320	% BR ATTRIBUTES FIELDS:			15517320	35.027.799	37
38	15517340	, CPDISABLEF	= [10:1]#	% SET => NOT CHECKPOINTABLE	15517340	33.045.739	38
39	15517360	, CPDSEDF	= [11:1]#	% SET = DO NOT RUN EPILOG PROCEDURES	15517360	33.045.739	39
40	15517380	, BRCLASSF	= [14:3]#	% BRCLASS ATTRIBUTE + 1	15517380	35.027.799	40
41	15517400	, CPSTATUSF	= [17:3]#	% SEE CHECKPOINTNULLV,ETC	15517400	35.027.799	41
42	15517405				15517405	340531285	42
43	15517600	, TEMPCNTF	= [27:10]#		15517600	35.027.799	43
44	15517800	, CPIXF	= [43: HEADERBITS]#		15517800	35.027.799	44
45	15517900	, RBCOUNTF	= [45:2]#	% COUNT OF ERRORHANDLER REBUILDS	15517900	35.027.799	45
46	15518000	, CPDKF	= [46:1]#		15518000	31.062.240	46
47	15518200	, CPPKF	= [47:1]#		15518200	31.062.240	47
48	15518210	% FIELDS IN PIB[PRINTLIMIT]			15518210	350792173	48
49	15518220	, BDNUMF	= [47:10]#		15518220	350792173	49
50	15518230	, PIBPRINTLIMF	= [37:38]#		15518230	350792173	50
51	15518250	% FIELDS IN PIB[PUNCHLIMIT]			15518250	350711762	51
52	15518300	, PIBTONGUEF	= [47: 8]#		15518300	350711762	52
53	15518350	, PIBPUNCHLIMF	= [39:40]#		15518350	350711762	53
54	15518400	% FIELDS IN PIB[RESTARTCOUNT]			15518400	33.045.739	54
55	15518600	, STACKSZF	= [47:16]#	% CLEARED BY INITIATE, SET BY RESTART	15518600	33.045.739	55
56	15518620	, TADSD1F	= [31: SNSIZE]#	% SEGDICTIONARY OF PROGRAM THAT INVOKE	15518620	350762003	56
57	15518640			% TADS	15518640	350531393	57
58	15518650	, TADSWANTEDF	= [19:1]#	% TADS TASK ATTRIBUTE SET	15518650	350531393	58
59	15518700	, TADSUSERF	= [18:1]#	% TADS ATTRIBUTE SET AND PROGRAM	15518700	350531393	59
60	15518750			% WAS COMPILED FOR TADS	15518750	350531393	60

1	15518760	, TADSVVALIDF	= [17:1]#	% TADS ATTRIBUTE HAS BEEN SET	15518760	350611718	1
2	15518770			% (PREVENTS DOCTOR FROM OVERRIDING	15518770	350611718	2
3	15518780			% PPB WITH COMPILE TIME PPB).	15518780	350611718	3
4	15518800	, RESTARTCOUNTF	= [16:17]#	% 2**17-1 RETRIES SHOULD BE ENOUGH	15518800	350611718	4
5	15518860	% FIELDS IN PIB [DBSINFO]			15518860	32.033.179	5
6	15518862	, DBSDYINGF	= [45:1]#	% SET TO 1 WHEN DBS IS PROCESSKILLED	15518862	33.041.627	6
7	15518864	, DBSSHUTDOWNF	= [44:1]#	% SET TO 1 TO TELL DBS TO GO AWAY	15518864	33.041.627	7
8	15518866	, STACKSINDMSOPENF	= [43:SNSIZE]#	% # OF STACKS WAITING IN DMSOPEN	15518866	35.026.736	8
9	15518868			% FOR DATABASE TO BECOME ACTIVE	15518868	33.041.627	9
10	15518870	, TOPOFDBSSCFW	= [31:16]#	% OFFSET OF SCW IN TOPMOST ENVIRONMENT	15518870	32.033.179	10
11	15518880	, TOPOFDBSF	= [15:16]#	% OFFSET FROM BOSR OF TOP OF FROZEN	15518880	32.033.179	11
12	15518900			% PORTION OF DBS (NEW DBS ONLY AFTER	15518900	32.033.179	12
13	15518920			% DMSFREEZE)	15518920	32.033.179	13
14	15518940	% FIELDS IN PIB[PORTLIB PIB]			15518940	35.005.118	14
15	15518950	, MPS COUNTF	= [39:20]#	% NO OF REMOTE FILES CONNECTED TO MPS	15518950	350772066	15
16	15518960	, PORT_COUNTF	= [19:20]#	% NO OF PORTS DECLARED AND TOUCHED	15518960	35.005.118	16
17	15519000	% FIELDS IN PIB[PROCESSFAMILYLINK]			15519000	31.170.674	17
18	15519200	, CAUSEREPLYEVENTF	= [47:1]#	% FOR DEATHINFAMILY COORDINATION	15519200	31.062.240	18
19	15519400	, EXTERNALPROCESSF	= [46:1]#	% THIS TASK HAS ITS OWN CODE FILE	15519400	31.062.240	19
20	15519600	, COPYPARAMF	= [45:1]#	% THERE IS A COPYDESC PARAMETER	15519600	31.071.002	20
21	15519800	% SEARCH CONTROL FIELD		% THIS STACK MAY:	15519800	31.062.240	21
22	15520000	, CANSEEF	= [44:1]#	% REFERENCE DATA IN PARENT STACK	15520000	31.062.240	22
23	15520200	, RUNCODEF	= [43:1]#	% COPY DESCRIPTORS FORM PARENT SEGDICT	15520200	31.062.240	23
24	15520400	, COPYDESCF	= [42:1]#	% COPY DESCRIPTORS FROM PARENT STACK	15520400	31.062.240	24
25	15520600	, BADPTRF	= [41:1]#	% ASSIGN "UPLEVEL" POINTER IN PARENT	15520600	31.062.240	25
26	15520800	, DRELBADPTRF	= [40:1]#	% ASSIGN UPLEVEL POINTER WITHIN OWN	15520800	31.062.240	26
27	15521000			% SCOPE (ACCORDING TO COMPILER)	15521000	31.062.240	27
28	15521200			% RUNCODE OR BADPTR IMPLIES COPYDESC;	15521200	31.062.240	28
29	15521400			% COPYDESC IMPLIES CANSEE	15521400	31.062.240	29
30	15521500	, GLOBALFILEEDGE	= [36:1]#	% GOT EDGE FROM JOB FOR GLOBAL FILE	15521500	32.123.458	30
31	15521600	, PARENTF	= [35:12]#		15521600	31.062.240	31
32	15521800	, SIBLINGF	= [23:12]#		15521800	31.062.240	32
33	15522000	, OFFSPRINGF	= [11:12]#		15522000	31.062.240	33
34	15522200	% FIELDS IN PIB[EXTERNALFAMILYLINK]			15522200	31.170.674	34
35	15522300	, EXTLINKAGELOCKEDF	= [47:1]#	% LINKAGE MAY NOT BE ALTERED	15522300	31.181.700	35
36	15522400	% PARENTF	= [35:12]#	% PARENTS STACK NUMBER	15522400	31.163.603	36
37	15522600	% SIBLINGF	= [35:12]#	% BROTHERS TAG	15522600	31.163.603	37
38	15522800	% OFFSPRINGF	= [11:12]#	% SONS TAG	15522800	31.163.603	38
39	15523200	% FIELDS IN PIB[HEADERACCESS]			15523200	35.013.388	39
40	15523400	, JOBFILEINDEXF	= [31:16]#	% DISKFILEHEADERS INDEX OF JOBFILE	15523400	35.013.388	40
41	15523600	, CODEFILEHEADERINDEXF			15523600	35.013.388	41
42	15523800		= [15:16]#	% DISKFILEHEADERS INDEX OF CODEFILE	15523800	35.013.388	42
43	15524400	% FIELDS IN PIB[JOBINFO]			15524400	31.170.674	43
44	15524600	, JOBSTKNRF	= [47:SNSIZE]#		15524600	350591706	44
45	15524660	, IBSF	= [35:6]#		15524660	350591706	45
46	15524680	, MYJOBFILEF	= [29:1]#		15524680	350591706	46
47	15524700	, CLASSBIT	= [28:1]#		15524700	350591706	47
48	15524720	, NOJOBSUMMARYIOBIT	= [27:1]#		15524720	351152609	48
49	15524740	%, AVAILABLE	= [26:7]#	% SAVE 2 BITS FOR STKNRF ADJUSTMENT	15524740	351112511	49
50	15524800	, JOBCLASSF	= [19:10]#		15524800	31.062.240	50
51	15525200	, DECKGROUPNOF	= [9:10]#		15525200	31.062.240	51
52	15526000	% FIELDS IN PIB[OPTION]			15526000	31.170.674	52
53	15526200	, LONGCLSF	= [0:1]#		15526200	31.141.003	53
54	15526400	, FAULTCLSF	= [1:1]#		15526400	31.062.240	54
55	15526600	, DSEDCLSF	= [2:1]#		15526600	31.062.240	55
56	15526800	% CCLISTCLSF	= [3:1]#		15526800	31.138.003	56
57	15527000	, BACKUPPRINTCLSF	= [4:1]#		15527000	31.062.240	57
58	15527200	, AUTORMCLSF	= [5:1]#		15527200	31.062.240	58
59	15527400	, BDBASECLSF	= [6:1]#		15527400	31.062.240	59
60	15527600	, BASECLSF	= [7:1]#	% NOTE:	15527600	31.062.240	60



1	15527800	, ARRAYSCLSF	= [8:1]#	% IF ADDING DUMP-CONTROL OPTIONS,	15527800	31.062.240	1
2	15528000	, CODECLSF	= [9:1]#	% EMBELLISH THE DEFINE "DSOPTIONS"	15528000	31.062.240	2
3	15528200	, FILESCLSF	= [10:1]#	% IN MIXREQUEST OF SETSTATUS.	15528200	31.062.240	3
4	15528400	, BACKUPPUNCHCLSF	= [11:1]#		15528400	31.062.240	4
5	15528600	, NOSUMMARYF	= [12:1]#		15528600	31.062.240	5
6	15528800	% HARDDUMPF	= [13:1]#		15528800	31.138.003	6
7	15529000	, PRIVATETASKF	= [14:1]#		15529000	31.062.240	7
8	15529200	, DBSDUMPF	= [15:1]#		15529200	31.062.240	8
9	15529400	% PDSDUMPF	= [16:1]#	***** AVAILABLE *****	15529400	32.092.773	9
10	15529420	, IPCOVERRIDEF	= [17:1]#		15529420	31.191.727	10
11	15529450	, SIBDUMPF	= [18:1]#		15529450	31.185.740	11
12	15529500	, LIBRARYDUMPF	= [19:1]#		15529500	31.185.740	12
13	15529550	, PRIVLIBDUMPF	= [20:1]#		15529550	31.185.740	13
14	15529600	% FIELDS IN PIB[PATHCONTROL]			15529600	31.170.674	14
15	15529800	, DESTCONTROLF	= [47:24]#	% VALIDITYBIT IS FOR THIS FIELD	15529800	31.062.240	15
16	15530000	, DESTMCSF	= [45:6]#		15530000	31.062.240	16
17	15530200	, DESTINATIONUNITF	= [39:16]#		15530200	31.062.240	17
18	15530400	, DESTISREMOTEF	= [39:1]#		15530400	31.062.240	18
19	15530600	, DESTUNITF	= [38:15]#		15530600	31.062.240	19
20	15530800	, ORGCONTROLF	= [23:24]#		15530800	31.062.240	20
21	15531000	, ORGWANTSF	= [23:2]#		15531000	31.062.240	21
22	15531200	, ORGWANTSSUMMARYF	= [23:1]#		15531200	31.062.240	22
23	15531400	, ORGWANTSMESSAGEF	= [22:1]#		15531400	31.062.240	23
24	15531600	, ORGMCSF	= [21:6]#		15531600	31.062.240	24
25	15531800	, ORIGINALUNITF	= [15:16]#		15531800	31.062.240	25
26	15532000	, ORGISREMOTEF	= [15:1]#		15532000	31.062.240	26
27	15532200	, ORGUNITF	= [14:15]#		15532200	31.062.240	27
28	15532400	% FIELDS IN PIB[REPLY]			15532400	31.170.674	28
29	15532600	, REPATNF	= [47:1]#	% TRUE IF 46:47 IS A MASK	15532600	31.062.240	29
30	15532800	, REPINFOF	= [45:40]#	% INFO DEPENDING ON TYPE OF REPLY	15532800	31.062.240	30
31	15533000	, REPINFO1F	= [45:20]#	% SUBFIELD 1	15533000	31.062.240	31
32	15533200	, REPINFO2F	= [25:20]#	% SUBFIELD 2	15533200	31.062.240	32
33	15533400	, REPVALF	= [5:6]#	% REPLY NUMBER (0 - 46)	15533400	31.062.240	33
34	15533600	% FIELDS IN PIB[SERIAL]			15533600	31.170.674	34
35	15533800	, SERIALSTKNRF	= [47:SNSIZE]#		15533800	350591706	35
36	15534000	, JOBNRF	= [31:16]#		15534000	31.062.240	36
37	15534200	, TSKNRF	= [15:16]#		15534200	31.062.240	37
38	15534400	% FIELDS IN PIB[USAGE]		% SET BEFORE INIT. OR FROM OWN STACK ONLY	15534400	321781213	38
39	15534600	, STKMCSSTATUSF	= [47:8]#		15534600	31.062.240	39
40	15534800	, STKISMCSF	= [47:1]#		15534800	31.062.240	40
41	15535000	, STKMCSNRF	= [46:7]#		15535000	31.062.240	41
42	15535200	, LIBPERMANENTF	= [39:1]#		15535200	34.017.440	42
43	15535300	, BNACOUNTF	= [38:6]#	% # BEGINHSUSE CALLS - ENDHSUSE CALLS	15535300	33.008.209	43
44	15535400	, SHARINGSPEC	= [32:4]#	% LIBRARIES: SHARING SPECIFICATION	15535400	351022457	44
45	15535600	, WARNIDDFLEF	= [28:1]#	% REMOVE ON 3.6	15535600	350762002	45
46	15535800	, INHIBSOFTINTSF	= [27:2]#	% NEXT TWO BITS:	15535800	31.062.240	46
47	15536000	, GENDISABLEF	= [27:1]#	% INTERRUPTS ARE DISABLED GENERALLY	15536000	31.062.240	47
48	15536200	, DOINGSOFTINTF	= [26:1]#	% WE ARE PROCESSING SOFT INTERRUPT	15536200	31.062.240	48
49	15536400	, PBITSTATEF	= [25:2]#		15536400	31.062.240	49
50	15536500	, UNITSWARNEDF	= [23:1]#	% MCP STATUS WARNING (REMOVE ON 37)	15536500	350451294	50
51	15536520	, GETSTATUSWARNEDF	= [22:1]#	% GETSTATUS WARNING GIVEN (REMOVE 36)	15536520	350471330	51
52	15536600	, IORESULTWARNEDF	= [21:1]#	% DIRECTBUF.IORESULT FOR MLI SYSTEMS	15536600	350431234	52
53	15536800			% CHANGED WARNING GIVEN (REMOVE @ 36)	15536800	350431234	53
54	15537200	, NOMEMF	= [20:1]#		15537200	31.062.240	54
55	15537400	, DUMPINGF	= [19:1]#		15537400	31.062.240	55
56	15537600	, SORTTINGF	= [18:1]#		15537600	31.062.240	56
57	15537800	, UNITATTACHEDF	= [17:1]#		15537800	31.062.240	57
58	15538000	, LIBSTATEF	= [16:3]#	% LIBRARY STATE	15538000	31.183.733	58
59	15538150	, LIBINITBYLINKER	= 1 #	% INITIATED BY LIBRARY LINKER	15538150	31.183.733	59
60	15538200	, LIBINITBYUSER	= 2 #	% NORMAL TASK STARTING TO FREEZE	15538200	31.183.733	60

1	15538250	, LIBFROZEN	= 3 #	%	USABLE STATIC LIBRARY	15538250	31.183.733	1
2	15538300	, LIBTHAWING	= 4 #	%	UNLINKABLE BUT STILL IN USE	15538300	31.183.733	2
3	15538350	, LIBRESUMED	= 5 #	%	NO LONGER LINKED NOR LINKABLE	15538350	31.183.733	3
4	15538400	, LIBNEVERFROZE	= 6 #	%	INIT BY LINKER BUT NEVER FROZE	15538400	31.183.733	4
5	15538600	, NOHISTORYF	=[13:1]#			15538600	31.062.240	5
6	15538700	, SYSTEMLIBRARYF	=[12:1]#	%	ALLOWED SYSTEM LIBRARY PRIVILEGES	15538700	321741167	6
7	15538800	, FREEZEVERRIDEF	=[11:1]#	%	OVERRIDE THE USERS FREEZE.	15538800	33.030.547	7
8	15539000	, DONTSTRECHF	=[10:1]#	%	DO NOT STRETCH THIS STACK	15539000	32.148.991	8
9	15539100	, BLOCKSEARCHLOOPF	=[ 9:1]#	%	GOTOSOLVER LOOPING ON BLOCKSEARCH	15539100	31.212.346	9
10	15539120	, SEGPTRWARNEDF	=[ 8:1]#	%	FAULT(S) DUE TO UNINDEXED SEGMENTED	15539120	35.015.521	10
11	15539140				POINTER ARGS TO STRING OPS; HAPPENS	15539140	35.015.521	11
12	15539160				ON B7XXX WITH PRE 3.4 STRING CODE.	15539160	35.015.521	12
13	15539200	, WARNSECURITYF	=[ 7:1]#	%	REMOVE ON 3.6	15539200	350762002	13
14	15539250	, CURRENTASNF	=[ 6:6]#	%	FOR MOVESTACK, CHANGED BY SELECTBOX	15539250	34.004.078	14
15	15539300	, ARP	=[ 0:1]#	%	ASSOCIATION OF RETIRED PROCESSORS	15539300	32.072.627	15
16	15539400	% FIELDS IN PIB[SCHEDDATA]			WHILE STATUS LEQ SELECTED	15539400	31.170.674	16
17	15539500	, SCHEDULEQLINKF	=[47:SNSIZE]#	%	NEXT STACK IN SCHEDULE QUEUE	15539500	350591706	17
18	15539800	, SCHEDD2SIZEF	=[35:16]#	%	LENGTH OF TASK STACK	15539800	31.140.002	18
19	15540000	, SCHEDCOREF	=[19:20]#	%	CORE ESTIMATE	15540000	31.140.002	19
20	15540200	% FIELDS IN PIB[STATION]				15540200	31.170.674	20
21	15540400	, TASKTANKF	=[33:2]#			15540400	31.062.240	21
22	15540600	, STATIONF	=[15:16]#			15540600	31.138.003	22
23	15540800	% FIELDS IN PIB[STOPPOINT]				15540800	31.170.674	23
24	15541000	, FAULTNUMF	=[47:8]#	%	FAULT REASON	15541000	31.062.240	24
25	15541100	, FAULTHANDLED F	=[39:1]#	%	ON IF FAULT PROCEDURE HANDLED IT	15541100	33.042.678	25
26	15541200	% RCWGUTSF	=[35:36]			15541200	31.062.240	26
27	15541400	% FIELDS IN PIB[HISTORY]				15541400	31.170.674	27
28	15541440	, NOFAULTOFHISF	=[47:1]#			15541440	321731114	28
29	15541460	, BITOFHISTORYF	=[46:1]#			15541460	321731114	29
30	15541480	, IGNOREDSEDF	=[45:1]#			15541480	321731114	30
31	15541500	, HISTINFOF	=[43:20]#			15541500	321731114	31
32	15541520	, HISTRCTF	=[23:24]#	%	COMBINED REASON,CAUSE,TYPE FIELDS	15541520	321731114	32
33	15541540	, HISREASONF	=[23:8]#			15541540	321731114	33
34	15541560	, HISCAUSEF	=[15:8]#			15541560	321731114	34
35	15541600	, HISTYPEF	=[7:8]#			15541600	31.062.240	35
36	15543000	% FIELDS IN PIB[TASKLIMIT]				15543000	31.170.674	36
37	15543100	, TASKDECLARERF	=[47:SNSIZE]#	%	STACK DECLARING PIB	15543100	35.026.736	37
38	15543110	% , UNUSED	=[35:8]#			15543110	34.035.841	38
39	15543120	, SWVALIDF	=[27:8]#	%	VALIDITY BITS FOR SWITCHES	15543120	33.104.993	39
40	15543140	, SWITCHESF	=[19:8]#	%	COBOL SWITCHES 1 TO 8	15543140	33.104.993	40
41	15543145	% UNUSED	=[11:2]#			15543145	33.104.993	41
42	15543150	, JOBSUMMARYF	=[ 9:4]#	%	CONTAINS JOBSUMMARY VALUE	15543150	33.044.732	42
43	15543200	, TASKLIMVALIDF	=[ 5:1]#			15543200	31.062.240	43
44	15543400	, TASKLIMF	=[ 4:5]#			15543400	31.062.240	44
45	15543450	% FIELDS IN PIB[LIBINITIATOR], ONLY VALID BEFORE LIBRARY FREEZES				15543450	32.147.947	45
46	15543500	, LIBINITSTKNRF	=[47:SNSIZE]#	%	SN OF INITIATOR OF THIS LIBRARY	15543500	350591706	46
47	15543520	, LIBRARYUSERF	=[35:SNSIZE]#	%	SN OF USER OF THE LIBRARY.	15543520	35.026.736	47
48	15543540	, INBYCALLINGPROCF	=[4:1]#	%	STACK IS IN A LIB BYCALLING PROC	15543540	351022457	48
49	15543550	, LIBACCESSTYPEF	=[3:4]#	%	VALUE OF LIBACCESS PARAM IN TEMPLTE	15543550	32.147.947	49
50	15543600	% FIELDS IN PIB[TASKINFO]			% SET BEFORE INIT. OR WITH PROCESSCHANGELOCK	15543600	321781213	50
51	15543800	, INUSESYSTEMF	=[47:1]#	%	TASK IS BEING STARTED OR RUNNING	15543800	321781213	51
52	15544000	, COMPILETYPEF	=[46:3]#	%	SEE COMPILETYPE	15544000	321781213	52
53	15544100	, PROGRAMLOCKEDF	=[43:1]#	%	ODT <MIXNO> LP	15544100	321781213	53
54	15544200	, QPRIORLIMTF	=[42:7]#	%	PRIORITY LIMIT FROM QUEUE	15544200	321781213	54
55	15544250	, ASN_WARNING_GIVEN	=[35:1]#	%	1 IF THE WARNING WAS GIVEN ALREADY	15544250	351612827	55
56	15544300	% AVAILABLE	=[34:1]#			15544300	351612827	56
57	15544400	, TASKVISIBLEF	=[33:1]#	%	TO BE INITIATED AS VISIBLE TASK	15544400	321781213	57
58	15544500	, RUNDBSF	=[32:1]#	%	DBS INITIATED BY DMSOPEN	15544500	321781213	58
59	15544550	, CRITICALSTACKF	=[31:1]#	%	B79 APS ALLOWED IF LAST CPM DIES	15544550	35.026.762	59
60	15544700	, OLAYFILELINKF	=[19:SNSIZE]#	%	STKNR OF SHARER OF OLAYFILE, OR O	15544700	35.026.736	60

1	15544720	% THE FIELD OLAYFILELINKF IS SEARCHED	15544720	34.016.398	1
2	15544730	% BY OLAYSCOUT UNDER OLAYINFOLOCK	15544730	34.016.398	2
3	15544740	% ( A BUZZ LOCK). THEREFORE	15544740	34.016.398	3
4	15544750	% BOJ & EOJ MUST OWN OLAYINFOLOCK	15544750	34.016.398	4
5	15544760	% AND PROCESSCHANGELOCK WHEN THEY	15544760	34.016.398	5
6	15544770	% CHANGE THIS FIELD.	15544770	34.016.398	6
7	15544800	%, AVAILABLE =:[7:8]#	15544800	35.013.388	7
8	15545000	% PIB[TASKPARAMS] IS USED VARIOUSLY ACCORDING TO THE PARAMETERS BEING	15545000	31.170.674	8
9	15545200	% SPECIFIED FOR THE TASK, THE MEANS OF THEIR SPECIFICATION, AND THE	15545200	31.071.002	9
10	15545400	% STAGES OF INITIALIZATION OF THE TASK:	15545400	31.071.002	10
11	15545600	% 1: A ZERO OPERAND MEANS THERE ARE NO PARAMETERS.	15545600	31.071.002	11
12	15545800	% 2: AN ORDINARY TASK INITIATION VIA DELIVERY/INITIATEUSERTASK	15545800	31.071.002	12
13	15546000	% NOTES THE DISPLACEMENT INTO THE PARENT STACK AND THE NUMBER OF	15546000	31.071.002	13
14	15546200	% WORDS OF PARAMETERS IN PARAMDISP AND PARAMWORDS OF AN OPERAND.	15546200	31.071.002	14
15	15546400	% SIMILAR ACTION IS TAKEN BY FORKCONTROLCARD FOR A ZIP.	15546400	31.071.002	15
16	15546460	% 3: WITH A KLUDGE INSTALLED IN MARK 31.1, INITIATEUSERTASK NOTICES	15546460	35.003.072	16
17	15546500	% ARRAY PARAMETERS FROM WFL OR FROM MCS TO SESSION OR TO FOREIGN	15546500	35.003.072	17
18	15546520	% TASK AND BUILDS AN (MCP) OVERLAYABLE ARRAY IN TASKPARAMS TO	15546520	35.003.072	18
19	15546530	% PASS THE ARRAY BY VALUE. UNRAVEL RETURNS LENGTH AND TEXT	15546530	35.003.072	19
20	15546540	% OF THE ARRAY PARAMETER TO HOST SERVICES VIA AD-HOC ATTRIBUTES.	15546540	31.191.776	20
21	15546560	% 4: SINCE MARK 31.1, THERE HAS BEEN A FIELD DEFINE (PARAMDESCLINK)	15546560	35.003.072	21
22	15546570	% AND SOME SKELETAL CODE IN INITIATEUSERTASK FOR AN UNIMPLEMENTED	15546570	35.003.072	22
23	15546580	% FEATURE:	15546580	35.003.072	23
24	15546600	% IF THE USER SPECIFIES PARAMETERS VIA MUTATE, TASKPARAMS BECOMES	15546600	35.003.072	24
25	15546800	% A MOM FOR AN (MCP) OVERLAYABLE ARRAY HOLDING HIS PARAMETER	15546800	31.071.002	25
26	15547000	% VALUES AND A BINDINFO-FORMAT DESCRIPTION OF HIS PARAMETERS.	15547000	31.071.002	26
27	15547200	% INITIATEUSERTASK AND/OR EXTERNALREFERENCE OF DOCTOR WILL CHECK	15547200	31.071.002	27
28	15547400	% THAT THE SUPPLIED PARAMETERS FIT THE CALLED PROCEDURE.	15547400	31.071.002	28
29	15547600	% 5: IF A FORK INVOLVES PARAMETERS, THEY ARE COPIED INTO THE GETAREA	15547600	35.003.072	29
30	15547800	% MESSAGE ENQUEUED FOR ANABOLISM, WHO THEN USES THE SAME MEMORY	15547800	31.071.002	30
31	15548000	% TO BUILD A PARAMETER ARRAY WITH MOM AT TASKPARAMS.	15548000	31.071.002	31
32	15548200	% 6: IF EXTERNALREFERENCE FINDS PARAMETERS ON THE STACK (STATE 2)	15548200	35.003.072	32
33	15548250	% AND THE PARAMETERS MISMATCH THE EXTERNAL PROCEDURE BUT CAN BE	15548250	35.003.072	33
34	15548300	% COERCED, A PARAMETER ARRAY (GETAREA) IS BUILT WITH COERCION	15548300	35.003.072	34
35	15548350	% INSTRUCTIONS IN THE PARAMETER SLOTS. THIS CASE IS DISTINGUISHED	15548350	35.003.072	35
36	15548400	% FROM CASES 3-5 BY RETAINING NON-ZERO PARAMDISP IN PARAMINFO.	15548400	35.003.072	36
37	15548450	% (IF COERCIBLE MISMATCH IS FOUND FROM STATE 3 OR 4, THE	15548450	35.003.072	37
38	15548500	% EXISTING TASKPARAMS ARRAY IS ADJUSTED; NO NEW STATE ARISES.)	15548500	35.003.072	38
39	15548550	% 7: THE FOLLOWING ACTIONS ARE TAKEN BY THE FIRST PART OF INITIATE	15548550	35.003.072	39
40	15548600	% (WHEN A STACK NUMBER HAS BEEN OBTAINED): IF TASKPARAMS	15548600	35.003.072	40
41	15548650	% CONTAINS A NON-ZERO PARAMDISP (STATE 2), TASKPARAMS IS CHANGED	15548650	35.003.072	41
42	15548700	% TO AN ARRAY (GETAREA) AND THE PARAMETERS ARE COPIED INTO IT	15548700	35.003.072	42
43	15548750	% FROM THE PARENT STACK. IF TASKPARAMS IS AN ARRAY AND THE	15548750	35.003.072	43
44	15548800	% PARAMINFO WORD HAS NON-ZERO PARAMDISP (STATE 6), THE PARAMETERS	15548800	35.003.072	44
45	15548850	% ARE COPIED FROM THE STACK INTO THE ARRAY, WITH ANY REQUIRED	15548850	35.003.072	45
46	15548900	% ADJUSTMENTS.	15548900	35.003.072	46
47	15548950	% IF THERE ARE COPY-DESCRIPTOR PARAMETERS IN THE PARAMETER ARRAY,	15548950	35.003.072	47
48	15549000	% A COPY OF THE TASKPARAMS DESCRIPTOR IS PUT IN THE NEW	15549000	35.003.072	48
49	15549050	% STACKVECTOR SLOT FOR SEARCHING.	15549050	35.003.072	49
50	15549100	% FOR A SWAPJOB, AN APPENDIX TO THE ARRAY RETAINS THE OWNER STACK	15549100	35.003.072	50
51	15549150	% FOR EACH COPY DESCRIPTOR OR PSEUDOSTACK SIRW.	15549150	35.003.072	51
52	15549400	% 8: AFTER THE SECOND PART OF INITIATE HAS ACQUIRED STACK SPACE, THE	15549400	35.003.072	52
53	15549600	% APPROPRIATE WORDS ARE COPIED INTO THE NEW STACK BEING BUILT.	15549600	31.071.002	53
54	15549800	% UNLESS THERE ARE ARRAYS BY VALUE, THE PARAMETER ARRAY IS	15549800	31.071.002	54
55	15550000	% DISCARDED AND PARAMINFO SAVED IN TASKPARAMS (FOR STATE 10).	15550000	35.003.072	55
56	15550200	% IF ANY ARRAY IS BEING PASSED BY VALUE, AN EXTRA BLOCK IS PUT	15550200	31.071.002	56
57	15550400	% INTO THE NEW STACK BETWEEN THE USER OUTER BLOCK AND THAT OF	15550400	31.071.002	57
58	15550600	% EOJ, TO HOLD THE MOM(S).	15550600	31.071.002	58
59	15550800	% 9: IF THERE IS STILL AN ARRAY AT BOJ TIME, NORMALBOJ SETS UP MOM	15550800	35.003.072	59
60	15551000	% DESCRIPTORS FOR ANY ARRAYS BEING PASSED BY VALUE AND COPIES	15551000	31.071.002	60

1	15551200	%	THEIR CONTENTS FROM THE TASKPARAMS ARRAY. THE PARAMETER ARRAY	15551200	31.071.002	1
2	15551400	%	IS THEN DISCARDED AND PARAMINFO SAVED IN TASKPARAMS.	15551400	35.003.072	2
3	15551600	%	10: BY THE TIME THE USER PROGRAM IS ENTERED, TASKPARAMS IS AN	15551600	35.003.072	3
4	15551800	%	OPERAND CONTAINING PARAMWORDS AND PARAMBLOCK.	15551800	31.071.002	4
5	15552000	%	INDICES INTO ARRAYSPIB[SN,TASKPARAMS,*]	15552000	35.003.072	5
6	15552200	%	TOSCW = 0# % DUMMY TOSCW FOR SEARCHING AS STACK	15552200	31.071.002	6
7	15552400	,	PARAMINFO = 1# % PARAMBLOCK/PARAMWORDS	15552400	31.071.002	7
8	15552600	,	PARAMLINKX = 2# % PARAMDESCLINK/PARAMLINK	15552600	31.071.002	8
9	15552800	,	PARAMFIXEDL = 3# % LINKS POINT HERE OR LATER	15552800	31.071.002	9
10	15553000	%	FIELDS IN SPIB[SN,TASKPARAMS] OR ARRAYSPIB[SN,TASKPARAMS,PARAMINFO]	15553000	35.003.072	10
11	15553200	,	PARAMDISP =[47:16]# % DISPLACEMENT IN PARENT STK OF 1ST PARAM	15553200	31.071.002	11
12	15553400	,	PARAMBLOCK =[31:16]# % WIDTH OF EXTRA MOM-PARAMETER BLOCK	15553400	31.071.002	12
13	15553600	,	PARAMWORDS =[15:16]# % NUMBER OF WORDS OF PARAMETERS	15553600	31.071.002	13
14	15553800	%	FIELDS IN ARRAYSPIB[SN,TASKPARAMS,PARAMLINKX]	15553800	35.003.072	14
15	15554000	,	PARAMDESCLINK =[31:16]# % INDEX OF PARAMETER "BINDINFO"	15554000	31.071.002	15
16	15554200	,	PARAMLINK =[15:16]# % INDEX OF 1ST PARAMETER IMAGE	15554200	31.071.002	16
17	15554400	%	PARAMLINK POINTS TO A STACK IMAGE OF PARAMBLOCK+PARAMWORDS WORDS.	15554400	31.071.002	17
18	15554600	%	IF PARAMBLOCK IS NON-0, THE FIRST PARAMBLOCK-3 WORDS OF THE IMAGE	15554600	31.071.002	18
19	15554800	%	ARE MOM SPECIFIERS, FOLLOWED BY PLACEHOLDERS FOR AN SCW, MSCW AND	15554800	31.071.002	19
20	15555000	%	RCW FOR THE MOM BLOCK. THE PARAMETER IMAGE FOLLOWS THE BLOCK IMAGE.	15555000	31.071.002	20
21	15555200	%	THE PARAMETER SLOT FOR ANY ARRAY-BY-VALUE IS A TAGBLASTED COPY	15555200	32.054.458	21
22	15555400	%	DESCRIPTOR, WHICH EVENTUALLY BECOMES A COPY OF THE LOCAL MOM.	15555400	32.054.458	22
23	15555600	%	WHEN PARAMETERS ARE ORIGINATED BY MUTATE, PARAMDESCLINK POINTS TO	15555600	31.071.002	23
24	15555800	%	A BINDINFO-FORMAT PARAMETER DESCRIPTION FOR THE PARAMETERS PRESENT.	15555800	31.071.002	24
25	15556000	%	FIELDS IN A MOM SPECIFIER IN ARRAYSPIB[SN,TASKPARAMS,*]	15556000	35.003.072	25
26	15556200	,	PCONLINK =[47:16]# % INDEX OF 1ST WORD OF ARRAY VALUE	15556200	31.071.002	26
27	15556400	,	PCONLEN =[31:16]# % LENGTH OF ARRAY VALUE (WORDS)	15556400	31.071.002	27
28	15556600	,	PCOPYLINK =[15:16]# % RELATIVE DISPLACEMENT TO COPY PARAMETER	15556600	35.003.072	28
29	15556620	%	COPY SLOT HAS A TAG-BLASTED COPY DESCRIPTOR WHOSE ADDRESSF IS THE	15556620	35.003.072	29
30	15556640	%	INDEX INTO TASKPARAMS OF THE CORRESPONDING MOM SPECIFIER.	15556640	35.003.072	30
31	15556660	%	FIELDS IN A COERCION SPECIFIER IN ARRAYSPIB[SN,TASKPARAMS,*]	15556660	35.003.072	31
32	15556680	,	PCOERCIONTYPE =[45: 2]# % COERCION ACTION REQUIRED:	15556680	35.003.072	32
33	15556700	,	PCOERCECONST = 0# % COPY THE COERCION SPECIFIER	15556700	35.003.072	33
34	15556720	,	PCOERCECOPY = 1# % COPY THE WORD AT PCOERCIONDISP	15556720	35.003.072	34
35	15556740	,	PCOERCIONSZ =[42: 3]# % SZF VALUE FOR COERCED DESCRIPTOR	15556740	35.003.072	35
36	15556760	,	PCOERCIONBITS =[39: 8]# % BITSPERSZ FOR COERCED DESCRIPTOR, OR 0	15556760	35.003.072	36
37	15556780	,	PCOERCIONDISP =[15:16]# % PARENT-STACK DISPLACEMENT OF PARAMETER	15556780	35.003.072	37
38	15556800	%	FIELDS IN PIB[SURROGATEINFO]	15556800	31.170.674	38
39	15556900	,	SURROGATESTKNRF =[47:SNSIZE]# % STACK LOCKING THE TASK	15556900	350591706	39
40	15557000	,	SURROGATEF =[35:1]# % TASK ORIGINATED AT ANOTHER HOST	15557000	31.181.700	40
41	15557200	,	RUNNINGEXTERNALF =[34:1]# % TASK RUNNING AT ANOTHER HOST	15557200	31.181.700	41
42	15557300	,	TASKLOCKEDF =[33:1]# % HOSTSERVICES VS MCP LOCKOUT SWITCH	15557300	32.055.474	42
43	15557350	,	EXTERNALSTATUSF =[32:8]# % STATUS OF AN EXTERNAL TASK	15557350	32.077.694	43
44	15557600	,	SURROGATEINDEXF =[11:SNSIZE]# % IF RUNNINGEXTERNALF, THIS IS THE	15557600	350591706	44
45	15558000	%	INDEX INTO THE EXTERNAL PIB VECTOR.	15558000	32.046.351	45
46	15558400	%	FIELDS IN WORD 0 OF ITINERARY	15558400	32.044.304	46
47	15558450	,	ITINERARYINHERITEDF =[47:01]# % TRUE IF ITINERARY WAS INHERITED	15558450	321741144	47
48	15558500	,	ITINERARYLENGTHF =[15:16]# % LENGTH (IN BYTES) OF ITINERARY	15558500	321741144	48
49	15558506	%	FIELDS IN PIB[USERCODEPRIV]	15558506	32.055.474	49
50	15558508	,	UCODEANDMCSF =[1:2]#	15558508	32.055.474	50
51	15558510	,	USERCODEPRIVF =[0:1]# % THIS USERCODE IS PRIVILEGED	15558510	32.077.694	51
52	15558512	,	MCSABSTAININGF =[1:1]# % MCS ABSTAINS FROM MCS PRIVILEGE	15558512	32.077.694	52
53	15558514	,	HOSTSERVICESRINGBIT =[2:1]# % THIS BIT PROVIDES THE EQUIVALENT OF	15558514	340631689	53
54	15558516	%	MCSABSTAININGF FOR HOST SERVICES. THE	15558516	32.055.474	54
55	15558518	%	BIT IS DEFINED AS FOLLOWS:	15558518	32.055.474	55
56	15558520	%	0 = THIS STACK MAY GET SPECIAL	15558520	32.055.474	56
57	15558522	%	PRIVILEGES WHILE IT IS	15558522	340631689	57
58	15558524	%	RUNNING CODE FROM THE HOST	15558524	32.055.474	58
59	15558526	%	SERVICES LIBRARY.	15558526	32.055.474	59
60	15558528	%	1 = THIS STACK CAN NOT GET THE	15558528	32.055.474	60

1	15558530	%	SPECIAL PRIVILEGES ASSOCIATED	15558530	340631689	1
2	15558532	%	WITH RUNNING HOSTSERVICES CODE.	15558532	32.055.474	2
3	15558534	%	THIS BIT'S DEFAULT IS 0. IT IS NOT	15558534	340631689	3
4	15558536	%	EXPLICITLY SET UP ANYWHERE; WE ASSUME	15558536	340631689	4
5	15558538	%	THAT UNLESS STATED OTHERWISE, ANYONE	15558538	340631689	5
6	15558540	%	IS ALLOWED TO PASS "IF	15558540	32.055.474	6
7	15558542	%	HOSTSERVICESCALLING" TESTS IN THE MCP.	15558542	32.055.474	7
8	15558544	, HSSTATUSTURNEDOFF	=[3:1]# % FLAG TO AVOID SENDING STATUS MESSAGES	15558544	32.077.694	8
9	15558546		% WHILE RUNNING HOSTSERVICES CODE,	15558546	32.077.694	9
10	15558548		% SET WHILE ON TOP ON USERS STACK	15558548	32.077.694	10
11	15558549	, CHARGEREQF	=[4:1]#	15558549	30.7X.330	11
12	15558552	, KEYEDIOCALLINGF	=[5:1]# % WE ARE RUNNING ON TOP OF KEYEDIO.	15558552	33.101.858	12
13	15558554	, INHERITMCSSTATUSF	=[6:1]# % TRUE STOPS THIS TASK FROM INHERITING	15558554	331051031	13
14	15558556		% "MCSNESS" FROM IT'S PARENT WHEN	15558556	331051031	14
15	15558558		% INITIATED.	15558558	331051031	15
16	15558560	, USEDEFLT	=[7:1]# % USEDEFAULTCHARGE BIT SET FOR USER.	15558560	340621631	16
17						17
18						18
19						19
20						20
21						21
22						22
23						23
24						24
25						25
26						26
27						27
28						28
29						29
30						30
31						31
32						32
33						33
34						34
35						35
36						36
37						37
38						38
39						39
40						40
41						41
42						42
43						43
44						44
45						45
46						46
47						47
48						48
49						49
50						50
51						51
52						52
53						53
54						54
55						55
56						56
57						57
58						58
59						59
60						60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	FILE INFORMATION BLOCK	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	APPENDIX F	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1	53627400	,INPUTTRANS = FIBW[51]#	% (1,32)	53627400	26.010.131	1
2	53627500	,USEROUTINES = FIBW[52]#	% (1,33)	53627500	26.010.131	2
3	53627600	,FLOPPYMISC = FIB[53]#	% (1,34)	53627600	31.018.136	3
4	53627700	,FIBLOCKSNR = FIB[54]#	% (1,35) FIBLOCK OWNERS SNR	53627700	31.018.136	4
5	53627800	,IOCB = FIBW[55]#	% (1,36) IOCB DESCRIPTOR	53627800	31.036.239	5
6	53627850	,LOGICALCOUNT = FIB[56]#	% (1,37) ACTUAL I/O COUNT	53627850	31.192.787	6
7	53627880	,LIBRARYINFO = FIB[57]#	% (1,38) INDIO & PORTS	53627880	32.073.641	7
8	53627885	,PHYSICALIOCOUNT = FIB[58]#	% (1,39) PHYSICAL I/O COUNT	53627885	33.003.091	8
9	53627888	,LOGINTERVALSTARTTIME = FIB[59]#	% (1,3A) COUNTER FOR LOGINT	53627888	33.026.464	9
10	53627890	,SELFDISC = FIBW[60]#	% (1,3B) SELF POINTER	53627890	33.026.464	10
11	53627892	,BUFLINKS = FIB[61]#	% (1,3C) BUF LINKAGE INFO	53627892	35.009.306	11
12	53627894	,IOMOM = FIBW[62]#	% (1,3D) MOM FOR IOAREA	53627894	35.009.306	12
13	53627896	,SIOMOM = FIBW[63]#	% (1,3E) MOM FOR SIOAREA	53627896	35.009.306	13
14	53627898	,UNITNUMBERS = FIB[64]#	% (1,3F) SIO/IO UNITNUMBERS	53627898	35.028.840	14
15	53627900	,INQ LIST DESC = FIBW[65]#	% (1,40) OLD INQ LIST	53627900	350721873	15
16	53627920	,INQ LIST_ARRAY = FIB2W[65,*]#	% (1,40)	53627920	350721873	16
17	53627930	,FABPLACE = 66 #		53627930	351012405	17
18	53627940	,FAB_DESC = FIBW[66] #	% (1,41) FILE ATTRIBUTES	53627940	351012405	18
19	53627950	%%%		53627950	.	19
20	53628000	% LABEL-EQUATION BLOCK (LEB)	%	53628000	.	20
21	53628100	% SAME FORMAT ALSO USED IN LABEL TABLE AND FPB	%	53628100	.	21
22	53628200	%%%		53628200	.	22
23	53628300	% FIXED PART LEB		53628300	.	23
24	53628400	,LEBC = LEB[0]#	% BASIC DATA: SIZE,KIND ...	53628400	35.006.200	24
25	53628500	,LEBO(LEB) = LEB[0]#		53628500	35.006.200	25
26	53628600	,GEN1 = GEN_1(LEB)#	% REEL, CYCLE, VERSION	53628600	35.023.628	26
27	53628650	,GEN_1(LEB) = LEB[1]#		53628650	35.023.628	27
28	53628700	,GEN2 = GEN_2(LEB)#	% DATE, SAVE FACTOR	53628700	350411173	28
29	53628750	,GEN_2(LEB) = LEB[2]#		53628750	350411173	29
30	53628800	,DSKS = LEB[3]#	% DISK INFO AND MYUSE,OTHERUSES	53628800	35.006.200	30
31	53628900	,OPENTIME = LEB[4]#	% TIME FILE OPEN	53628900	35.006.200	31
32	53629000	,LEBDATA2 = LEB[5]#	% ORIGINAL TANKDATA2.	53629000	35.006.200	32
33	53629100	,ATTVALUE = LEB[6]#		53629100	35.006.200	33
34	53629150	,DISKMISCINX = 7#		53629150	321761190	34
35	53629200	,DISKMISC = LEB[DISKMISCINX]#		53629200	35.006.200	35
36	53629400	,TANKDATA3 = LEB[8]#		53629400	35.006.200	36
37	53629500	,LABELATT = LEB[9]#		53629500	35.006.200	37
38	53629550	,LABATT(LEB) = LEB[9]#		53629550	35.006.200	38
39	53629600	,FILEACCESS = LEB[10]#		53629600	35.006.200	39
40	53629650	,FILEACC(LEB) = LEB[10]#		53629650	35.006.200	40
41	53629700	,FIRSTSN = LEB[11]#		53629700	35.006.200	41
42	53629750	,FOREIGNINFO = LEB[12]#		53629750	35.006.200	42
43	53629780	,FAILUREREASON = LEB[13]#		53629780	32.122.892	43
44	53629790	,LEBMISC = LEB[14]#		53629790	32.142.970	44
45	53629818	,LEBVALINX = 15#,	% WORD OFFSET FROM THE BASE OF A LEB	53629818	340511164	45
46	53629820		% WHERE THE ATTRIBUTE VALIDITY BITS	53629820	340511164	46
47	53629822		% ARE LOCATED. THE ATTRIBUTE VALIDITY	53629822	340511164	47
48	53629824		% BITS OCCUPY WORDS 15 - 20 OF A LEB.	53629824	340511164	48
49	53629826			53629826	340511164	49
50	53629828	LEBVALIDITY(A) =		53629828	340511164	50
51	53629830	BOOLEAN(LEB[LEBVALINX + (A) DIV BITS PER WORD]).		53629830	340511164	51
52	53629832	(((BITS_PER_WORD - 1) - ((A) MOD BITS_PER_WORD)):1)]#,		53629832	340511164	52
53	53629834	% SELECTS THE ATTRIBUTE VALIDITY BIT		53629834	340511164	53
54	53629836	% FROM THE LEB VALIDITY MASK ARRAY		53629836	340511164	54
55	53629838	% WHICH CORRESPONDS TO THE ATTRIBUTE		53629838	340511164	55
56	53629840	% IDENTIFIED BY THE ATTRIBUTE NUMBER		53629840	340511164	56
57	53629842	% A. BIT 47 OF WORD ZERO IS THE		53629842	340511164	57
58	53629844	% VALIDITY BIT FOR FILE ATTRIBUTE 0,		53629844	340511164	58
59	53629846	% BIT 0 OF WORD ZERO IS THE VALIDITY		53629846	340511164	59
60	53629848	% BIT FOR FILE ATTRIBUTE 47.		53629848	340511164	60

1	53629850			53629850	340511164	1
2	53629852	LEBVALIDITYLENGTH = BITSTOWORDS(256)#		53629852	340511164	2
3	53629854		% SIZE IN WORDS OF THE SECTION OF A LEB	53629854	340511164	3
4	53629856		% USED TO HOLD THE FILE ATTRIBUTE	53629856	340511164	4
5	53629858		% VALIDITY BITS. AT LEAST 256 BITS	53629858	340511164	5
6	53629860		% ARE NEEDED TO HAVE A VALIDITY BIT	53629860	340511164	6
7	53629862		% FOR EVERY POSSIBLE FILE ATTRIBUTE	53629862	340511164	7
8	53629864		% NUMBER.	53629864	340511164	8
9	53629866	,LEBVAL = LEB[LEBVALINX]#	% FOR ATTRIBUTE VALIDITY BITS	53629866	321761190	9
10	53629867	,TIMELIMIT = LEB[21]#	% TIMELIMIT ATTRIBUTE FOR REMOTE FILES.	53629867	351042479	10
11	53629868	% NEXT AVAILABLE INDEX = 22		53629868	351042479	11
12	53629870	,FIXEDLEBSZ= 22 #	% CAN'T BE > 31 (SEE FXD)	53629870	351042479	12
13	53629874	,VARIABLELEBSZ(TITLESZ) =	% ESTIMATE FOR LEB ALLOCATION	53629874	35.028.845	13
14	53629876		% IF TITLE SIZE IS UNKNOWN, PASS 0 TO GET DEFAULT	53629876	35.028.845	14
15	53629878	(		53629878	35.028.845	15
16	53629880	5	% FOR FAMILY NAME (20 CHAR)	53629880	350341009	16
17	53629882	+ 2	% FOR KINDLIST (1 WORD)	53629882	35.028.845	17
18	53629884	+ 5	% FOR INTNAME (20 CHAR)	53629884	350341009	18
19	53629886	+ MAX(TITLESZ,8)%	% FOR TITLE (44 CHAR MIN)	53629886	35.028.845	19
20	53629890	) #		53629890	35.027.792	20
21	53629900	% LABEL TABLE		53629900	.	21
22	53630600	,PBRECORDCOUNT = 298 #	% NUMBER OF RECORDS IN THIS BLOCK	53630600	31.040.271	22
23	53630700	,PBNEXTRECORD = 299 #	% RECORD NUMBER OF THE NEXT	53630700	31.040.271	23
24	53630800		% RECORD IN THIS BLOCK	53630800	31.040.271	24
25	53630900	% DO NOT USE BIT 42 (LEBSB) BECAUSE OF O-LENGTH FIELD		53630900	321761190	25
26	53630950	,NULL_TYPE = 0#		53630950	35.028.834	26
27	53631000	,FRSTPNTR = 1#	% FIRST POINTER TYPE.	53631000	35.015.442	27
28	53631100	,INTNAME = 1#		53631100	35.015.442	28
29	53631200	,KINDLIST = 2#		53631200	35.015.442	29
30	53631300	,FRMS = 3#		53631300	35.015.442	30
31	53631400	,SECGUARD = 4#		53631400	35.015.442	31
32	53631500	,NAMEDPF = 5#	% NAMED PACK FILE.	53631500	35.015.442	32
33	53631600	,SERIALNO = 6#		53631600	35.015.442	33
34	53632000	,PRINTERCTL = 7#	% FILENAME OF NIP CONTROL FILE	53632000	35.015.442	34
35	53632020	,PRINTERCTL2 = 8#	% ACTUAL NAME OF OPENED CONTROL	53632020	35.015.442	35
36	53632040		% FILE (USED BY BACKUPLABELS)	53632040	35.015.442	36
37	53632060	,EIO LONGSTAT = 9#	% EIO LONG STATUS (UINFO ENTRIES)	53632060	35.015.442	37
38	53632080	% AVAILABLE = 10		53632080	35.015.442	38
39	53632100	,PSEUDONAME = 11#	% NAME CHECKPOINT CALLED IT.	53632100	35.015.442	39
40	53632200	,PSEUDOFAM = 12#	% FAMILYNAME AT CHECKPOINT TIME	53632200	35.015.442	40
41	53632300	% AVAILABLE = 13		53632300	35.015.442	41
42	53632400	% AVAILABLE = 14		53632400	35.015.442	42
43	53632500	,HOSTNAEM = 15#	% HOSTNAME (FILE ON ANOTHER HOST)	53632500	35.015.442	43
44	53632600	,ISAMKEYS = 16#	% INDEXED IO KEYS	53632600	35.015.442	44
45	53632800	,TITEL = 17#	% SPECIAL. ALSO MAKE SURE	53632800	35.015.442	45
46	53632820		% OPTMASK CAN FIT IN THE WORD	53632820	321761190	46
47	53632840		% OPTMASK IS IN IF THIS NUMBER IS	53632840	321761190	47
48	53632860		% CHANGED.	53632860	321761190	48
49	53632900	,LEBSB = 42#	% STARTING BIT OF OPTMASK	53632900	321761190	49
50	53633000	,LEEB(FLD) = LEB[LEBC.FXD+ONES(LEBC.[LEBSB:FLD])]#		53633000	.	50
51	53633050	,LEEB1(LEB,FLD) = LEB[LEBO(LEB).FXD + ONES(LEBO(LEB).[LEBSB:FLD])] #		53633050	35.015.442	51
52	53633100	,LEEB(FLD) = BOOLEAN(LEBC.[LEBSB-FLD:1])#		53633100	.	52
53	53633150	,LEEB1(LEB,FLD) = BOOLEAN(LEBO(LEB).[LEBSB-FLD:1])#		53633150	35.028.834	53
54	53633200	,LEBPTR8(FLD) = POINTER(LEB[LEEB(FLD)],8)#		53633200	31.092.398	54
55	53633250	,LEBPTR(LEB,FLD) = POINTER(LEB[LEEB1(LEB,FLD)],8) #		53633250	350411173	55
56	53633300	,FIDPTR(A) = POINTER(A[A[0].FXD+ONES(A[0].OPTMASK)],8)#		53633300	31.092.398	56
57	53633400	,FORMSPTR(A) = POINTER(A[A[A[0].FXD+ONES(A[0].[LEBSB:FRMS])]],8) #		53633400	31.092.398	57
58	53633500	,FORMS = LEBB(FRMS)#		53633500	.	58
59	53633600	,FORMSINDEX= LEBB(FRMS)#		53633600	.	59
60	53633700	,GUARDINFO = LEBB(SECGUARD)#		53633700	20.000.105	60

1	53633800	,GUARDINDEX= LEEB(SECGUARD)#	53633800	20.000.105	1
2	53633900	,NAMEINDEX = LEEB(NAMEDPF)#	53633900	22.000.083	2
3	53633905	,NAMEDB = LEEB(NAMEDPF)#	53633905	22.000.083	3
4	53633910	,KLUDGED1 = 0#	53633910	.	4
5					5
6					6
7					7
8					8
9					9
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
18					18
19					19
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
28					28
29					29
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1	53694100		% EXPLICITELY IN I/O STATEMENT.	53694100	26.010.131	1
2	53694200	% 4 = "I"	- VARIABLE LENGTH. SIZE IN RECORD	53694200	26.010.131	2
3	53694300		% AT FIXED LOCATION (SIZE ATTRIBUTES).	53694300	26.010.131	3
4	53694400	% 5 = "L"	- VARIABLE LENGTH. XALGOL LINKED.	53694400	26.010.131	4
5	53694500	% 6 = "Z"	- VARIABLE LENGTH. FORTRAN LINKED.	53694500	26.010.131	5
6	53694600	% 7 =	DEPENDANT SPECIFICATION. FIXED UP	53694600	26.010.131	6
7	53694700		% AT OPEN.	53694700	26.010.131	7
8	53694800	% 8 =	DEPENDANT SPECIFICATION. DONT CHANGE	53694800	26.010.131	8
9	53694900		% INTMODE.	53694900	26.010.131	9
10	53695000	, SIZEMODE	=[34:3]# % SIZEMODE ATTRIBUTE.	53695000	31.061.240	10
11	53695100	, SIZEOFF	=[31:16]# % SIZEOFFSET ATTRIBUTE.	53695100	31.061.240	11
12	53695200	, SIZESZ	=[15:16]# % SIZE2 ATTRIBUTE.	53695200	31.061.240	12
13	53695300		% TANKDATA2	53695300	26.010.131	13
14	53695400	, BLOCKSIZEF	=[47:16]# % PHYSICAL BLOCKSIZE	53695400	340521258	14
15	53695500	, MINRECSIZEF	=[31:16]# % PHYSICAL MINRECSIZE	53695500	350441252	15
16	53695600	, MAXRECSIZEF	=[15:16]# % PHYSICAL MAXRECSIZE	53695600	340521258	16
17	53695700		% DISKBLOCK	53695700	26.010.131	17
18	53695800	, SEGPBLKF	=[47:12]# % SEGMENTS PER BLOCK.	53695800	31.061.240	18
19	53695900	, RECPBLKF	=[35:16]# % RECORDS PER BLOCK.	53695900	34.009.255	19
20	53695915	, BINARYIPCF	=[19:3]# % CONTAINS SELECTOR.[2:3], BINARY IPC	53695915	34.009.255	20
21	53695925	, RANDOMIOF	=[16:1]# % IF ONLY RANDOM I/O IS TO BE DONE	53695925	34.009.255	21
22	53695935	, STACKSEARCHF	=[15:1]# % FOR PORTS	53695935	35.005.118	22
23	53695950	, PORTLINKDF	=[14:1]# % LINKED TO PORT LIBRARY	53695950	32.081.715	23
24	53696000	, DONTUNLOCKF	=[13:1]# % FOR MULTI-ENTRY FIBSTACK CALLS	53696000	31.141.240	24
25	53696100	, TANKINGF	=[12:2]# % TANKING VALUE	53696100	31.061.240	25
26	53696200	, NORESWAITF	=[10:1]# % DON-T WAIT FOR RESOURCES	53696200	31.061.240	26
27	53696300	, INPUTEVENTUSED	F=[9:1]# % 1 IF INPUTEVENT HAS BEEN USED	53696300	350721873	27
28	53696400	, MPS_USERF	=[8:1]# % FIB USED MPS OUTPUT ROUTER	53696400	350772066	28
29	53696500	, DUPLCOPIES	=[7:4]# % NUMBER OF COPIES MINUS ONE.	53696500	31.061.240	29
30	53696550	, ISAMLINKDF	=[3:1]# % LINKED TO KEYEDIO LIBRARY.	53696550	32.035.228	30
31	53696600	, UPDATEIOF	=[2:1]# % 1 IF DOING UPDATE I/O.	53696600	31.061.240	31
32	53696700	, UPDATEFILEF	=[1:1]# % 1 IF UPDATE I/O DISK FILE.	53696700	31.061.240	32
33	53696800	, DUPLFILEF	=[0:1]# % 1 IF DUPLICATED DISK FILE.	53696800	31.061.240	33
34	53696900		% PAGESPEC (PRINTER FILES LOGICAL PAGE SPECIFICATIONS)	53696900	26.010.131	34
35	53697000	, FIBCHOOZEF	=[47:12]#	53697000	31.061.240	35
36	53697100	, PGSPFCTNCODEF	=[47:4]# % CHOOZE.FUNCTIONCODEF.	53697100	31.061.240	36
37	53697200	, PGSPAFTERF	=[43:1]# % CHOOZE.AFTERF.	53697200	31.061.240	37
38	53697300	, PGSPACTIONF	=[42:6]# % CHOOZE.ACTIONF.	53697300	31.061.240	38
39	53697400	, PGSKEYEDF	=[36:1]# % CHOOZE.KEYEDF.	53697400	31.061.240	39
40	53697500	, BUFFERD	=[35:1]# % 1 IF MORE THAN ONE BUFFER.	53697500	31.061.240	40
41	53697600	, LTDLINEF	=[34:1]# % 1 IF MAXLINES COUNTING.	53697600	31.061.240	41
42	53697700	, LTPUNCHF	=[33:1]# % 1 IF MAXCARDS COUNTING.	53697700	31.061.240	42
43	53697800	, PAGESPECF	=[32:1]# % 1 IF LOGICAL PAGE COUNTING.	53697800	31.061.240	43
44	53697900	, PAGECOUNTF	=[31:16]# % PAGE ATTRIBUTE.	53697900	31.061.240	44
45	53698000	, PAGESIZEF	=[15:8]# % PAGESIZE ATTRIBUTE.	53698000	31.061.240	45
46	53698100	, LINECOUNTF	=[7:8]# % LINENUM ATTRIBUTE.	53698100	31.061.240	46
47	53698200		% IOINFO	53698200	26.010.131	47
48	53698300	, SKLTNIOCW	=[47:28]# % SKELETON IOCW (EXCEPT FOR DATACOM),	53698300	340521258	48
49	53698350		% SEE IOSTANDARDFIELD ETC.	53698350	340521258	49
50	53698400	, LASTSTATIONF	=[47:24]# % DATACOM LASTSTATION ATTRIBUTE.	53698400	31.061.240	50
51	53698850	%	=[23:4]# % AVAILABLE FOR DATACOM	53698850	340521258	51
52	53698900	, RDCHECKF	=[19:1]# % READCHECK ATTRIBUTE.	53698900	31.061.240	52
53	53699000	, BACKUPSWITCHF	=[18:1]# % 1 IF SIOINFO IN IOINFO SLOT.	53699000	31.061.240	53
54	53699100	, OVERROWF	=[17:1]# % 1 IF BADLY BLOCKED DISK I/O.	53699100	31.061.240	54
55	53699200	, BACKUPF	=[16:1]# % 1 IF CREATING BACKUP FILE.	53699200	31.061.240	55
56	53699300	%	, AVAILABLE = [15:10]# % FORMERLY UNITNUM LOCATION	53699300	35.027.799	56
57	53699400	, UNITYPP	=[5:6]# % ASSIGNED UNIT TYPE.	53699400	340521258	57
58	53699500		% FIBEOF (DISK EOF CALCUALTIONS).	53699500	26.010.131	58
59	53699600	% EOFU	=[47:20]# % # OF VALID UNITS IN LAST BLOCK.	53699600	31.061.240	59
60	53699700	% EOFV	=[27:28]# % SEGMENT WHERE LAST BLOCK BEGINS	53699700	31.061.240	60

1	53699800	% ACTNUM	(MAINTAINED BY IODISKADDRESS).	53699800	26.010.131	1
2	53699850	, ROWINVALIDF	= [47:1]# % (IN)VALIDITY BIT.	53699850	350541499	2
3	53699900	, ROWUNITNUMF	= [46:LUB(14)]# % LOGICAL UNIT NUMBER.	53699900	350541499	3
4	53700100	, ROWBASEADDRESSF	= [32:33]# % DISK ADDRESS IN BINARY.	53700100	350541499	4
5	53700200	% FLOPPYMISC		53700200	31.031.208	5
6	53700300	, NORMALIOLENGTHF	= [47:20]# % NORMAL IO AREADESC LENGTH	53700300	331121095	6
7	53700400	, CUREXTENTF	= [27:20]# % TOTAL EXTENT USED	53700400	31.061.240	7
8	53700500	, CHARSPERSECTORF	= [7:8]# % FOR FLOPPY TEST OP	53700500	331121095	8
9	53700510	% HCPARTNERMISC	% = FLOPPYMISC	53700510	340581503	9
10	53700520	% WRITEPARTNER	= [11:04] % SAME HCFILES IOPARTNER FIELDS AS	53700520	340581503	10
11	53700530	% READPARTNER	= [07:04] % THOSE IN IOCD[DIRECTIOCW], BUT	53700530	340581503	11
12	53700540	% READPARTNERVALID	= [02:01] % HOLDS HUBMAP-LEVEL RESTRICTIONS;	53700540	340581503	12
13	53700550	% WRITEPARTNERVALID	= [01:01] % PARTNERMISC IS SET-UP IN FIBOPEN.	53700550	340581503	13
14	53700556	, % BUFLINKS	- THIS FIELD CONTAINS THE NUMBER OF BUFFERS AND	53700556	35.009.306	14
15	53700558	%	CURRENT "TOP" BUFFER FOR THE IOMOM & SIOMOM	53700558	35.009.306	15
16	53700560	%	BUFFER POOLS. THE BOTTOM 24 BITS ARE USED BY	53700560	35.009.306	16
17	53700562	%	IOMOM, TOP 24 BY SIOMOM. IT IS VERY IMPORTANT TO	53700562	35.009.306	17
18	53700564	%	NOTE THAT THE FIELDS FOR IOMOM AND SIOMOM ARE	53700564	35.009.306	18
19	53700566	%	IDENTICAL IN FORMAT. THIS IS NECESSARY BECAUSE	53700566	35.009.306	19
20	53700568	%	THE TWO HALVES OF THE WORD ARE SWAPPED WHENEVER	53700568	35.009.306	20
21	53700570	%	IOMOM AND SIOMOM ARE SWAPPED (DONE FOR BACKUP	53700570	35.009.306	21
22	53700572	%	FILES)	53700572	35.009.306	22
23	53700576	, BACKUP_FLAG_FIELD	= [47:01]# % USED TO MARK THE SIO AREA	53700576	35.028.847	23
24	53700578		% OF THE BUFFER - THIS HELPS	53700578	35.028.847	24
25	53700580		% WHEN ANALYZING DUMPS	53700580	35.028.847	25
26	53700582	, BUFFER_CONTROL_FIELD	= [23:48]# % FIELD USED WHEN SWAPPING	53700582	35.009.306	26
27	53700584		% IOMOM AND SIOMOM	53700584	35.009.306	27
28	53700586	, CURRENT_BUFFER_FIELD	= [07:08]# % # OF TOP BUFFER	53700586	35.009.306	28
29	53700588	, POPULATION_FIELD	= [15:08]# % # OF BUFFS IN POOL	53700588	35.009.306	29
30	53700590	, SIOMOM_POPULATION_FIELD		53700590	35.009.306	30
31	53700592	%	= [39:08]# % # OF BUFFS IN SIOMOM POOL	53700592	35.009.306	31
32	53700594	%	----- END FIB FIELDS -----	53700594	340521258	32
33	53700595			53700595	35.028.847	33
34	53700596	% UNITNUMBERS	- CONTAINS THE LOGICAL UNIT NUMBER FOR THIS FILE	53700596	350431237	34
35	53700598	%	(EXCEPT FOR DISK FILES WHERE IT CONTAINS THE	53700598	35.027.799	35
36	53700600	%	DISKFILEHEADER INDEX). "LUORHDRBITS" OF THE BOTTOM	53700600	350431237	36
37	53700602	%	24 BITS OF THE WORD CONTAIN THE FIELD WHICH IS	53700602	35.027.799	37
38	53700604	%	IS REFERENCED TO OBTAIN THE UNIT NUMBER.	53700604	35.027.799	38
39	53700606	%	WHEN IOINFO AND SIOINFO ARE SWAPPED, THIS WORD IS	53700606	35.027.799	39
40	53700608	%	ROTATED (SIMILARLY TO BUFLINKS) AND THE SIO	53700608	35.027.799	40
41	53700610	%	UNIT NUMBER IS RETURNED WHEN UNITORHEADERINDEXF	53700610	35.027.799	41
42	53700612	%	IS ACCESSED. RE-ROTATION WOULD RETURN THE UNIT	53700612	35.027.799	42
43	53700614	%	NUMBER SAVED IN THE TOP HALF OF THE WORD TO ITS	53700614	35.027.799	43
44	53700616	%	ORIGINAL POSITION IN THE BOTTOM HALF OF THE WORD.	53700616	35.027.799	44
45	53700617	, LUORHDRBITS	= MAX(HEADERBITS,LUBMIN)#	53700617	350431237	45
46	53700618	% AVAILABLE	= [47:(24-LUORHDRBITS)]	53700618	350431237	46
47	53700619	, SIO_UNITORHEADERINDEXF	= [39:LUORHDRBITS]#	53700619	350431237	47
48	53700620	% AVAILABLE	= [23:(24-LUORHDRBITS)]	53700620	350431237	48
49	53700621	, UNITORHEADERINDEXF	= [15:LUORHDRBITS]#	53700621	350431237	49
50	53700623	%	----- END FIB FIELDS -----	53700623	35.027.799	50
51	53700624			53700624	35.027.799	51
52	53700626			53700626	35.027.799	52
53	53700628			53700628	35.027.799	53
54	53700640	%	----- LEB FIELDS -----	53700640	340521258	54
55	53700660			53700660	340521258	55
56	53700700	, FXD	= [47:5]# % FIXED LENGTH OF LEB.	53700700	340521258	56
57	53700800	, OPTMASK	= [LEBSB:TITEL]#% VARIABLE LENGTH PARTS, ALSO USES	53700800	340521258	57
58	53700850		% NEXT BIT, SO USED BITS ARE [42:18]	53700850	321761190	58
59	53700900	% (CANNOT USE)	= 42:1,	53700900	321761190	59
60	53701000	% (INTNAME)	= 41:1,	53701000	321761190	60

1	53701100	% (KINDLIST) = 40:1,	53701100	321761190	1
2	53701200	% (FORMMESSAGE) = 39:1,	53701200	321761190	2
3	53701300	% (GUARDFILE) = 38:1,	53701300	321761190	3
4	53701400	% (PACKNAME) = 37:1,	53701400	321761190	4
5	53701500	% (SERIALNO) = 36:1,	53701500	321761190	5
6	53701600	% (PRINTERCTL) = 35:1,	53701600	35.015.442	6
7	53701700	% (PRINTERCTL2) = 34:1,	53701700	35.015.442	7
8	53701800	% (EIDLONGSTAT) = 33:1,	53701800	35.015.442	8
9	53701900	% AVAILABLE = 32:1,	53701900	35.015.442	9
10	53702000	% (PSEUDONAME) = 31:1,	53702000	321761190	10
11	53702100	% (PSEUDOFAM) = 30:1,	53702100	321761190	11
12	53702200	% AVAILABLE = 29:1,	53702200	35.015.442	12
13	53702300	% AVAILABLE = 28:1,	53702300	35.015.442	13
14	53702400	% (HOSTNAEM) = 27:1,	53702400	321761190	14
15	53702500	% (ISAMKEYS) = 26:1,	53702500	321761190	15
16	53702700	% (TITEL) = 25:1, % NOT IN OPTMASK, BUT USED	53702700	321761190	16
17	53702800	, OPTIONALF = [24:1]# % OPTIONAL ATTRIBUTE	53702800	31.078.398	17
18	53702900	, BYPASSF = [23:1]# % FILE BYPASS INDICATOR	53702900	31.078.398	18
19	53703000	, LEOFFIBF = [22:1]# % TRUE IF THERE IS A FIB	53703000	31.078.398	19
20	53703100	, FPRTCTD = [21:1]#	53703100	31.078.398	20
21	53703200	, LPARITY = [20:1]# % PARITY (1 IS NON-STANDARD).	53703200	340521258	21
22	53703300	, DENSITYQ = [19:3]# % VALID IF GTR 0	53703300	31.155.541	22
23	53703400	% VALUE OF ATTRIBUTE (GIVEN	53703400	31.155.541	23
24	53703500	% BELOW) IS LEBC.DENSITYQ-1.	53703500	31.155.541	24
25	53703600	% 0 = HIGH (800).	53703600	26.010.131	25
26	53703700	% 1 = MEDIUM (556).	53703700	26.010.131	26
27	53703800	% 2 = LOW (200).	53703800	26.010.131	27
28	53703900	% 3 = SUPER (1600).	53703900	26.010.131	28
29	53704000	% 4 = (6250)	53704000	31.155.541	29
30	53704100	, LABELYPES = [16:1]# % 1 IF LABELTYPE IS VALID.	53704100	340521258	30
31	53704200	, LABELYPEL = [15:4]# % LABELTYPE ATTRIBUTE.	53704200	340521258	31
32	53704300	% LEBC LCNTRL	53704300	26.010.131	32
33	53704400	% 0 = STANDARD. 0 = B6500 USASI.	53704400	26.010.131	33
34	53704500	% 1 = OMITTED. 1 = UNLABELED.	53704500	26.010.131	34
35	53704600	% 2 = B5500 STANDARD.	53704600	26.010.131	35
36	53704700	% 3 = OMITTEDEOF. 3 = SYSTEM STANDARD.	53704700	26.010.131	36
37	53704800	% 4 = B3500 USASI.	53704800	26.010.131	37
38	53704900	% 5 = B3500 STANDARD.	53704900	26.010.131	38
39	53705000	% 6 = BACKUP.	53705000	26.010.131	39
40	53705100	% 7 = LIBRARY MAINTENANCE.	53705100	26.010.131	40
41	53705200	% 8 = LOAD CONTROL.	53705200	26.010.131	41
42	53705300	% 10 = B5500 BACKUP.	53705300	26.010.131	42
43	53705400	% 11 = B5500 LIBRARY.	53705400	26.010.131	43
44	53705500	% 12 = NON-STANDARD USASI.	53705500	26.010.131	44
45	53705600	% 13 = USERDEFINED.	53705600	26.010.131	45
46	53705700	, EXTVALID = [11:1]# % 1 IF EXTMODE IS SPECIFIED.	53705700	31.061.240	46
47	53705800	, EXTFORML = [10:3]# % EXTMODE ATTRIBUTE.	53705800	31.061.240	47
48	53705900	, PTYPEL = [7:8]#	53705900	31.061.240	48
49	53706000	, PTYPECODEF = [7:3]#	53706000	31.061.240	49
50	53706100	, PTYPEF = [4:5]#	53706100	31.061.240	50
51	53706200	% GEN1 (LEB WORD)	53706200	26.010.131	51
52	53706300	, LABLEV = [47:9]# % LABEL LEVEL.	53706300	31.061.240	52
53	53706400	, CYCLES = [38:1]# % 1 IF GENEALOGY IS SPECIFIED.	53706400	31.061.240	53
54	53706500	, GENEALOGYF = [37:22]# % (CYCLE + VERSION).	53706500	31.061.240	54
55	53706600	, CYCLE = [37:14]# % CYCLE ATTRIBUTE.	53706600	31.061.240	55
56	53706700	, GENVERSN = [23:8]# % VERSION ATTRIBUTE.	53706700	31.061.240	56
57	53706800	, REELS = [15:1]# % 1 IF REEL NUMBER IS SPECIFIED.	53706800	31.061.240	57
58	53706900	, REEL = [14:15]# % REEL ATTRIBUTE.	53706900	31.061.240	58
59	53707000	% GEN2 (LEB WORD)	53707000	26.010.131	59
60	53707050	% = [47:2]# % AVAILABLE	53707050	35.011.316	60

1	53707100	, CRTNSITEIDF = [45:14]#	% SYSTEM SERIAL NUMBER.	53707100	35.011.316	1
2	53707200	, GENERATIONF = [31:4]#	% GENERATION ATTRIBUTE.	53707200	31.061.240	2
3	53707300	, SAVFACTOR = [27:11]#	% SAVEFACTOR ATTRIBUTE.	53707300	31.061.240	3
4	53707400	, CRTNDATE = [16:17]#	% CREATION DATE.	53707400	31.061.240	4
5	53707500	, % DSKS (LEB WORD)		53707500	26.010.131	5
6	53707600	, DIRECTIONF = [47:1]#	% DIRECTION ATTRIBUTE.	53707600	31.061.240	6
7	53707700	, SINGLEPACKF = [46:1]#	% SINGLEPACK ATTRIBUTE.	53707700	31.078.398	7
8	53707800	, USECATALOGF = [45:1]#	% USECATALOG ATTRIBUTE.	53707800	31.061.240	8
9	53707900	, SENSITIVEF = [44:1]#	% SENSITIVEDATA ATTRIBUTE.	53707900	31.061.240	9
10	53708000	, DRCTRYF = [43:1]#	% 1 IF GOES INTO DIRECTORY.	53708000	31.061.240	10
11	53708100	, CYLMODEF = [42:1]#	% CYLINDERMODE ATTRIBUTE.	53708100	31.078.398	11
12	53708200	, FLEXIBLEF = [41:1]#	% FLEXIBLE ATTRIBUTE.	53708200	31.061.240	12
13	53708300	, SIZEVISBIASF = [40:3]#	% SIZEVISIBLE SIZE BIAS	53708300	350942344	13
14	53708400	, OTHERUSEF = [37:2]#		53708400	31.061.240	14
15	53708500	, MYUSEF = [35:2]#	% MYUSE ATTRIBUTE.	53708500	31.061.240	15
16	53708600	, EXCLUSIVEF = [33:1]#	% EXCLUSIVE ATTRIBUTE.	53708600	31.078.398	16
17	53708700	, INTERCHGEF = [32:1]#	% INTERCHANGE ATTRIBUTE.	53708700	31.078.398	17
18	53708800	, FILEUSEF = [31:2]#	% FILEUSE ATTRIBUTE.	53708800	31.092.424	18
19	53708900	, NUMAREAS = [29:10]#	% AREAS ATTRIBUTE.	53708900	31.061.240	19
20	53709000	, AREASIZE = [19:20]#	% AREASIZE ATTRIBUTE.	53709000	31.061.240	20
21	53709100	, % DISKMISC (LEB WORD)		53709100	26.010.131	21
22	53709200	, LFILEORGANIZF = [47:3]#	% FILEORGANIZATION ATTRIBUTE	53709200	350391118	22
23	53709400	, LEXTENTF = [44:17]#	% EXTENT OF DISKETTE FILE	53709400	32.141.948	23
24	53709500	, LEXCLUSIVECANF = [27:1]#	% FOR RESERVE	53709500	31.061.240	24
25	53709600	, LUPDATEIOVF = [26:1]#	% 1 IF UPDATEFILE IS VALID.	53709600	31.061.240	25
26	53709700	, LUPDATEIOF = [25:1]#	% UPDATEFILE ATTRIBUTE.	53709700	31.061.240	26
27	53709800	, LDUPLFILEF = [24:1]#	% DUPLICATED ATTRIBUTE.	53709800	31.061.240	27
28	53709900	, LFAMILYINDEX = [23:8]#	% FAMILYINDEX ATTRIBUTE.	53709900	31.061.240	28
29	53709950	, LWRITECHECKF = [15:1]#	% WRITECHECK ATTRIBUTE	53709950	350391118	29
30	53710000	, % AVAILABLE = [14:7]#	% AVAILABLE	53710000	350391118	30
31	53710100	, LFILEKIND = [7:8]#	% FILEKIND ATTRIBUTE.	53710100	31.061.240	31
32	53710200	, % TANKDATA3 (LEB WORD)		53710200	26.010.131	32
33	53710300	, BUFFREQSTD = [47:8]#	% BUFFERS ATTRIBUTE.	53710300	31.061.240	33
34	53710350	, BKUPKINDF = [39:6]#	% ACTUAL KIND FOR PRINTER/PUNCH	53710350	31.061.240	34
35	53710400	, % **NOT USED** = [33:1]#	% **** AVAILABLE ****.	53710400	350571579	35
36	53710450	, NEWFILEF = [32:1]#	% NEWFILE ATTRIBUTE.	53710450	350571579	36
37	53710500	, BLANKF = [31:1]#	% BLANK ATTRIBUTE.	53710500	34.006.198	37
38	53710550	, % **NOT USED** = [30:1]#	% **** AVAILABLE ****.	53710550	350571579	38
39	53710600	, COBOL74LOCKF = [29:1]#	% CLOSED WITH COBOL74 LOCK	53710600	32.048.381	39
40	53710650	, OPT = [28:1]#	% 1 IF NON-PRESENT OPTIONAL FILE.	53710650	31.061.240	40
41	53710700	, ATTERRF = [27:1]#	% ATTERR ATTRIBUTE.	53710700	31.061.240	41
42	53710750	, ATTYPEF = [26:8]#	% ATTYPE ATTRIBUTE.	53710750	32.141.948	42
43	53710800	, SIZE_INVISF = [18:1]#	% SIZE FIELD IS CURRENTLY INVISIBLE	53710800	350571579	43
44	53710850		% ~(VALID ONLY AFTER SETUPFIB)	53710850	350571579	44
45	53710900	, SIZEVISIBLEF = [17:1]#	% SIZEVISIBLE ATTRIBUTE.	53710900	350571579	45
46	53710950	, % **NOT USED** = [16:1]#	% **** AVAILABLE ****.	53710950	351122546	46
47	53710960	, % INDICATES USER EXPLICITLY	DECLARES TRANSLATE MODE	53710960	350762030	47
48	53711000	, SOFTTRANSF = [15:1]#	% TRANSLATING ATTRIBUTE.	53711000	31.061.240	48
49	53711050	, TRANSLATEF = [14:3]#	% TRANSLATE ATTRIBUTE.	53711050	31.061.240	49
50	53711100	, CRIGECONTRLF = [11:2]#	% CARRIAGECONTROL ATTRIBUTE.	53711100	31.061.240	50
51	53711150	, CONTROL360F = [11:1]#	% IBM 1403 PRINTER CODE.	53711150	32.142.970	51
52	53711200	, CONTROLASAF = [10:1]#	% FORTRAN PRINTER CODE.	53711200	32.142.970	52
53	53711250	, CHECKPOINTF = [8:1]#	% 1 IF ENTERED UNDER CP DIRECTORY	53711250	31.061.240	53
54	53711300	, DEFKINDF = [7:1]#	% 1 IF KIND WAS ZERO.	53711300	31.061.240	54
55	53711350	, EOFDEF = [6:1]#	% 1 IF GIVEN ONE EOF ERROR MSG.	53711350	31.061.240	55
56	53711400	, %*AVAILABLE** = [5:2]#	%	53711400	33.101.837	56
57	53711450	, BLOCKSTRUCTV = [3:1]#	% BLOCKSTRUCTURE VALIDITY BIT	53711450	32.142.970	57
58	53711500	, BLOCKSTRUCTF = [2:3]#	% BLOCKSTRUCTURE ATTRIBUTE	53711500	32.142.970	58
59	53712000	, % LABELATT (LEB WORD)		53712000	26.010.131	59
60	53712050	, USERSETFAMILYNAMEF = [47:1]#	% TRUE IFF THE USER SET THE FAMILYNAME	53712050	331521321	60

1	53712100	, LABELTYP	= [46:3]#	% ASSIGNED LABEL TYPE.	53712100	31.061.240	1
2	53712200		%	0 = USASI LABEL.	53712200	26.010.131	2
3	53712300		%	1 = COBOL UNLABELED (REELSWITCH).	53712300	26.010.131	3
4	53712400		%	2 = NO TAPEMARK BETWEEN LABELS.	53712400	26.010.131	4
5	53712500		%	3 = UNLABELED MULTI/FILE REELS (EOF).	53712500	26.010.131	5
6	53712550	, ACTUAL PRINTERKINDF	= [43:4]#	% ACTUAL PRINTERKIND SET BY NIP	53712550	35.006.200	6
7	53712600	, SECTYPEBITF	= [37:1]#	% 1 IF SECTYPE EXPLICITELY SET.	53712600	31.061.240	7
8	53712700	, SECUSEBITF	= [36:1]#	% 1 IF SECUSE EXPLICITELY SET.	53712700	31.061.240	8
9	53712800	, SECTYPEF	= [35:2]#	% SECURITYTYPE ATTRIBUTE.	53712800	31.061.240	9
10	53712900	, SECUSEF	= [33:2]#	% SECURITYUSE ATTRIBUTE.	53712900	31.061.240	10
11	53712950	%	= [31:3]#	% AVAILABLE	53712950	331121095	11
12	53713000	, USEINDEXF	= [28:8]#	% USERROUTINES INDEX.	53713000	31.061.240	12
13	53713100	, LABELUSEF	= [20:1]#	% 1 IF LABEL USERROUTINES PRESENT.	53713100	31.061.240	13
14	53713200	, ERRUSEF	= [19:1]#	% 1 IF ERROR USERROUTINES PRESENT.	53713200	31.061.240	14
15	53713250	%	= [18:2]#	% AVAILABLE	53713250	331121095	15
16	53713300	, USERSIZE	= [16:4]#	% NUMBER OF USER LABELS.	53713300	31.061.240	16
17	53713400	, LBLEQTD	= [12:1]#	% 1 WHEN FINISHED LABEL EQUATION.	53713400	31.061.240	17
18	53713410	%	= [11:4]#	% AVAILABLE	53713410	331121095	18
19	53713420	, FRF	= [7:8] #	% FILE REGULATION FIELD =SECURITY INFO	53713420	33.101.837	19
20	53713440			% MAINLY USED FOR TEMPORARY FILES	53713440	33.101.837	20
21	53713460			% SAVED BY CHECKPOINT	53713460	33.101.837	21
22	53713500	% FILEACCESS	(LEB WORD)		53713500	26.010.131	22
23	53713550	, VOLUMEDMTBITF	= [47:1]#	% CONSISTANT VOLUMED/UNVOLUMED	53713550	30.7X.685	23
24	53713600	, CREATEFILEF	= [46:1]#	% FILE WAS CREATED IN OPEN	53713600	331121095	24
25	53713700	, PRINTERCTL_OKF	= [45:1]#	% CONTROL FILE IS OK TO USE	53713700	35.015.442	25
26	53713800	, IMPLICIT_OPENF	= [44:1]#	% FILE WAS IMPLICITLY OPENED ON WRITE	53713800	35.032.944	26
27	53713910	, INUSEROUTINEF	= [43:1]#	% IN COBOL USE ROUTINE	53713910	29.7X.454	27
28	53713920	, APPLICATIONF	= [42:1]#	% FILE PROPAGATES MA	53713920	32.035.221	28
29	53713950	, INDEXEDMODEF	= [41:2]#		53713950	32.025.140	29
30	53713980	, UNVOLUMEDMTBITF	= [39:1]#	% INDICATES IF 1ST REEL IS UNVOLUMED	53713980	331051020	30
31	53713990	, POSITIONATENDF	= [38:1]#	% POSITION FILE AT END AT END OF OPEN	53713990	331061044	31
32	53713995	, USER_PRINTERKINDF	= [37:4]#	% INDICATES THE TYPE OF PRINTER	53713995	35.006.200	32
33	53714020			% 0 DONT CARE	53714020	35.006.200	33
34	53714040			% 1 IPONLY NIP PRINTER ONLY	53714040	35.006.200	34
35	53714060			% 2 LPONLY LINEPRINTER ONLY	53714060	35.006.200	35
36	53714080	, AUDITF	= [33:2]#	% RESERVE FOR B7700	53714080	31.061.240	36
37	53714100	, TRAINIDF	= [31:8]#	% TRAINID ATTRIBUTE.	53714100	31.061.240	37
38	53714200	, PCWERRF	= [23:8]#	% IF THE CORRESPONDING PCW	53714200	35.015.440	38
39	53714300	, PWRITESERRF	= [23:1]#	% HAS "BAD" IN IT, THESE FIELDS	53714300	35.015.440	39
40	53714400	, PREADSERRF	= [22:1]#	% INDICATE THE REASON.	53714400	35.015.440	40
41	53714500	, PWRITENERRF	= [21:1]#	%	53714500	35.015.440	41
42	53714600	, PREADNERRF	= [20:1]#	%	53714600	35.015.440	42
43	53714700	, PSEEKERRF	= [19:1]#	%	53714700	35.015.440	43
44	53714800	%	= [18:3]#	% AVAILABLE PCW ERROR FIELDS	53714800	35.015.440	44
45	53714810	%	= [15:16]#	% AVAILABLE	53714810	35.015.440	45
46	53714820	% FOREIGNINFO	(LEB WORD)	(HOST SERVICES INFO)	53714820	31.170.670	46
47	53714830	%	[47:16]	% AVAILABLE	53714830	32.081.715	47
48	53714835	, MAXSUBFILESF	= [31:16]#	% MAX SUB PORT FILES ALLOWED	53714835	32.081.715	48
49	53714840	, DIALOGNUMBERF	= [15:08]#	% DIALOG NUMBER FROM HOST SERVICES	53714840	31.170.670	49
50	53714850	%	= [07:05]	% AVAILABLE	53714850	331121095	50
51	53714855	, FILEXFERFILEF	= [02:01]#	% THIS FILE IN USE BY HS FILE XFER	53714855	31.187.766	51
52	53714860	, FCFILEF	= [01:01]#	% FILECOOP'S FILE FOR ANOTHER HOST	53714860	31.170.670	52
53	53714870	, FOREIGNFILEF	= [00:01]#	% THIS FILE IS ON A FOREIGN HOST	53714870	31.170.670	53
54	53714875	% FAILUREREASON	(LEB WORD)	(OPEN/CLOSE FAILURE REASON)	53714875	32.122.892	54
55	53714880	, FAILFATALF	= [47:01]#	% CALL MCPHANDLEERROR	53714880	340441097	55
56	53714882	, FAILWHICHSIGNF	= [46:01]#	% SIGN OF FAILWHICHF	53714882	340441097	56
57	53714884	, FAILPRINTDF	= [45:01]#	% 1 => MESSER CALLED FOR THIS MESSAGE	53714884	340441097	57
58	53714885	, MYOFFSETF	= [44:16]#	% STACK RELATIVE OFFSET OF FILE	53714885	35.015.449	58
59	53714890			% (FOR USER LABELS)	53714890	35.015.449	59
60	53714894	%	= [28: 5]#	% AVAILABLE	53714894	35.015.449	60

1	53714896	, FAILWHATF	= [23:12]#	% CLOSEERRORVV OR OPENERORVV	53714896	340441097	1							
2	53714898	, FAILWHICHF	= [11:12]#	% MESSER ALREADY CALLED FOR THIS WHICH	53714898	340441097	2							
3	53714900	% LEBMISC	(LEB WORD)	(MISCELLANEOUS INFORMATION)	53714900	32.142.970	3							
4	53714905	% UNUSED BIT	[47:01]		53714905	32.142.970	4							
5	53714910	, FRAMESIZEV	= [46:01]#	% FRAMESIZE VALIDITY BIT	53714910	32.142.970	5							
6	53714915	, FRAMESIZEF	= [45:06]#	% FRAMESIZE VALUE	53714915	32.142.970	6							
7	53714920	, DEPENDENTSPECS	= [39:01]#	% DEPENDENTSPECS HAVE BEEN APPLIED	53714920	350571579	7							
8	53714925	, DEPENDENTSPECSF	= [38:01]#	% DEPENDENTSPECS VALUE	53714925	32.142.970	8							
9	53714930	% UNUSED BITS	[37:05]		53714930	32.177.970	9							
10	53714935	, AREALENGTHV	= [32:01]#	% AREALENGTH VALIDITY BIT	53714935	32.177.970	10							
11	53714940	, AREALENGTHF	= [31:32]#	% AREALENGTH VALUE	53714940	32.177.970	11							
12	53714960				53714960	340521258	12							
13	53714970	% -----	END	LEB	F	I	E	L	D	S	-----	53714970	340521258	13
14														14
15														15
16														16
17														17
18														18
19														19
20														20
21														21
22														22
23														23
24														24
25														25
26														26
27														27
28														28
29														29
30														30
31														31
32														32
33														33
34														34
35														35
36														36
37														37
38														38
39														39
40														40
41														41
42														42
43														43
44														44
45														45
46														46
47														47
48														48
49														49
50														50
51														51
52														52
53														53
54														54
55														55
56														56
57														57
58														58
59														59
60														60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	D I S K F I L E H E A D E R	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	A P P E N D I X G	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1	WORKFILE [NOT NAMED]	5:04 PM TUESDAY, APRIL 9, 1985	1
2			2
3			3
4	37660626 DEFINE	37660626 350922267	4
5	37660628 % WORD ZERO OF HEADER	37660628 350922267	5
6	37660630 MARKERF = [47:16]#	37660630 321721099	6
7	37660635 ,INUSEMARK = 4'3F3F'#	37660635 321721099	7
8	37660640 ,AVAILMARK = 4'3C3C'#	37660640 321721099	8
9	37660645 ,BADAREAMARK = 4'3A3A'#	37660645 321721099	9
10	37660650 ,MCPUSEMARK = 4'3939'#	37660650 321721099	10
11	37660655 ,PRESUSEMARK = 4'3838'# % PRESERVER USED AREA	37660655 321721099	11
12	37660656 ,HDROBUF = [47:LUB(16)]# % SAME AS MARKERF, BUT HDRO USES	37660656 350711836	12
13	37660658 % FIELD TO HOLD FILE'S BASEUNIT.	37660658 350711836	13
14	37660660 ,HDRBLOCKLENGTHF = [31:11]# % TOTAL HEADER SIZE IN WORDS	37660660 350922267	14
15	37660662 ,MAXHEADERSIZEV = (2 ** 11)# % SIZE OF LARGEST POSSIBLE HDR	37660662 350922267	15
16	37660665 ,HDRLOCATIONF = [20:21]#	37660665 321721099	16
17	37660670 ,AVAILFLAG(REC) = ((O & AVAILMARK MARKERF	37660670 321721099	17
18	37660675 & 1 HDRBLOCKLENGTHF) & (REC) HDRLOCATIONF)#	37660675 321721099	18
19	37660680 ,INUSEFLAG(L,R) = (O & INUSEMARK MARKERF & (L) HDRBLOCKLENGTHF	37660680 321721099	19
20	37660685 & (R) HDRLOCATIONF)#	37660685 321721099	20
21	37660700 % HEADERINFOONE	37660700 31.061.240	21
22	37660800 % WORD 1	37660800 31.061.240	22
23	37660900 , UPDATEBITF = [47:1]# %TRUE IF FILE UPDATED	37660900 321721099	23
24	37661000 , OPENCOUNTF = [46:11]# % NUMBER OF PROCESSES %LOOKING AT TH	37661000 31.061.240	24
25	37661050 , AVAILOF = [35:04]# % AVAILABLE, ZEROED BY HEADERKEEPER	37661050 34.028.677	25
26	37661100 , FILEKINDF = [31:8]# % DATA, CODE, ETC	37661100 31.061.240	26
27	37661200 , HEADERSIZEF = [23:16]# % SIZE IN WORDS	37661200 31.061.240	27
28	37661300 , SECINFOSIZEF = [07:08]# % SIZE IN WORDS OF THE % SECURITY INF	37661300 31.061.240	28
29	37661500 % HEADERTWO	37661500 31.061.240	29
30	37661600 %FILE ORGANIZATION	37661600 31.061.240	30
31	37661700 , IADBIT = [47:1]# % 1 IF IAD	37661700 31.061.240	31
32	37661800 , PROTECTIONF = [46:1]# % 1 IF FILE PROTECTED	37661800 33.012.356	32
33	37661900 , PERMANENCYF = [45:1]# % 0 IF TEMPORARY FILE.	37661900 33.012.356	33
34	37661920 % 1 IF FILE HEADER IS IN FLAT	37661920 33.012.356	34
35	37661940 % DIRECTORY OF JOBDESC FILE.	37661940 33.012.356	35
36	37661950 , SYSTEMFYLEF = [44:1]# % NON REMOVABLE FILE.	37661950 321721099	36
37	37661980 , APLMAF = [43:1]# % NEW APLMA BIT. SEE HEADERKEEPER	37661980 33.030.546	37
38	37662000 , PHYSICALMODEF = [42:3]# % EXTMODE AT CREATION	37662000 31.061.240	38
39	37662100 , FILEORGF = [39:40]# % SEE FIB[3]=TANKDATA1	37662100 31.061.240	39
40	37662220 % FIB TANKDATA2	37662220 32.7X.205	40
41	37662240 %WORD 3	37662240 32.7X.205	41
42	37662250 % TIMESTAMP	37662250 32.7X.205	42
43	37662260 %WORD 4	37662260 32.7X.205	43
44	37662300 % ROWINFO	37662300 31.061.240	44
45	37662400 %WORD 5	37662400 31.061.240	45
46	37662500 , VERSIONF = [47:4]# % :=MCPHDRLEVEL % IDENTIFIES FORMAT	37662500 321721099	46
47	37662700 , PRIVUSERF = [43:01]# % 1=PRIV USER HDR	37662700 31.061.240	47
48	37662750 , SENSITIVEDATAF = [42:1]# % SCRUB WHEN RELEASED.	37662750 321721099	48
49	37662800 , SECURITYCODEF = [41:02]# % CLASS OF SECURITY.	37662800 31.061.240	49
50	37662900 , READWRITEF = [39:02]# % READ/WRITE SECURITY	37662900 31.061.240	50
51	37663000 , NUMROWSF = [37:14]# % ROWS AVAIL FOR USE.	37663000 321721099	51
52	37663100 , ROWSIZEF = [23:24]# % SEGMENTS IN EACH ROW.	37663100 321721099	52
53	37663200 % NAMEQUALIFICATIONINFO	37663200 31.061.240	53
54	37663300 %WORD 6	37663300 31.061.240	54
55	37663320 , JOBORDERF = [47:43]# % USED BY CONTROLLER FOR JOB FILES	37663320 31.193.794	55
56	37663400 , SAVEFACTORF = [47:10]# %SAVE TIME	37663400 31.061.240	56
57	37663500 , CREATEDDATEF = [37:18]# %WHEN THE FILE WAS MADE	37663500 31.061.240	57
58	37663600 , OPENERF = [19:SNSIZE]# % STK # OF OPENER	37663600 35.026.736	58
59	37663700 , FILEORGANIZATIONF = [7:3]# % FILEORGANIZATION ATTRIBUTE	37663700 31.163.604	59
60	37663800 , COPYINFOF = [4:5]# % VERSION DRCTRY STUFF	37663800 31.061.240	60

1	37663900	, DUPLICATEDBIT=[4:1]#	% FILE IS DUPLICATED	37663900	31.061.240	1
2	37664000	, COPYNUMBERF=[3:4]#	% NUMBER OF THIS COPY	37664000	31.061.240	2
3	37664100	, % ACCESSINFO		37664100	31.061.240	3
4	37664200	, %WORD 7		37664200	31.061.240	4
5	37664300	, DISKPACKF =[47:1]#	%DISKPACK(HDR)	37664300	31.061.240	5
6	37664350	, WILLCRUNCHF=[45:1]#	% WILL BE CRUNCHED WHEN OPEN COUNT=0	37664350	321721099	6
7	37664400	, CRUNCHEDF =[44:1]#	% 1 IF FILE IS CRUNCHED	37664400	321721099	7
8	37664500	, CONTENDORSF=[42:11]#		37664500	31.061.240	8
9	37664600	, IADVALIDITYF=[43:1]#	%FOR EXTERNAL IAD	37664600	31.061.240	9
10	37664700	, REBUILD F =[42:1]#	%REPRESENTATION	37664700	31.061.240	10
11	37664800	, IADINFOF =[41:10]#	% ONLY.....	37664800	31.061.240	11
12	37664900	, MODEF =[31:3]#		37664900	31.061.240	12
13	37665300	, WRITTENONF =[28:1]#	% 1 IF FILE WRITTEN ON.	37665300	31.061.240	13
14	37665400	, WROTELASTROWF=[27:1]#	% 1 IF EOF ROW WRITTEN.	37665400	31.061.240	14
15	37665500	, CHECKEOFF =[26:1]#	% 1 IF MUST FIND EOF.	37665500	31.061.240	15
16	37665510	, ORIGHEADERVERSIONF=[25:4]#	% ORIG. HDR VERSION IF THIS HEADER	37665510	350922267	16
17	37665520		% IS A HDR WHICH WAS CONVERTED WHEN	37665520	350922267	17
18	37665530		% IT WAS READ FROM DISK; 0 IF NOT	37665530	350922267	18
19	37665540		% CONVERTED (MEMORY VERSION ONLY)	37665540	350922267	19
20	37665542	, ROWADVERSIONF=[21:4]#	% VERSION OF ROWAD WORDS IN HEADER ON	37665542	350922267	20
21	37665544		% DISK (VERSION 5 AND LATER HEADERS)	37665544	350922267	21
22	37665550	, LASTACCESSDATEF=[17:18]#	%DATE LAST USED	37665550	321721099	22
23	37665600	, % D1LINKL		37665600	31.061.240	23
24	37665700	, % D1LINKWORD (CORE ONLY)	% WORD 8 OF CODEFILE	37665700	34.004.088	24
25	37665750	, % WORD 8 IS ALSO USED AS THE DMTIMESTAMPWORD (DATA FILES ONLY)		37665750	34.004.088	25
26	37665800	, DISTACKLINKF=[35:SNSIZE]#		37665800	35.026.736	26
27	37666000	, % BDINFOWORDL		37666000	31.061.240	27
28	37666100	, % WORD 9		37666100	31.061.240	28
29	37666150	, % WORD 9 IS ALSO USED AS AN ERRORWORD IN ACTIVE MCP CODE FILE HEADERS		37666150	33.030.546	29
30	37666200	, % DESTCONTROLF = 47:24,		37666200	321721099	30
31	37666400	, BDSTKNUMF =[11:SNSIZE]#		37666400	35.026.736	31
32	37666410	, % CREATIONORAPLORJOBIX WORD	% WORD 11	37666410	33.030.546	32
33	37666412	, CREATIONDATEF =[47:16]#		37666412	33.030.546	33
34	37666415	, CREATIONTIMEF =[31:31]#	% 3.3 FILE ATTRIBUTE CREATIONTIME	37666415	33.030.546	34
35	37666420	, APLAPPLICATIONF =[0:1]#	% APL IS BIT 0	37666420	32.035.221	35
36	37666500	, % JOBNUMBERL		37666500	31.061.240	36
37	37666600	, % WORD 11 OF A JOB FILE HEADER		37666600	31.061.240	37
38	37666700	, JOBVALIDIF =[47:1]#		37666700	31.061.240	38
39	37666800	, PBBITS =[45:6]#	%% 45->LPDK,44->LPSRPK,43->LPNAMEDPK	37666800	31.061.240	39
40	37666900		% 42->CPDK,41->CPSRPK,40->CPNAMEDPK	37666900	31.061.240	40
41	37667000	, LPWHEREF =[45:3]#		37667000	31.061.240	41
42	37667100	, CPWHEREF =[42:3]#		37667100	31.061.240	42
43	37667150	, JOBSUMF =[39:4]#	% CONTAINS VALUE OF JOBSUMMARY	37667150	33.044.732	43
44	37667180	, NOJOBSUMMARYIOF =[35:1]#	% CONTAINS VALUE OF NOJOBSUMMARYIO	37667180	35112511	44
45	37667190	, JSTITLESETF=[34:1]#	% IS SET IF JOBSUMMARYTITLE IS SET	37667190	351122543	45
46	37667195	, % UNUSED [33:2]	%%%%%% UNUSED [33:2] %%%%	37667195	351122543	46
47	37667200	, JOBORGNITF=[31:16]#		37667200	31.061.240	47
48	37667300	, MIXNOF =[15:16]#		37667300	31.061.240	48
49	37667400	, % COREINDEXL		37667400	31.061.240	49
50	37667500	, TIMESTAMPSYCNF=[47:1]#	% 0 IMPLIES UPDATE CATALALOG	37667500	31.061.240	50
51	37667600	, COREINDEXF =[46:47]#		37667600	31.061.240	51
52	37667700	, % DISKPACKINFO		37667700	31.061.240	52
53	37667800	, % WORD 13		37667800	31.061.240	53
54	37667900	, INTERCHANGEF=[47:1]#		37667900	31.061.240	54
55	37667940	, SINGLEF =[46:1]#		37667940	321721099	55
56	37667960	, CYLMODEHDRF=[44:1]#	% CYLIN BOUNDARY ALLOC	37667960	321721099	56
57	37668000	, SERNUMF =[43:20]#		37668000	31.061.240	57
58	37668050	, BASEUNITF =[23:LUB(12)]#		37668050	350421199	58
59	37668100	, LASTF =[11:LUB(12)]#		37668100	350421199	59
60	37668500	, % HDREOFL		37668500	31.061.240	60

1	37668600	, EOFU	= [47:20]#	% WORD 14.	CF: DSKEOFU AND FIBEOFU	37668600	340621611	1
2	37668700	, EOFV	= [27:28]#	% WORD 14.	CF: DSKEOFV AND FIBEOFV	37668700	340621611	2
3	37668740		% WORD 15			37668740	33.030.546	3
4	37668760	%%	USED FOR 3.3 FILE ATTRIBUTE USERINFO			37668760	33.030.546	4
5	37668800	%	BACKUPCONTROLL			37668800	31.061.240	5
6	37668820	%%	WORD 16 (NEXTROWWRD)			37668820	31.218.969	6
7	37668830	, ALLOCATINGF	= [47:1]#	% = 1	ALLOCATE RUNNING FOR THIS DIRECTORY HDR	37668830	31.218.969	7
8	37668840	, HDRCATF	= [46:1]#	% = 1	IF FILE IS CATALOGED	37668840	31.218.969	8
9	37668850	%	[39:10]	%	NEXT FASTIOV ROW	37668850	31.218.969	9
10	37668860	%	[29:10]	%	NEXT VASTIOV ROW	37668860	31.218.969	10
11	37668870	%	[19:10]	%	NEXT CATIOV ROW	37668870	31.218.969	11
12	37668880	%	[9:10]	%	NEXT FLATIOV OR VOLIOV ROW	37668880	31.218.969	12
13	37668900	%%	WORD 17 OF A JOB FILE HEADER.			37668900	31.218.969	13
14	37669000	, PHASEF	= [1:2]#			37669000	31.061.240	14
15	37669020	%	VALUES FOR JOBPHASE(HDR)			37669020	33.012.356	15
16	37669040	, RESTARTPHASE	= 1 #			37669040	33.012.356	16
17	37669060	, RUNPHASE	= 2 #			37669060	33.012.356	17
18	37669080	, PRINTPHASE	= 3 #			37669080	33.012.356	18
19	37669100	, NOSUMMARYOVRDF	= [4:1]#			37669100	31.061.240	19
20	37669200	, INFSF	= [5:1]#			37669200	31.061.240	20
21	37669300	, FSFF	= [6:1]#			37669300	31.061.240	21
22	37669400	, RESFF	= [7:1]#			37669400	31.061.240	22
23	37669500	, BDPRIORF	= [29:22]#			37669500	31.061.240	23
24	37669600	, CHECKPTF	= [39:10]#			37669600	31.061.240	24
25	37669700	, JFCOPIEDF	= [40:1]#			37669700	31.061.240	25
26	37669800	, CPRESTARTF	= [41:1]#			37669800	31.061.240	26
27	37669900	, CCRESTARTF	= [42:1]#			37669900	31.061.240	27
28	37670000	, CPPACKF	= [43:1]#			37670000	31.061.240	28
29	37670100	, FSF	= [44:1]#			37670100	31.061.240	29
30	37670200	, IBF	= [45:1]#			37670200	31.061.240	30
31	37670300	, RESF	= [46:1]#			37670300	31.061.240	31
32	37670400	, ROWF	= [47:1]#			37670400	31.061.240	32
33	37670450	%%	WORD 17 ALSO USED FOR 3.3 FILE ATTRIBUTES ALTERDATE AND ALTERNATE			37670450	33.030.546	33
34	37670460	, ALTERDATEF	= [47:16]#			37670460	33.030.546	34
35	37670480	, ALTERNATEF	= [31:32]#			37670480	33.030.546	35
36	37670500	%	PBITIOCONTROLL			37670500	31.061.240	36
37	37670600		% WORD 18			37670600	31.061.240	37
38	37670650	%%	HDRCM = [47:01] = 1 ---> FOR POST 33 CM. = 0 ---> PRE 33			37670650	33.030.546	38
39	37670700	, PBITIOHIATUSF	= [1:1]#			37670700	31.061.240	39
40	37670800	, CODEIOCOUNTF	= [15:14]#			37670800	31.061.240	40
41	37670840		% WORD 19			37670840	33.030.546	41
42	37670860	%%	WORD 19 IS USED FOR 3.3 FILE ATTRIBUTE USETIME ALIAS ACESSTIME.			37670860	33.030.546	42
43	37670870	, ACCESSDATEF	= [47:16]#			37670870	33.030.546	43
44	37670880	, ACESSTIMEF	= [31:32]#			37670880	33.030.546	44
45	37670882					37670882	350922267	45
46	37670884	%	ROW ADDRESSES BEGIN AT FIRSTROWINDEX AND CONTINUE FOR AS MANY			37670884	350922267	46
47	37670886	%	WORDS AS THERE ARE ROWS (MAXIMUM=1023). ROW ADDRESSES ARE IN			37670886	350922267	47
48	37670888	%	STANDARD MASS-ADDRESS FORM.			37670888	350922267	48
49	37670890	%	THE SECURITY INFORMATION FOLLOWS THE ROW ADDRESSES;			37670890	350922267	49
50								50
51								51
52								52
53								53
54								54
55								55
56								56
57								57
58								58
59								59
60								60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

Line	Code	Description	Address	Value	Line
1		WORKFILE [NOT NAMED]		5:05 PM TUESDAY, APRIL 9, 1985	1
2					2
3					3
4	37623600	DEFINE	37623600	34.034.823	4
5	37623700		37623700	31.036.234	5
6	37623800	%	37623800	22.000.145	6
7	37623900	%	37623900	31.041.240	7
8	37624000	% THE FOLLOWING IS A DESCRIPTION OF A FILE HEADER	37624000	31.041.240	8
9	37624100	%	37624100	31.041.240	9
10	37625500	% THOSE NAMES FOR ACCESSING THINGS FROM THE HEADER.	37625500	26.030.523	10
11	37625600	ACCESSDATE(HDR) = ACCESSDATEWORD(HDR).LASTACCESSDATEF #	37625600	321721099	11
12	37625700	,ACCESSDATEWORD(HDR) = HDR[7] #	37625700	21.000.119	12
13	37625720	,ACCESSTIMESTAMP(HDR) = HDR[19]# % 3.3 FILE ATTRIBUTE USETIME	37625720	33.030.546	13
14	37625740	,ACCESSTIME(HDR) = ACCESTIMESTAMP(HDR).ACCESTIMEF#	37625740	33.030.546	14
15	37625745	,ACCESSTSDATE(HDR) = ACCESTIMESTAMP(HDR).ACCESTDATEF#	37625745	33.030.546	15
16	37625800	,ALLOCATEDROW(HDR,ROW) = BOOLEAN(HDR[FIRSTROWINDEX + ROW])	37625800	26.025.307	16
17	37625900	.ALLOCATEDROWF #	37625900	26.025.307	17
18	37626000	,ALLOCATING(HDR)= NEXTROWWRD(HDR).ALLOCATINGF #	37626000	31.218.969	18
19	37626020	,ALERTTIMESTAMP(HDR) = HDR[17] # % 3.3 FILE ATTRIBUTES	37626020	33.030.546	19
20	37626030	,ALERTIME(HDR) = ALERTTIMESTAMP(HDR).ALERTIMEF #	37626030	33.030.546	20
21	37626040	,ALTERDATE(HDR) = ALERTTIMESTAMP(HDR).ALTERDATEF #	37626040	33.030.546	21
22	37626060	,APLMA(HDR) = HDR[2].APLMAF# % THIS IS A DUPLICATED APLMA BIT. THE	37626060	33.030.546	22
23	37626080	% OLD APLMA BIT (OLDAPLMA) IS IN WORD 11 BIT 0. SEE HEADERKEEPER.	37626080	33.030.546	23
24	37626090	,AVAILO(HDR) = HDR[1].AVAILOF# % FORMERLY SPEED ATTRIBUTE, AVAILABLE	37626090	34.028.677	24
25	37626100	,BASEUNIT(HDR) = HDR[13].BASEUNITF #	37626100	22.000.371	25
26	37626200	,BACKUPFILE(HDR) = FILEKIND(HDR) IS BACKUPPRINTER	37626200	26.025.252	26
27	37626300	OR FILEKIND(HDR) IS BACKUPPUNCH #	37626300	26.025.252	27
28	37626400	,BDINFOWORD(HDR) = HDR[9] #	37626400	26.030.483	28
29	37626500	,BACKUPCONTROLIX = 17 #	37626500	26.030.523	29
30	37626600	,BACKUPCONTROL(HDR) = HDR[BACKUPCONTROLIX] #	37626600	24.000.238	30
31	37626800	,BDSTACKNUM(HDR) = BDINFOWORD(HDR).BDSTKNUMF#	37626800	24.000.183	31
32	37626900	,BDSERIALNUM(HDR) = BDINFOWORD(HDR).BDSERIALF #	37626900	24.000.183	32
33	37627000	,BUMPCODEIOCOUNT(HDR,TEMP)=	37627000	31.016.124	33
34	37627100	BEGIN [ CONTROLSTATE ]	37627100	32.032.159	34
35	37627200	TEMP:=BUZZ(HDR[18]);	37627200	31.016.124	35
36	37627300	TEMP.CODEIOCOUNTF:=TEMP.CODEIOCOUNTF+1;	37627300	31.016.124	36
37	37627400	HDR[18]:=TEMP; % UNLOCK	37627400	31.016.124	37
38	37627500	END#	37627500	31.016.124	38
39	37627600	,CATALOGED(HDR) = NEXTROWWRD(HDR).HDCATF#	37627600	31.218.969	39
40	37627900	,CHECKEOF(HDR) = HDR[7].CHECKEOFF #	37627900	26.025.196	40
41	37628000	,CONTENDORS(HDR) = HDR[7].CONTENDORSF #	37628000	22.000.265	41
42	37628300	,COPYNUMBER(HDR)= FILEQUALITY(HDR).COPYNUMBERF #	37628300	26.090.564	42
43	37628400	,COREINDEX(HDR) = HDR[12].COREINDEXF #	37628400	29.0362	43
44	37628500	,CREATIONDATE(HDR) = HDR[6].CREATEDDATEF#	37628500	21.000.003	44
45	37628550	,CREATIONORAPLORJOBIX = 11#	37628550	33.030.546	45
46	37628560	,CREATIONTIMESTAMP(HDR) = HDR[CREATIONORAPLORJOBIX].[47:47]#	37628560	33.030.546	46
47	37628580	,CREATIONTSDATE(HDR) = HDR[CREATIONORAPLORJOBIX].CREATIONDATEF #	37628580	33.030.546	47
48	37628590	,CREATIONSTIME(HDR) = HDR[CREATIONORAPLORJOBIX].CREATIONTIMEF#	37628590	33.030.546	48
49	37628600	,CRUNCHED(HDR) = HDR[7].CRUNCHEDF #	37628600	24.000.268	49
50	37628700	,CYLMODE(HDR) = HDR[13].CYLMODEHDRF # %CYLINDER ALLOC	37628700	26.000.086	50
51	37628800	,DILINKWORD(HDR) = HDR[8] #	37628800	34.004.088	51
52	37628850	,DILINK(HDR) = DILINKWORD(HDR).DISTACKLINKF #	37628850	34.004.088	52
53	37628900	,DECREMENTCODEIOCOUNT(HDR,TEMP)=	37628900	DATEANDT	53
54	37629000	BEGIN [ CONTROLSTATE ]	37629000	32.032.159	54
55	37629100	TEMP:=BUZZ(HDR[18]);	37629100	31.016.124	55
56	37629200	TEMP.CODEIOCOUNTF:=TEMP.CODEIOCOUNTF-1;	37629200	31.016.124	56
57	37629300	IF TEMP.CODEIOCOUNTF = 0 THEN	37629300	31.016.124	57
58	37629400	TEMP.PBITIOHIATUSF:=0;	37629400	31.016.124	58
59	37629500	HDR[18]:=TEMP; % UNLOCK	37629500	31.016.124	59
60	37629600	END#	37629600	31.016.124	60

1	37629700	,DELAYEDCRUNCH(HDR) = BOOLEAN(HDR[7].WILLCRUNCHF) #	37629700	24.000.268	1
2	37629800	,DELAYCRUNCH(HDR) = HDR[7].WILLCRUNCHF #	37629800	24.000.268	2
3	37629900	,DISKADR(HDR,REC,RWSZ) = % RETURNS A SEGMENT ADDRESS	37629900	350541499	3
4	37630000	(ROWADDRESSWORD(HDR,(REC) DIV RWSZ).SEGADDRESSF+(REC) MOD RWSZ) #	37630000	350541499	4
5	37630025	,DISKADRINFO(HDR,SEG) = % RETURNS A SEGMENT ADDRESS AND LU #	37630025	350611756	5
6	37630035	(O & ROWUNIT(HDR,(SEG DIV ROWSIZE(HDR))) ROWUNITNUMF	37630035	350611756	6
7	37630055	& (ROWADDRESSWORD(HDR,(SEG DIV ROWSIZE(HDR))).SEGADDRESSF+	37630055	350611756	7
8	37630060	(SEG MOD ROWSIZE(HDR)))	37630060	350611756	8
9	37630065	ROWBASEADDRESSF) #	37630065	350541499	9
10	37630100	,DISKBLOCKING(HDR) = HDR[3]#	37630100	21.000.003	10
11	37630200	,DISKBLOCKSIZE(HDR) = DISKBLOCKING(HDR).BLOCKSIZEF#	37630200	31.163.604	11
12	37630300	,DISKCYCLE(HDR) = DISKGENEALOGYWORD(HDR).CYCLE #	37630300	26.030.523	12
13	37630400	,DISKFILEMODE(HDR) = HDR[7].MODEF #	37630400	22.000.265	13
14	37630500	,DISKGENEALOGY(HDR) = DISKGENEALOGYWORD(HDR).GENEALOGYF #	37630500	26.030.523	14
15	37630600	,DISKGENEALOGYWORD(HDR) = HDR[10] #	37630600	26.030.523	15
16	37630700	,DISKMAXRECSIZE(HDR) = DISKBLOCKING(HDR).MAXRECSIZEF#	37630700	31.163.604	16
17	37630800	,DISKPACK(HDR)=BOOLEAN(HDR[7].DISKPACKF)#	37630800	21.000.187	17
18	37630900	,DISKPACKHDR(HDR)=HDR[7].DISKPACKF#	37630900	21.000.187	18
19	37631000	,DISKPACKSERNUM(HDR,ROW)=HDR[ROW].DPSERIALOREUF#	37631000	21.000.199	19
20	37631100	,DISKPACKWORD(HDR) = HDR[13]#	37631100	22.000.152	20
21	37631200	,DISKSAVEFCTR(HDR) = HDR[6].SAVEFACTORF #	37631200	21.000.003	21
22	37631400	,DISKVERSION(HDR) = DISKGENEALOGYWORD(HDR).GENVERSN #	37631400	26.030.523	22
23	37631500	,DMSROWLOCK(HDR,ROW)=HDR[ROW+FIRSTROWINDEX].DMLOCKBITSF#	37631500	25.020.200	23
24	37631600	,DMTIMESTAMP(HDR) = DILINKWORD(HDR) #	37631600	34.004.088	24
25	37631700	,DSKEOF(HDR)=HDR[14]#	37631700	21.000.119	25
26	37631800	,DSKEOFU(HDR)=DSKEOF(HDR).EOFU#	37631800	21.000.119	26
27	37631900	,DSKEOFV(HDR)=DSKEOF(HDR).EOFV#	37631900	21.000.119	27
28	37631920	% THE CONVENTION FOR DSKEOF IS SUPPOSED TO BE:	37631920	331241208	28
29	37631940	% DSKEOFV = NUMBER OF WHOLE SEGMENTS IN THE FILE;	37631940	331241208	29
30	37631960	% DSKEOFU = NUMBER OF BITS IN ANY PARTIAL LAST SEGMENT (MAY BE 0).	37631960	331241208	30
31	37632000	,DUPLICATED(HDR)=BOOLEAN(HDR[6].DUPLICATEDBIT)#	37632000	21.000.232	31
32	37632100	,ERRORWORD(HDR) = HDR[9] # % NOTE WORD 9 IS ALSO USED FOR BDINFO	37632100	33.030.546	32
33	37632200	,FILEORGANIZATION(HDR)=HDR[6].FILEORGANIZATIONF #	37632200	31.163.604	33
34	37632300	,FILEQUALITY(HDR)=HDR[6]#	37632300	21.000.119	34
35	37632400	,FILESTRUCTURE(HDR)=HDR[5]#	37632400	21.000.119	35
36	37632500	,FILEKIND(HDR) = HDR[1].FILEKINDF #	37632500	26.025.252	36
37	37632600	,FIBINFO(HDR) = HDR[2]#	37632600	21.000.003	37
38	37632700	,FIRSTROWINDEX = 20 #	37632700	26.030.480	38
39	37632900	,HDRBLOCKLENGTH(HDR) = HDR[0].HDRBLOCKLENGTHF#	37632900	26.025.406	39
40	37632950	,HDRCM = [47:01] # % MAY BE TEMPORARY. SEE HEADERKEEPER NOTES.	37632950	33.030.546	40
41	37633000	,HDRLOCATION(HDR) = HDR[0].HDRLOCATIONF#	37633000	26.025.406	41
42	37633100	,HDRMAKER(HDR) = HDR[6].OPENERF#	37633100	31.016.124	42
43	37633200	,HDRMARKER(HDR) = HDR[0].MARKERF#	37633200	26.025.406	43
44	37633300	,HEADERGUARDP(HDR) = POINTER(HDR[FIRSTROWINDEX	37633300	26.090.564	44
45	37633400	+ NUMBEROFROWS(HDR)],8) #	37633400	26.090.564	45
46	37633500	,HEADERINFO(HDR) = HDR[1] #	37633500	21.000.119	46
47	37633550	,HEADERROWADVERSION(HDR) = HDR[7].ROWADVERSIONF #	37633550	350922267	47
48	37633600	,HEADERSERNUM(HDR) = DISKPACKWORD(HDR).SERNUMF #	37633600	26.090.564	48
49	37633700	,HEADERSIZE(HDR) = HDR[1].HEADERSIZEF #	37633700	21.000.003	49
50	37633800	,HEADERTITLE(HDR) = POINTER(HDR[HEADERSIZE(HDR)],8)#	37633800	26.030.483	50
51	37633900	,HEADERTITLEHEAD(HDR) = HDR[HEADERSIZE(HDR)]#	37633900	31.143.489	51
52	37634000	,HEADERVERSION(HDR) = HDR[5].VERSIONF#	37634000	21.000.199	52
53	37634100	,IADFILE(HDR)=BOOLEAN(HDR[2].IADBIT)#	37634100	21.000.232	53
54	37634200	,IADFILER(HDR)=HDR[2].IADBIT#	37634200	23.001.257	54
55	37634300	,IADINFO(HDR)=HDR[7].IADINFOF#	37634300	23.001.257	55
56	37634400	,INTERCHANGE(HDR)=BOOLEAN(HDR[13].INTERCHANGEF)#	37634400	21.000.087	56
57	37634500	,INTERCHG(HDR)=HDR[13].INTERCHANGEF#	37634500	22.000.098	57
58	37634600	,JOBNUMBERIX = CREATIONORAPLORJOBIX #	37634600	33.030.546	58
59	37634700	,JOBNUMBER(HDR) = HDR[JOBNUMBERIX] #	37634700	24.000.238	59
60	37634800	,JOBPHASE(HDR) = BACKUPCONTROL(HDR).PHASEF #	37634800	24.000.238	60

1	37634900	,JOBMIXNO(HDR) = JOBNUMBER(HDR).MIXNOF #	37634900	24.000.238	1
2	37635000	,LASTCP(HDR) = BACKUPCONTROL(HDR).CHECKPTF #	37635000	DISKFILE	2
3	37635100	,LASTU(HDR)=HDR[13].LASTF#	37635100	21.000.157	3
4	37635200	,LINKVALIDITY(HDR)=HDR[7].IADVALIDITYF#	37635200	23.001.257	4
5	37635300	,NEXTROW(HDR,T) = NEXTROWWRD(HDR)	37635300	26.030.480	5
6	37635400	.[(CASE T OF (39,29,9,19,9)):10] #	37635400	26.030.480	6
7	37635500	,NEXTROWWRD(HDR)= HDR[16] #	37635500	26.030.480	7
8	37635600	,NOROWV = 1023 #	37635600	26.030.480	8
9	37635700	,NUMBEROFROWS(HDR) = HDR[5].NUMROWSF #	37635700	21.000.003	9
10	37635720	,NUMBEROFSEGMENTS(HDR) = (DSKEOFV(HDR) +	37635720	32.033.180	10
11	37635740	(DSKEOFU(HDR)+(30*48)-1) DIV (30*48)) #	37635740	32.033.180	11
12	37635800	,OLDAPLMA(HDR)=HDR[CREATIONORAPLORJOBIX].APLAPPLICATIONF#	37635800	33.030.546	12
13	37635900	,OPENCOUNT(HDR) = HDR[1].OPENCOUNTF #	37635900	21.000.003	13
14	37635950	,ORIGHEADERVERSION(HDR) = HDR[7].ORIGHEADERVERSION #	37635950	350922267	14
15	37636000	,OUTSTANDINGIOCOUNT(HDR) = HDR[18]# % FROM PBIT,EOJ,PROGRAMDUMP	37636000	31.016.124	15
16	37636100	,PERMANENT(HDR) = BOOLEAN(HDR[2].PERMANENCYF) #	37636100	22.000.457	16
17	37636200	,PERMANENTDISKFILE(HDR) = HDR[2].PERMANENCYF #	37636200	21.000.052	17
18	37636300	,PMODE(HDR) = FIBINFO(HDR).PHYSICALMODEF#	37636300	33.003.076	18
19	37636400	,PRIVILEGEDBIT(HDR) = HDR[5].PRIVUSERF #	37636400	21.000.199	19
20	37636500	,PROTECTEDBIT(HDR)=HDR[2].PROTECTIONF #	37636500	21.000.003	20
21	37636600	,READWRITEUSE(HDR)=HDR[5].READWRITEF #	37636600	21.000.199	21
22	37636700	,REBUILDLINKS(HDR)= HDR[7].REBUILDF#	37636700	23.001.257	22
23	37636900	,ROWADDRESSWORD(HDR,ROWNR) = HDR[FIRSTROWINDEX + ROWNR] #	37636900	26.025.307	23
24	37637000	,ROWUNIT(HDR,ROW)=FAMINDEXUNIT(BASEUNIT(HDR),	37637000	350541499	24
25	37637050	HDR[FIRSTROWINDEX+ROW].FAMILYINDEXF) #	37637050	350541499	25
26	37637100	% ROWINDEX(FIB,SEGADDRESS,LU) IS A PROCEDURE THAT RETURNS THE ROW #	37637100	351802860	26
27	37637150	% OF THE FIB HEADER ROW THAT RESIDES ON LU AND CONTAINS SEGADDRESS.	37637150	351802860	27
28	37637200	,ROWSIZE(HDR) = HDR[5].ROWSIZEF #	37637200	21.000.003	28
29	37637300	,SAVETIME(HDR) = HDR[6].SAVEFACTORF #	37637300	21.000.003	29
30	37637400	,SECURITYAREA = 0#	37637400	31.016.124	30
31	37637500	,SECURITYCLASS(HDR)=HDR[5].SECURITYCODEF #	37637500	21.000.199	31
32	37637600	,SECURITYINFO(HDR) = HDR[HDR[5].NUMROWSF+FIRSTROWINDEX] #	37637600	21.000.003	32
33	37637700	,SECURITYSIZE(HDR) = HDR[1].SECINFOSIZEF #	37637700	21.000.199	33
34	37637800	,SENSITIVEDATA(HDR) = BOOLEAN(SENSITIVEDATAFILE(HDR)) #	37637800	26.090.555	34
35	37637900	,SENSITIVEDATAFILE(HDR) = FILESTRUCTURE(HDR).SENSITIVEDATAF #	37637900	26.090.555	35
36	37637940	,SETCREATIONTIMESTAMP(HDR,VALU)=	37637940	33.030.546	36
37	37637960	CREATIONTIMESTAMP(HDR):=(VALU).[47:47] #	37637960	33.030.546	37
38	37638000	,SINGLEPACK(HDR)=BOOLEAN(HDR[13].SINGLEF)#	37638000	21.000.157	38
39	37638200	,SYSTEMFILE(HDR)= FIBINFO(HDR).SYSTEMFYLF #	37638200	26.030.480	39
40	37638300	,TANKD1(HDR)=HDR[2].[43:44]#	37638300	21.000.257	40
41	37638400	,TANKD2(HDR)=HDR[3]#	37638400	21.000.257	41
42	37638500	,TEMPORARY(HDR) = NOT PERMANENT(HDR) #	37638500	26.030.480	42
43	37638600	,TIMESTAMP(HDR) = HDR[4] #	37638600	26.030.480	43
44	37638700	,TIMESTAMPHEADER(HDR)= TIMESTAMP(HDR) := DATEANDTIME #	37638700	26.030.480	44
45	37638800	,TIMESTAMPSYNC(HDR) = HDR[12].TIMESTAMPSYNCF #	37638800	29..0362	45
46	37638900	,TIMESTAMPSYNCD(HDR) = BOOLEAN(HDR[12].TIMESTAMPSYNCF) #	37638900	29..0362	46
47	37638920	,TSTOMICRO = 38.4 # % THE TIMESTAMP (TS) TIME IS STORED IN 2.4 MICRO	37638920	33.030.546	47
48	37638940	% SECOND UNITS RIGHT TRUNCATED BY 4 BITS. TO GET THE TS TIME TO	37638940	33.030.546	48
49	37638960	% MICROSECONDS, MUST MULTIPLY BY 2.4 AND ADJUST BY 16, THUS 38.4;	37638960	33.030.546	49
50	37638980	% CONVERSELY, TO OBTAIN TS TIME FROM MICROSECONDS, DIV BY THIS #.	37638980	33.030.546	50
51	37639000	,UPDATED(HDR) = HDR[1].UPDATEBITF#	37639000	21.000.003	51
52	37639050	,USERINFO(HDR) = HDR[15]# % 3.3 FILE ATTRIBUTE.	37639050	33.030.546	52
53	37639100	,WAITFORIOHIATUS(HDR,TEMP)= % RETURNS 0 IN TEMP IF OK	37639100	31.016.124	53
54	37639200	BEGIN [ CONTROLSTATE ]	37639200	32.032.159	54
55	37639300	TEMP:=BUZZ(HDR[18]);	37639300	31.016.124	55
56	37639400	IF TEMP.CODEIOCOUNTF = 0 THEN	37639400	31.016.124	56
57	37639500	BEGIN	37639500	31.016.124	57
58	37639600	HDR[18]:=TEMP; % UNLOCK	37639600	31.016.124	58
59	37639700	TEMP:=0; % RETURN OK RESULT	37639700	31.016.124	59
60	37639800	END ELSE	37639800	31.016.124	60

1	37639900	BEGIN	37639900	31.016.124	1
2	37640000	HDR[18]:=TEMP:==&1 PBITIOHIATUSF; % UNLOCK	37640000	31.016.124	2
3	37640100	THRU 10 DO	37640100	31.016.124	3
4	37640200	BEGIN	37640200	31.016.124	4
5	37640300	IF TEMP.PBITIOHIATUSF = 1 THEN	37640300	31.016.124	5
6	37640400	BEGIN	37640400	31.016.124	6
7	37640500	TIMETUNNEL(1);	37640500	31.016.124	7
8	37640600	TEMP:=BUZZ(HDR[18]);	37640600	31.016.124	8
9	37640700	HDR[18]:=TEMP; % UNLOCK	37640700	31.016.124	9
10	37640800	END;	37640800	31.016.124	10
11	37640900	END;	37640900	31.016.124	11
12	37641000	TEMP:=*.PBITIOHIATUSF; % SET RESULT	37641000	31.016.124	12
13	37641100	END;	37641100	31.016.124	13
14	37641200	END#	37641200	31.016.124	14
15	37641300	,WRITTENON(HDR) = HDR[7].WRITTENONF #	37641300	26.025.196	15
16	37641400	,WROTELASTROW(HDR) = HDR[7].WROTELASTROWF #	37641400	26.025.196	16
17	37641500	%THESE DEFINITIONS ARE TO BE USED FOR "FILEKIND" IN	37641500	31.050.290	17
18	37641600	%FILE HEADERS AND DIRECTORY NAME ENTRIES:	37641600	31.050.290	18
19	37641800	,VERSIONDIRECTORY = (VERSIONDIRECTORYAV).MVALUE #	37641800	34.022.438	19
20	37642000	,CATALOGFILE = (CATALOGAV).MVALUE #	37642000	34.022.438	20
21	37642100	,BACKUPPRINTER = (BACKUPPRINTERAV).MVALUE #	37642100	34.022.438	21
22	37642200	,RECONSTRUCTIONFILE = (RECONSTRUCTIONFILEAV).MVALUE #	37642200	34.022.438	22
23	37642300	,SYSTEMDIRFILE = (SYSTEMDIRFILEAV).MVALUE #	37642300	34.022.438	23
24	37642400	,JOBDESCFILE = (JOBDESCFILEAV).MVALUE #	37642400	34.022.438	24
25	37642500	,ARCHIVELOG = (ARCHIVELOGAV).MVALUE #	37642500	34.022.438	25
26	37642550	,BOOTCODE = (BOOTCODEAV).MVALUE # % INITIALIZATION CODEFILE	37642550	350731899	26
27	37642600	,XDISKFILE = (XDISKFILEAV).MVALUE #	37642600	34.022.438	27
28	37642700	% FILEKIND 16 IS RESERVED FOR FUTURE IMPLEMENTATION %	37642700	31.156.552	28
29	37642800	% , BACKUPPRINTER = 16 # % FOR FUTURE IMPLEMENTATION. %	37642800	22.000.071	29
30	37642900	,BACKUPPUNCH = (BACKUPPUNCHAV).MVALUE # %	37642900	34.022.438	30
31	37643000	,REMOTEBACKUP = (REMOTEBACKUPAV).MVALUE # %	37643000	34.022.438	31
32	37643100	,REMOTEAUDIT = (REMOTEAUDITAV).MVALUE # %	37643100	34.022.438	32
33	37643200	,COMPILERCODEFILE = (COMPILERCODEFILEAV).MVALUE #	37643200	34.022.438	33
34	37643300	,CHECKPOINTFILE = (CHECKPOINTFILEAV).MVALUE #	37643300	34.022.438	34
35	37643400	,CPJOBFILE = (CPJOBFILEAV).MVALUE #	37643400	34.022.438	35
36	37643500	,DCPCODE = (DCPCODEAV).MVALUE #	37643500	34.022.438	36
37	37643700	,RECOVERYFILE = (RECOVERYFILEAV).MVALUE #	37643700	34.022.438	37
38	37643800	,SCHEDULEFILE = (SCHEDULEFILEAV).MVALUE #	37643800	34.022.438	38
39	37643900	,INFOFILE = (INFOFILEAV).MVALUE #	37643900	34.022.438	39
40	37644000	,LIBRARYCODE = (LIBRARYCODEAV).MVALUE #	37644000	34.022.438	40
41	37644100	,INTRINSICFILE = (INTRINSICFILEAV).MVALUE #	37644100	34.022.438	41
42	37644200	,MCPCODEFILE = (MCPCODEFILEAV).MVALUE #	37644200	34.022.438	42
43	37644300	,ALGOLCODE = (ALGOLCODEAV).MVALUE #	37644300	34.022.438	43
44	37645000	,DCALGOLCODE = (DCALGOLCODEAV).MVALUE #	37645000	34.022.438	44
45	37645300	,JOBCODE = (JOBCODEAV).MVALUE #	37645300	34.022.438	45
46	37645400	,DMALGOLCODE = (DMALGOLCODEAV).MVALUE #	37645400	34.022.438	46
47	37645600	,CODEFILE = (CODEFILEAV).MVALUE #	37645600	34.022.438	47
48	37646000	,BOUNDPCODE = (BOUNDPCODEAV).MVALUE #	37646000	350812184	48
49	37647900	,FIRMWARE = (FIRMWAREAV).MVALUE # % DISK PACK FIRMWARE FILE	37647900	34.022.438	49
50	37648000	,DATA = (DATAAV).MVALUE #	37648000	34.022.438	50
51	37648100	,SEQDATA = (SEQDATAAV).MVALUE #	37648100	34.022.438	51
52	37648500	,CDATA = (CDATAAV).MVALUE #	37648500	34.022.438	52
53	37648700	,DBRESTARTSET = (DBRESTARTSETAV).MVALUE #	37648700	34.022.438	53
54	37648800	,DBDATA = (DBDATAAV).MVALUE #	37648800	34.022.438	54
55	37648900	,DATAKIND(T) = (T GEQ DATA) #	37648900	29..0379	55
56	37649000	,NONEXEC(T)=(T GEQ LIBRARYCODE AND T LEQ MCPCODEFILE) #	37649000	DISKFILE	56
57	37649100	,SYMBOLFILE(T)=(T GEQ ALGOLSYMBOL AND T LSS RSNETFILE) #	37649100	27..0515	57
58	37649200	,DANGEROUS(T)= (T EQL BOOTCODE OR	37649200	350731899	58
59	37649250	(T GEQ COMPILERCODEFILE AND	37649250	350731899	59
60	37649300	T LEQ CODEFILE AND T NEQ INFOFILE)) #	37649300	350731899	60

1	37649400	% DANGEROUS FILES ARE THOSE WHICH CAN BE SYSTEM FATAL	37649400	27..0147	1
2	37649500	% IF THEY BECOME CORRUPTED	37649500	27..0147	2
3	37649600	%DISK FILE ACCESS MODES	37649600	22.000.265	3
4	37649700	,SHARED = 0 #	37649700	22.000.265	4
5	37649800	,EXCLUSIVE = 1 #	37649800	22.000.265	5
6	37649900	,INPUTONLY = 2 #	37649900	22.000.265	6
7	37650000	,OUTPUTONLY = 3 #	37650000	22.000.265	7
8	37650100	% MNEMONICS FOR FILEORGANIZATION ATTRIBUTE	37650100	31.163.604	8
9	37650200	,NOTRESTRFO = 0 #	37650200	31.163.604	9
10	37650300	,RELATIVEFO = 1 #	37650300	31.163.604	10
11	37650350	,INDEXEDFO = 2 #	37650350	331221137	11
12	37650500	,INDEXEDNOTRESTRFO = 3 # % ISAM FILE WITH RELATIVE KEYS.	37650500	331221137	12
13	37650600	% INFORMER CODE FOR CALLS ON JEDGARHOOVER	37650600	21.000.003	13
14	37650700	,OPENSEcurity = 0 #	37650700	21.000.003	14
15	37650800	,DIDDLERSECURITY = 1 #	37650800	21.000.003	15
16	37650900	,ENTERFILESECURITY = 2 #	37650900	21.000.003	16
17	37651000	,DOCTORSECURITY = 3 #	37651000	21.000.003	17
18	37651100	,USERDATASECURITY = 4 #	37651100	23.001.507	18
19	37651200	,ATTRIBUTESECURITY = 5 #	37651200	21.000.003	19
20	37651300	,LRSECURITY = 6 #	37651300	21.000.003	20
21	37651400	,LIBRARY1 = 7 #	37651400	21.000.003	21
22	37651500	,LIBRARY2 = 8 #	37651500	21.000.003	22
23	37651600	,NEWFIBBER = 9 #	37651600	21.000.003	23
24	37651700	,ILLEGALCOL = 10 #	37651700	21.000.003	24
25	37651800	,BDFILESECURITY = 11 #	37651800	29..0055	25
26	37651900	% FILE SECURITY CLASSES	37651900	21.000.003	26
27	37652000	,CLASSA = 0 #	37652000	21.000.003	27
28	37652100	,CLASSB = 1 #	37652100	21.000.003	28
29	37652200	,CLASSC = 2 #	37652200	21.000.003	29
30	37652300	,PRIVATE = 3 #	37652300	21.000.003	30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
38					38
39					39
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
48					48
49					49
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
58					58
59					59
60					60



1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	M I S C	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	APPENDIX H	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1	JSD/MEMDUMP/EXAMPLE (11/08/84)	3:00 PM TUESDAY, APRIL 9, 1985	1
2			2
3			3
4	BEGIN		4
5	REAL		5
6	R1		6
7	,R2		7
8	,R3		8
9	,R4		9
10	;		10
11			11
12	EBCDIC ARRAY EB [0:179];		12
13	ARRAY A[0] = EB;		13
14			14
15	PROCEDURE CAUSEFINALFAULT;		15
16	BEGIN		16
17			17
18	REPLACE POINTER(A) BY "THIS SHOULD CAUSE AN ERROR"		18
19	FOR 32 WORDS;        % SEG ARRAY		19
20			20
21	END;		21
22			22
23	PROCEDURE CAUSEFIRSTFAULT;		23
24	BEGIN		24
25			25
26	R1:=R2/R3;  % DIV ZERO		26
27			27
28	END;		28
29			29
30	INTERRUPT INT;		30
31	BEGIN		31
32			32
33	CAUSEFINALFAULT;		33
34			34
35	END;		35
36			36
37	EPILOG PROCEDURE EP;		37
38	BEGIN		38
39			39
40	ENABLE INT;		40
41	ON ANYFAULT,		41
42	WAIT(MYSELF.EXCEPTIONEVENT);		42
43	CAUSEFIRSTFAULT;		43
44			44
45	END;		45
46			46
47	RESET(MYSELF.EXCEPTIONEVENT);		47
48	ATTACH INT TO MYSELF.EXCEPTIONEVENT;		48
49	DISABLE INT;		49
50			50
51	END.		51
52			52
53			53
54			54
55			55
56			56
57			57
58			58
59			59
60			60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60



1		1
2	INPUT: STACK 88	2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1	##### PROCESSOR 4 TOOK DUMP FROM THIS STACK #####			1
2	STACKDUMP FOR STACK 088 MIX NUMBER 2575/2701			2
3	NAME: (LFPROG)OBJECT/JSD/MEMDUMP/EXAMPLE.			3
4				4
5	(DS-ED @ 0003:0005:2)			5
6	JOB MESSAGES: RSVP: (NULL)			6
7	DISPLAY: SEG ARRAY ERROR @ (00001300)*			7
8	ACCEPT: (NULL)			8
9	LOCK STATUS: (NONE FOUND)			9
10	STACK KIND: TASK STACKINFO = 0 004801 780000			10
11	STATUS: ALIVE DSED STACKSTATUS = 0 FFF342 F13910			11
12	PROCESS TYPE: PROCESS LANGUAGE: DCALGOL 35.220			12
13	BOSR=338D6 LOSR=33AFE LENGTH=00228 (552)			13
14				14
15	O083 (01,0003) 1 570A55 C95CFO OP: BCL:E RESET <i>last valid dump</i>			15
16	O082 (01,0002) 0 CD5A64 665473			16
17	O081 3 000807 184CB9 RCW: LL=1, CNTRL STATE [MCP SEGMENT @ OCB9:0071:4 (21075040)]			17
18	SEG DESC: 3 880017 77BD65 ----- PROCESSKILL			18
19	CODE: 3 CD5A64 665473 3 BEFFFF FFFFFF 3 570A55 C95CFO >3 B195B4 ABB235< 3 95B840 CBA5B2			19
20	O080 ----D[01]=>3 C08000 804009 *MSCW: PREVIOUS MSCW @ 0077; D[0]=0008 IN STACK 008			20
21				21
22	O07F (01,0008) 5 C00034 7338D6 DESC [PRESENT-COPY]: DATA, LENGTH=839 (POINTS @ OFFSET=0000 IN THIS STACK)			22
23	O07E (01,0007) 0 000000 000000			23
24	O07D (01,0006) 0 000000 00007F			24
25	O07C (01,0005) 0 000000 00033A			25
26	O07B (01,0004) 0 000000 000000			26
27	O07A (01,0003) 0 FFFFFFFF FFFFFFFF			27
28	O079 (01,0002) 0 000000 000014			28
29	O078 3 000A25 B84C72 RCW: LL=1, CNTRL STATE [MCP SEGMENT @ OC72:025B:5 (11385000)]			29
30	SEG DESC: 3 A00025 D014EC ----- HARDWAREINTERRUPT68			30
31	CODE: 3 60ADAB B23595 3 B840CB A5B260 3 A6B7BD 9E27B6 >3 B8AE40 3AABB5< 3 A3A3FF FFFFFFFF			31
32	O077 ----D[01]=>3 C08000 804008 *MSCW: PREVIOUS MSCW @ 006F; D[0]=0008 IN STACK 008			32
33				33
34	O076 (01,0007) 7 088224 308C72 PCW			34
35	O075 (01,0006) 0 000000 000004			35
36	O074 (01,0005) 0 000000 00000E			36
37	O073 (01,0004) 0 000000 000003			37
38	O072 (01,0003) 0 AB23E0 OD3038			38
39	O071 (01,0002) 0 000009 000400 ***** SEGMENTED ARRAY *****			39
40	O070 3 000400 50E003 RCW: LL=3, NORML STATE [USER SEGMENT @ 0003:0005:2]			40
41	CODE FILE NOT AVAILABLE: (LFPROG)OBJECT/JSD/MEMDUMP/EXAMPLE ON DISK.			41
42	SEG DESC: 3 800002 B9F403			42
43	CODE: 3 A2A012 B05007 3 A6B204 9C2A03 3 B06004 A6962A >3 B220D3 A31003< 3 100483 5002B8			43
44	O000:0 NVLD FF			44
45	O000:1 NAMC 02,0006 5006			45
46	O000:3 LOAD BD			46
47	O000:4 LT8 1E(30) B21E			47
48	O001:0 INSR 42:23 9C2A17			48
49	O001:3 NAMC 02,0007 5007			49
50	O001:5 OVRD BA			50
51	O002:0 BRUN 0012:5 A2A012			51
52	O002:3 ZERO B0			52
53	O002:4 NAMC 02,0007 5007			53
54	O003:0 INDX A6			54
55	O003:1 LT8 04(4) B204			55

1	0003:3	INSR	42:3	9C2A03							1
2	0004:0	ZERO		B0							2
3	0004:1	NAMC	01,0004	6004							3
4	0004:3	INDX		A6							4
5	0004:4	BSET	42	962A							5
6	0005:0	LT8	20(32)	B220							6
7	>>>>0005:2	TWSD		D3							7
8	0005:3	EXIT		A3							8
9	0005:4	VALC	02,0003	1003							9
10	0006:0	VALC	02,0004	1004							10
11	0006:2	DIVD		83							11
12	0006:3	NAMC	02,0002	5002							12
13	0006:5	STOD		B8							13
14	0007:0	EXIT		A3							14
15	0007:1	BRUN	0008:3	A26008							15
16	0007:4	MKST		AE							16
17	0007:5	NAMC	02,0008	5008							17
18											18
19	006F	----	D[01]=>3	C08000	804005	*MSCW: PREVIOUS MSCW @ 006A; D[0]=0008 IN STACK 008					19
20											20
21	006E	(03,0004)	0	000000	00001B						21
22	006D	(03,0003)	5	EC0000	5AB239	DESC [PRESENT-COPY]: STRING (8-BIT), INDEX=5+0, READONLY (MOM @ OFFSET=0008 IN SEGDICT)					22
23	006C	(03,0002)	5	E40000	5C267B	DESC [PRESENT-COPY]: STRING (8-BIT), INDEX=5+0 (MOM @ OFFSET=001A IN THIS STACK)					23
24	006B		3	000400	80E003	RCW: LL=3, NORML STATE [USER SEGMENT @ 0003:0008:2]					24
25						SEG DESC: 3 800002 B9F403					25
26						CODE: 3 B220D3 A31003 3 100483 5002B8 3 A3A260 08AE50 >3 08ABA3 AE6005< 3 500CAF ABBEFF					26
27	006A	----	D[03]=>3	488001	40C002	*MSCW: PREVIOUS MSCW @ 0068; D[2]=0014 IN THIS STACK					27
28											28
29	0069		3	00020C	E84C88	RCW: LL=1, CNTRL STATE [MCP SEGMENT @ 0C88:00CE:1 (12908400)]					29
30						SEG DESC: 3 88000D BAF570 ----- PROCESSSOFTWAREINTERRUPTS					30
31						CODE: 3 9613B3 0A009C 3 2FOCB2 0695B4 3 6008BA AE6007 >3 ABBO60 08BAA2< 3 A003B2 4F6005					31
32	0068	----	D[03]=>3	488001	40C009	*MSCW: PREVIOUS MSCW @ 005F; D[2]=0014 IN THIS STACK					32
33											33
34	0067	(01,0008)	6	A0040C	F84C88	SCW: (BLOCK BELOW DECLARED INTERRUPTS, SNGL-DIM ARRAYS, LIBRARY TEMPLATE)					34
35	0066	(01,0007)	1	488001	40000C	SIRW: OFFSET=0020 (0014+000C) IN THIS STACK					35
36	0065	(01,0006)	0	000000	000000						36
37	0064	(01,0005)	5	C00008	34917A	DESC [PRESENT-COPY]: DATA, LENGTH=131 (MOM NOT OF THIS STACK OR SEGDICT)					37
38	0063	(01,0004)	6	000000	000000	(UNINITIALIZED VARIABLE)					38
39	0062	(01,0003)	6	000000	000000	(UNINITIALIZED VARIABLE)					39
40	0061	(01,0002)	6	000000	000000	(UNINITIALIZED VARIABLE)					40
41	0060		3	000213	C88CCE	RCW: LL=2, CNTRL STATE [MCP SEGMENT @ 0CCE:013C:1 (23992040)]					41
42						SEG DESC: 3 880017 B7BEE0 ----- INNERCEDE OF INTERCEDE					42
43						CODE: 3 1AB6B8 B040A1 3 B80019 A0613B 3 B0958F AE426B >3 ABA280 05B235< 3 95B840 CBA5B2					43
44	005F	----	D[01]=>3	C08000	804004	*MSCW: PREVIOUS MSCW @ 005B; D[0]=0008 IN STACK 008					44
45											45
46	005E	(02,0003)	0	000000	000000						46
47	005D	(02,0002)	0	000000	000002						47
48	005C		3	000828	984BCE	RCW: LL=1, CNTRL STATE [MCP SEGMENT @ 0BCE:0289:4 (24001200)]					48
49						SEG DESC: 3 A00036 C076FB ----- INTERCEDE					49
50						CODE: 3 95BCB6 BAB230 3 6003A6 B795BC 3 B09C2F 23B6BA >3 AE6006 ABA3A3< 3 BOBOBO BOBOBO					50
51	005B	----	D[02]=>3	488005	408007	*MSCW: PREVIOUS MSCW @ 0054; D[1]=0054 IN THIS STACK					51
52											52
53	005A	(01,0006)	7	088000	088CCE	PCW					53
54	0059	(01,0005)	0	005500	000004						54
55	0058	(01,0004)	6	000000	000000	(UNINITIALIZED VARIABLE)					55
56	0057	(01,0003)	5	C00008	34917A	DESC [PRESENT-COPY]: DATA, LENGTH=131 (MOM NOT OF THIS STACK OR SEGDICT)					56
57	0056	(01,0002)	0	000000	000003						57
58	0055		3	020200	F12003	RCW: LL=4, NORML STATE [USER SEGMENT @ 0003:000F:1]					58
59						SEG DESC: 3 800002 B9F403					59
60						CODE: 3 BAA240 OFAE60 3 06AE60 07AE60 3 08ABB2 15ABBO >3 ABA3AE 5009AB< 3 A3A240 OFBOBF					60

1	0054	----	D[01]=>3	C08000	804004	*MSCW: PREVIOUS MSCW @ 0050; D[0]=0008, IN STACK 008	1
2							2
3	0053	(04,0003)	0	000000	000001		3
4	0052	(04,0002)	0	000000	00003D		4
5	0051		3	020803	784CDD	RCW: LL=1, CNTRL STATE [MCP SEGMENT @ OCDD:0037:4 (24643400)]	5
6						SEG DESC: 3 A00004 807EC9 ----- FAULTHANDLER	6
7						CODE: 3 089A0C 01A040 3 36AE60 05ABA2 3 8037AE 600520 >3 032002 ABB060< 3 06BA60 04B235	7
8	0050	----	D[04]=>3	488003	61000B	*MSCW: PREVIOUS MSCW @ 0045; D[3]=0036 IN THIS STACK	8
9							9
10	004F	(01,000A)	0	000000	000036		10
11	004E	(01,0009)	0	000000	000038		11
12	004D	(01,0008)	0	024FFF	E20000		12
13	004C	(01,0007)	7	088803	784CDD	PCW	13
14	004B	(01,0006)	4	000000	320000	"ON <FAULT>" MARKER	14
15	004A	(01,0005)	1	488003	600003	SIRW: OFFSET=0039 (0036+0003) IN THIS STACK	15
16	0049	(01,0004)	0	000000	000000		16
17	0048	(01,0003)	0	000000	00003D		17
18	0047	(01,0002)	0	000000	000001		18
19	0046		3	000624	784C72	RCW: LL=1, CNTRL STATE [MCP SEGMENT @ OC72:0247:3 (11382600)]	19
20						SEG DESC: 3 A00025 D014EC ----- HARDWAREINTERRUPT68	20
21						CODE: 3 000224 308C72 3 B4AE40 392005 3 B195B8 B22595 >3 B881AB 2005B2< 3 038CA0 A24D20	21
22	0045	----	D[01]=>3	C08000	804008	*MSCW: PREVIOUS MSCW @ 003D; D[0]=0008 IN STACK 008	22
23							23
24	0044	(01,0007)	7	088224	308C72	PCW	24
25	0043	(01,0006)	0	000000	000004		25
26	0042	(01,0005)	0	000000	000001		26
27	0041	(01,0004)	0	000000	000000		27
28	0040	(01,0003)	0	9F40A0	083020		28
29	003F	(01,0002)	0	000009	000004	***** DIVIDE BY ZERO *****	29
30	003E		3	000400	60E003	RCW: LL=3, NORML STATE [USER SEGMENT @ 0003:0006:2]	30
31						SEG DESC: 3 800002 B9F403	31
32						CODE: 3 A6B204 9C2A03 3 B06004 A6962A 3 B220D3 A31003 >3 100483 5002B8< 3 A3A260 08AE50	32
33		0001:0	INSR	42:23	9C2A17		33
34		0001:3	NAMC	02,0007	5007		34
35		0001:5	OVRD		BA		35
36		0002:0	BRUN	0012:5	A2A012		36
37		0002:3	ZERO		B0		37
38		0002:4	NAMC	02,0007	5007		38
39		0003:0	INDX		A6		39
40		0003:1	LT8	04(4)	B204		40
41		0003:3	INSR	42:3	9C2A03		41
42		0004:0	ZERO		B0		42
43		0004:1	NAMC	01,0004	6004		43
44		0004:3	INDX		A6		44
45		0004:4	BSET	42	962A		45
46		0005:0	LT8	20(32)	B220		46
47		0005:2	TWSD		D3		47
48		0005:3	EXIT		A3		48
49		0005:4	VALC	02,0003	1003		49
50		0006:0	VALC	02,0004	1004		50
51	>>>>	0006:2	DIVD		83		51
52		0006:3	NAMC	02,0002	5002		52
53		0006:5	STOD		B8		53
54		0007:0	EXIT		A3		54
55		0007:1	BRUN	0008:3	A26008		55
56		0007:4	MKST		AE		56
57		0007:5	NAMC	02,0008	5008		57
58		0008:1	ENTR		AB		58
59		0008:2	EXIT		A3		59
60		0008:3	MKST		AE		60

1	0008:4	NAMC	01,0005	6005						1
2										2
3	003D	----	D[01]=>3	C08000 804003	*MSCW: PREVIOUS MSCW @ 003A; D[0]=0008	IN STACK	008			3
4										4
5	003C	(03,0002)	0	000000 000000						5
6	003B		3	000001 00E003	RCW: LL=3, NORML STATE	[USER SEGMENT @ 0003:0010:0]				6
7					SEG DESC: 3 800002 B9F403					7
8					CODE: 3 06AE60 07AE60 3 08ABB2 15ABBO 3 ABA3AE 5009AB	>3 A3A240 OFBOBF< 3 000800 C12003				8
9	003A	----	D[03]=>3	488001 40C004	*MSCW: PREVIOUS MSCW @ 0036; D[2]=0014	IN THIS STACK				9
10										10
11	0039	(03,0003)	7	088800 C12003	PCW					11
12	0038	(03,0002)	4	024FFF E20000	"ON <FAULT>" MARKER					12
13	0037		3	000806 684CD8	RCW: LL=1, CNTRL STATE	[MCP SEGMENT @ OCD8:0066:4 (24531400)]				13
14					SEG DESC: 3 880016 1FD41E ----- BLOCKEXIT					14
15					CODE: 3 6A2009 9A1401 3 A00066 AE6004 3 BOABB5 A28066	>3 AE6004 ABB234< 3 B22695 B8B225				15
16	0036	----	D[03]=>3	488001 40C012	*MSCW: PREVIOUS MSCW @ 0024; D[2]=0014	IN THIS STACK				16
17										17
18	0035	(01,0011)	6	A00A06 A84CD8	SCW: (BLOCK BELOW DECLARED INTERRUPTS, SNGL-DIM ARRAYS, LIBRARY TEMPLATE)					18
19	0034	(01,0010)	7	088000 000000	PCW					19
20	0033	(01,000F)	0	000000 000000						20
21	0032	(01,000E)	0	000000 000000						21
22	0031	(01,000D)	0	000000 000000						22
23	0030	(01,000C)	0	000000 000000						23
24	002F	(01,000B)	0	000000 000000						24
25	002E	(01,000A)	0	000000 000014						25
26	002D	(01,0009)	0	800000 204800						26
27	002C	(01,0008)	0	000000 000000						27
28	002B	(01,0007)	5	C0FFFF F338EA	DESC [PRESENT-COPY]: DATA, LENGTH=1048575 (POINTS @ OFFSET=0014	IN THIS STACK)				28
29	002A	(01,0006)	0	000000 000000						29
30	0029	(01,0005)	0	000000 000000						30
31	0028	(01,0004)	1	488001 40000E	SIRW: OFFSET=0022 (0014+000E) IN THIS STACK					31
32	0027	(01,0003)	5	C00022 8338D6	DESC [PRESENT-COPY]: DATA, LENGTH=552 (POINTS @ OFFSET=0000	IN THIS STACK)				32
33	0026	(01,0002)	0	000000 00000F						33
34	0025		3	000201 A0A003	RCW: LL=2, NORML STATE	[USER SEGMENT @ 0003:001A:1]				34
35					SEG DESC: 3 800002 B9F403					35
36					CODE: 3 B215AB 500CAF 3 BOABAE 600B50 3 OCAFAB AE600C	>3 ABA3B0 BOBOBO< 3 BEFFFF FFFFFFF				36
37	0024	----	D[01]=>3	C08000 804010	*MSCW: PREVIOUS MSCW @ 0014; D[0]=0008	IN STACK	008			37
38										38
39	0023	(02,000F)	6	800000 004800	SCW: (BLOCK BELOW DECLARED INTERRUPTS, SNGL-DIM ARRAYS)					39
40	0022	(02,000E)	7	088801 00E003	PCW					40
41	0021	(02,000D)	7	088801 00E003	PCW					41
42	0020	(02,000C)	7	888800 70E003	PCW					42
43	001F	(02,000B)	5	E10002 94917A	DESC [PRESENT-COPY]: DATA, INDEX=41, DOUBLE (MOM NOT OF THIS STACK OR SEGDICT)					43
44	001E	(02,000A)	0	000000 000000						44
45	001D	(02,0009)	7	088800 50E003	PCW					45
46	001C	(02,0008)	7	088600 20E003	PCW					46
47	001B	(02,0007)	5	C00001 EC267B	DESC [PRESENT-COPY]: DATA, LENGTH=30 (MOM @ OFFSET=001A	IN THIS STACK)				47
48	001A	(02,0006)	5	84000B 4C267B	DESC [PRESENT-MOM]: STRING (8-BIT), LENGTH=180					48
49	0019	(02,0005)	0	000000 000000						49
50	0018	(02,0004)	0	000000 000000						50
51	0017	(02,0003)	0	000000 000000						51
52	0016	(02,0002)	0	000000 000000						52
53	0015		3	000249 D88C9A	RCW: LL=2, CNTRL STATE	[MCP SEGMENT @ OC9A:049D:1 (16562000)]				53
54					SEG DESC: 3 8000B0 BD8A1E ----- BOJE0J					54
55					CODE: 3 BEFFFF FFFFFFF 3 4A6870 C30C30 3 ABA3AE 400AAB	>3 A3B234 B22695< 3 B8B180 95B9A2				55
56	0014	----	D[02]=>3	C8A000 408002	*MSCW: PREVIOUS MSCW @ 0012; D[1]=0004	IN STACK	08A			56
57										57
58	0013		3	000000 002000	RCW: DUMMY (RUN)					58
59	0012	----	D[02]=>3	F84917 A08011	*MSCW: PREVIOUS MSCW @ 0001; D[1]=917A	IN STACK	384			59
60										60

1	0011	5	270000	841742	THEFILE																1
2	0010	0	000000	000000	STRINGPOOLMOM																2
3	000F	0	000000	000000	STRINGPOOLMARK																3
4	000E	0	000000	000000	LOCKCOUNT																4
5	000D	0	000300	052000	JUNK																5
6	000C	0	000000	0011FO	BLOWBY																6
7	000B	0	000000	000000	WARNINGLIST																7
8	000A	5	000000	400000	SEGMNTARRAY																8
9	0009	0	000000	000000	CONDLISTDESC																9
10	0008	0	000000	000000	DATDESC																10
11	0007	0	000000	000000	AITDESC																11
12	0006	7	088000	000000	STACKINFOTOP																12
13	0005	5	800000	635884	OLAYINFODESC																13
14	0004	6	800000	000000	OLAYCW																14
15	0003	0	000000	000000	INTMAPLINKS																15
16	0002	0	00FFFF	F020D2	MEMLOC																16
17	0001	3	488000	000000	STUFFITMSCW																17
18	0000	0	000000	000004	T0SCWD																18
19																					19
20	SPIVECTOR[088] = 5 C00008 34917A MOM @ 21631																				20
21																					21
22	00	3	C08000	804000	PIBMSCW																22
23	01	0	000000	000001	ACTUALBOXES																23
24	02	0	000000	00008A	CODELINKS	SEG DICT = 08A															24
25	03	0	000008	2300DC	COMPILERINFO																25
26	04	0	000001	11DCDE	COREINTEGRAL	= 17947870															26
27	05	0	000000	00038C	COREINUSE	= 908															27
28	06	0	000000	00036A	COREINUSESAVED	= 874															28
29	07	0	000000	000000	FORCEDLAYTIME																29
30	08	0	000000	000107	GRAPHHEADWORD																30
31	- - - GRAPHS FOR STACK 088 - - -																				31
32	PROGRAMDUMPGRAPH:																				32
33	08A ((LFPROG)OBJECT/JSD/MEMDUMP/EXAMPLE ON DISK.)																				33
34																					34
35	09	0	000000	000000	OLAYCNTL																35
36	0A	0	000000	000001	ORIGBOXES																36
37	0B	0	000000	000000	SEARCHINFO																37
38	0C	0	000000	000008	SUBSPACEID	= 8															38
39	0D	0	000000	000000	SWAPHOLD																39
40	0E	0	400C18	CD05	TIMESTARTED	@ 34:38:13= 15.9476904 SECONDS PRIOR TO DUMP															40
41	0F-10	0	000000	000000	RUNNINGCOUNT	ACCESSCODE															41
42	11	5	800000	2B5990	CHARGECODE	= 6701															42
43	12-16	0	000000	000000	DBEQTNBLK	DBSINFO	DESTNAME	HOSTNAME	FAMILYSUB												43
44	17-19	0	000000	000000	JSTITLE	MYBDNAME	MYFPB														44
45	1A	5	800000	80E9F3	MYNAME	= (LFPROG)OBJECT/JSD/MEMDUMP/EXAMPLE															45
46	1B-1D	0	000000	000000	MYPFB	SUBSYSTEMID	YOURNAME														46
47	1E	5	800000	1C57AF	BACKUPFAMILY																47
48	O(0000) O 04C4C9 E2D200																				48
49	1F	0	000880	000000	BDINFO																49
50	20	5	800000	6100ED	EVENTCAR																50
51	O(0000) O 000000 000000 THRU 5(0005)																				51
52	21	0	000000	000000	BEDWORD																52
53	22	0	05B000	000000	BLOCK																53
54	23-24	0	000000	000000	CARDSPUNCHED	ALARMTIMES															54
55	25	0	3AFB63	EFCFD8	TIMESTAMP @ BOT	= 04/09/85 17:53:04															55
56	26	0	000000	000400	CPINFO																56
57	27	0	000000	000000	DISKUSED																57
58	28	1	48A000	400002	ENTRYPOINT	SIRW: OFFSET=0006 (0004+0002) STACK = 08A															58
59	29	2	800000	000001	EXCEPTIONEVENT	(HAPPENED, INT ATTACHED)															59
60	2A	2	000000	031907	(EXCEPTIONEVENT2)																60

1	2B-2C	0 000000 000000	EXTERNALFAMILYLINKFREEPBITS							
2	2D	0 000000 ADOOB2	HEADERACCESS							
3	2E	0 000000 000000	INSTRYINFO							
4	2F	3 020200 F12003	INTERCEPTEDRCW	LL=4, NORML [USER @ 0003:000F:1]						
5	30-32	0 000000 000000	INTERCESSION	ITINERARY			JOBDECKINFO			
6	33	0 088020 000000	JOBINFO							
7	34	5 800000 436E0D	JOBMSG							
8			0(0000) 0 000000 000000	5 800000 63C2F8						
9			0(0000) 0 002401 002100	0 00E2C5 C740C1	0 D9D9C1 E840C5	0 D9D9D6 D9407C	0 404DFO FOF0FO			
10			5(0005) 0 F1F3FO F05D5C							
11			2(0002) 0 000000 000000	THRU 3(0003)						
12	35-36	0 000000 000000	LIBRARYDIR	LINESPRINTED						
13	37	5 800001 9100B6	OLAYFILEDESC							
14			0(0000) 0 000000 000000	THRU 4(0004)						
15			5(0005) 0 000005 000320	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000			
16			10(000A) 0 000000 000000	THRU 12(000C)						
17			13(000D) 0 000000 001000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000			
18			18(0012) 0 000000 000000	THRU 19(0013)						
19			20(0014) 0 010040 435F79	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000			
20	38	0 000000 000000	PORTLIB PIB							
21	39	0 40005B 117000	PROCESSFAMILYLINK	PARENT=05B SIBLING=117						
22	3A-3E	0 000000 000000	REPRINTMASK	REMPUNCHMASK	REPLY		REPLYEVENTX	(REPLYEVENT2)		
23	3F-41	0 000000 000000	ROLLOUTINFO	SOFTINTQ	SWAPDATA					
24	42	0 000000 000001	TASKBOXES							
25	43	0 800200 000000	TASKINFO							
26	44	0 000000 000000	TASKPARAMS							
27	45	5 800001 E4EA87	PIBIOCB							
28			MLIP CNTRL: 0 10CB00 000C48	WRITE, WORD, FORWARD, PROTECT, CAUSE IO FINISH						
29			DLP ADDRESS: 0 000000 400002	MLIPNO = 4, MLIP PORT = 0, LEM PORT = 0, REL DLP = 2						
30			CMND QUEUE: 5 C00000 81D5B9							
31			SELF PTR: 5 C00001 E4EA87							
32			COMMAND PTR: 5 E00001 94EA87							
33			RESULT PTR: 5 E00001 B4EA87							
34			C/R LENGTHS: 0 000000 0A000C							
35			RESULT MASK: 0 000000 000000							
36			RSLT QUEUE: 5 E00000 10BFDA							
37			NEXT LINK: 0 000000 000000							
38			DATA PTR: 5 E00001 F0E6CE							
39			CURR LENGTH: 0 000000 000000							
40			MLIP RSLT: 0 000000 000000							
41			START TIME: 0 00063F 621FC3	17:53:19 (00:00:00.04071 PRIOR TO DUMP)						
42			FINISH TIME: 0 000000 003AC8	17:53:19 (00:00:00.00460 PRIOR TO DUMP)						
43			INFO1 WORD: 0 008088 140001	UNITNUMBER = 220, OWNER STKNO = 008, INITIATING STKNO = 088, REQUESTOR = 10 (MCPIO)						
44			INFO2 WORD: 0 0000F0 044100	STATE = 1 (INACTIVE), PATHNAME = OF, PROC NO = 4, MLIP NO = 4						
45			LOG DESC: 0 000000 000000							
46			IOCW: 0 030000 0FFAF9	IOSTANDARDFIELD: WRITE, 8-BIT, MEMORY PROTECT						
47			BUFFER DESC: 5 C00005 B0E6CE							
48			EVENT REF: 1 48800C 80000B							
49			FIB DESC: 0 000000 000000							
50			IO MASK: 0 000000 000000							
51			LOGICAL RD: 0 0001F0 000000	UNITS XFERRED = 0001F (31)						
52			IO INFO: 0 800001 F00000	WORD, BUFFERINDEX = 00000 (0), IOLENGTH = 0001F (31)						
53			COMMAND1: 0 400000 0FFAF9							
54			COMMAND2: 0 000000 000000							
55			RESULT1: 0 000000 000000							
56			RESULT2: 0 000000 000000							
57			SAVED INFO: 0 030000 0000DC							
58										
59	46-4A	0 000000 000000	TASKTYPE	HSINFO	SURROGATEINFO	WAITSTART	DEIMPRUNTIMESET			
60	4B	0 000000 000000	ACTIONQ							

1	4C	0 000000	006DF4	RECENTPROC	= 0.0675552 SECONDS				1
2	4D	7 384050	788BCA	PALACE	PCW: LL=2, D[0] SEGMENT @ OBCA:0507:0, CNTRL STATE, STACK=384 (12780000)				2
3	4E	1 784917	A0004D	PALACEREF	SIRW: OFFSET=91C7 (917A+004D) STACK = 384				3
4	4F	0 000044	000000	USAGE					4
5	50	0 000000	000000	DMSBED					5
6	51	0 000C19	334B9A	READYON	@ 34:38:29= 0.0036528 SECONDS PRIOR TO DUMP				6
7	52	0 000000	000004	PROCESSOR					7
8	53	0 000C19	334C95	CLOCKONTIME	@ 34:38:29= 0.0030504 SECONDS PRIOR TO DUMP				8
9	54	0 000000	000000	LETGEORGEDOIT					9
10	55	0 000000	00FF8E	READYTIME	= 0.1570128 SECONDS				10
11	56	0 000000	006DF4	PROCESSTIME	= 0.0675552 SECONDS				11
12	57	0 000000	017FFE	IOTIME	= 0.2359248 SECONDS				12
13	58	0 000000	000014	INITPBITKOUNT	= 20				13
14	59	0 000000	004469	INITPBITTYME	= 0.0420312 SECONDS				14
15	5A-5B	0 000000	000000	OTHERPBITCOUNT	OTHERPBITTIME				15
16	5C	0 000000	0E0404	HISTORY	REASON=14, CAUSE=4, TYPE=4				16
17	5D-5F	0 000000	000000	IOCOUNT1	IOCOUNT2 MYSTACKHISTORY				17
18	60	0 0E0400	50E003	STOPPOINT	LL=3, NORML [USER @ 0003:0005:2]				18
19	61-62	0 000000	000000	LOCKEDEVENT	(LOCKEDEVENT2)				19
20	63	0 000008	681004	VALIDITYBITS					20
21	64	0 000000	000000	AVALUE					21
22	65	0 007FFF	FFFFFF	MAXPROCESSTIME	= 549755813887				22
23	66	0 000008	000300	=>SWAPSPEX	ACTMTYPE=0, ORIGMTYPE=3				23
24	67	0 000000	00001D	CODECORE					24
25	68	0 000000	000043	DATACORE					25
26	69-6A	0 000000	000000	DISKLIMIT	ELAPSEDLIMIT				26
27	6B	0 007FFF	FFFFFF	MAXIOTIME	= 549755813887				27
28	6C-6D	0 000000	000000	MAXWAIT	WAITLIMIT				28
29	6E	0 000000	00001F	ERROR	= 31				29
30	6F	5 E00005	B093F0	EXCEPTIONTASK					30
31	70	0 000000	000070	=>OPTION					31
32	71	0 000000	000000	PARTNER					32
33	72	0 000000	4180BC	PATHCONTROL	(ORG MCS=1 LSN=188)				33
34	73	0 000000	000000	PRINTLIMIT					34
35	74	0 000000	000032	PRIORITY	= 50				35
36	75	0 010000	000000	PUNCHLIMIT	= 0				36
37	76	0 000000	000000	RESTARTCOUNT					37
38	77	0 08800A	0FOA8D	=>SERIAL					38
39	78	0 000000	001194	STACKLIMIT	= 4500				39
40	79	0 800001	200200	=>STACKSIZE	=18+512=530				40
41	7A	0 000000	0000BC	=>STATIONINFO					41
42	7B-7D	0 000000	000000	TARGETTIME	TAPECOUNT TAPEPOOL				42
43	7E	0 05B000	000000	TASKLIMIT	= 1.77088743108E+21				43
44	7F	0 0A0101	06D3C6	=>USERCODE	= LFPROG				44
45	80	0 D7D9D6	C70000	(USERCODE2)					45
46	81	0 000000	000000	(USERCODE3)					46
47	82	0 000000	000093	USERCODEPRIV	PRIVLEDGED USER, MCS ABSTAINING, HS SPECIAL PRIVS ALLOWED FOR THIS STACK, THIS STACK DID				47
48									48
49									49
50									50
51									51
52									52
53									53
54									54
55									55
56									56
57									57
58									58
59									59
60									60



1		1
2	INPUT: RELEASE	2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25	MEMORY DUMP REVIEW	25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56	APPENDIX I	56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

dump analyzer  
INPUT

0937

stack vector

1																					2			
2	INPUT: MD 2100 FOR CD																				3			
3	02100(00000)	3 408000 000007	5 880144 8B2226	5 800039 009875	1 408000 800763	5 80FFFF F00000	6 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	3			
4	02107(00007)	7 008000 084C36	7 008000 005204	5 080144 84FA40	7 008000 084CD8	7 008602 484CD2	0 000000 000000	7 008020 984BCA	4															
5	0210E(0000E)	7 008000 084C8A	7 008000 004C33	7 008000 084CDO	6 000000 000000	7 008000 384BCD	7 008000 084C89	7 008000 085288	5															
6	02115(00015)	7 008001 5051A8	7 008007 804CF6	7 0086A7 9050D3	7 008000 084D02	0 000000 000000	7 008000 004E68	7 008000 005045	6															
7	0211C(0001C)	7 408024 484BD6	0 000000 000060	7 00800A 404EC8	7 008000 005043	7 008000 004CDC	5 080001 040002	5 080004 040003	7															
8	02123(00023)	5 890004 5D8567	5 090004 04006E	5 090000 840073	5 880004 0B570D	5 880000 CC0869	7 00841C 60513B	7 0086B4 305050	8															
9	0212A(0002A)	7 0082AE 90504A	7 008A07 7050F7	7 008ACF 1050E0	7 00824D 085118	7 008000 0050D9	7 008000 004C3E	7 008000 084E25	9															
10	02131(00031)	7 008000 005200	7 008009 205209	7 00803F A84BEC	7 008000 005202	7 008000 084BE3	7 008A0A A04CFO	0 000000 000000	10															
11	02138(00038)	7 00800B 484BC2	7 008000 084CDD	7 008000 084CB9	7 008000 004E26	7 008000 004E27	7 008000 084BC0	7 008076 E84BBF	11															
12	0213F(0003F)	7 008066 884BBF	7 008000 004E51	7 408A00 184C15	7 008A0D 484BBC	7 008003 E84BE3	7 008004 684BC2	7 008000 0051AA	12															
13	02146(00046)	7 008000 004C31	7 008000 084CD1	0 0005D9 D0EF86	7 00800B B84BBD	7 008000 00512A	7 008000 084C3D	7 008816 104CF9	13															
14	0214D(0004D)	7 008000 0051A7	7 008000 005042	0 000000 014000	7 008027 684BCE	7 008005 684BBD	7 008009 C84BCE	7 008801 B84C90	14															
15	02154(00054)	7 008000 004C91	7 008000 005088	7 008404 E051FA	7 008000 C85092	7 008000 C8508F	7 008000 0051FO	7 008000 084C3B	15															
16	0215B(0005B)	7 008000 004E81	5 80000C 841510	7 008000 004C97	7 008000 084C3C	7 008800 B04E63	7 008000 004E65	7 008000 005058	16															
17	02162(00062)	7 008000 005059	7 008000 004CF3	5 080001 040007	5 880004 0FB90C	7 008A02 604CB1	7 008005 904CB3	7 008000 084BCD	17															
18	02169(00069)	7 008000 084C8E	7 008000 004CC4	7 008000 004EE3	7 008000 004EE2	7 008000 004EE5	7 008801 204EE8	7 008000 004C8F	18															
19	02170(00070)	7 008200 105044	7 408000 004FOA	7 008000 084C8B	7 008000 004EE6	7 408002 B04FO4	7 008804 004EE0	7 008403 505209	19															
20	02177(00077)	7 008000 004EE7	7 00860B F04EE1	7 008000 004E39	7 008A07 1051F3	7 008004 F84BCE	3 000003 580BCE	7 008000 084C39	20															
21	0217E(0007E)	7 008000 004C3F	7 008418 F04EC6	7 008000 0050DC	0 000000 00037D	7 008803 E04E74	7 008000 004CD5	7 008000 004E60	21															
22	02185(00085)	7 008000 005054	7 008000 004F8E	7 008000 004F88	7 008000 004F84	7 008009 C84C39	7 008000 004F55	7 008600 D04CD3	22															
23	0218C(0008C)	7 008060 F84BBF	7 008055 284BBF	7 00801E F84BCB	6 800000 000000	6 800000 000000	5 000000 000000	5 000000 000000	23															
24	02193(00093)	7 008000 004E38	7 008000 004E2B	7 00841F 985053	7 008033 A84BCA	0 000000 000000	7 00801C 984BCA	7 008045 484BCA	24															
25	0219A(0009A)	7 008046 284BCA	7 008000 005127	7 00804A 984BCA	7 008021 784BCB	7 008025 884BCB	7 008020 584BCA	0 000000 000000	25															
26	021A1(000A1)	0 000000 000000	0 000000 000000	7 00801D 304C1D	7 008000 004CC5	7 008000 004CAF	2 000000 000000	2 000000 000000	26															
27	021A8(000A8)	7 008A11 0050D1	0 000000 000003	7 40860E 884C1B	7 008000 0051F7	7 008061 00512D	7 00880E 10512D	7 008001 805135	27															
28	021AF(000AF)	7 008000 A84BCD	5 080001 04000B	5 080004 04000C	5 080004 04000F	5 080004 040012	5 080004 040015	7 008000 005174	28															
29	021B6(000B6)	7 408000 00523F	0 000000 000000	7 008000 004E5F	7 008000 084C9C	7 008000 0051E4	7 008655 D05214	7 008802 F0521A	29															
30	021BD(000BD)	5 880000 4A7430	5 080000 440019	7 008000 005235	0 000000 000000	0 000000 000000	7 008000 584BFA	7 008000 000000	30															
31	021C4(000C4)	0 000000 000000	0 000000 000000	0 000000 000000	6 000000 000000	0 000000 000000	7 008000 004C66	7 008000 00508D	31															
32	021CB(000CB)	5 800037 DP93F0	7 008000 004C67																					32
33																					33			
34																					34			
35	INPUT: MD 9875 FOR A0																				35			
36	09875(00000)	3 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	36			
37	0987C(00007)	0 000000 000000	5 800145 0020F8	5 80001C 41CF3E	5 800014 20BFFF	5 80004B 20C164	0 000000 000000	0 000000 000000	37															
38	09883(0000E)	0 000000 000000	THRU 09894(0001F)																					38
39	09895(00020)	5 800016 00A704	0 000000 000000	0 000000 000000	5 800016 00CC1C	0 000000 000000	5 80001C 40D188	0 000000 000000	39															
40	0989C(00027)	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	5 800006 30C639	0 000000 000000	40															
41	098A3(0002E)	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	5 80001C 90C972	41															
42	098AA(00035)	0 000000 000000	0 000000 000000	5 80000D 010715	0 000000 000000	5 800005 10EB31	0 000000 000000	0 000000 000000	42															
43	098B1(0003C)	0 000000 000000	THRU 098BE(00049)																					43
44	098BF(0004A)	5 800022 510A4E	5 80003C C10C77	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	44															
45	098C6(00051)	5 800075 91AFF7	0 000000 000000	0 000000 000000	5 80001B B129D8	0 000000 000000	5 80004E 2134C4	5 80004B 213999	45															
46	098CD(00058)	0 000000 000000	0 000000 000000	0 000000 000000	5 80004F 111CC6	5 80001F 018AA5	0 000000 000000	5 800086 31A202	46															
47	098D4(0005F)	0 000000 000000	0 000000 000000	0 000000 000000	5 80004B 21F2F5	5 80004B 21F8CA	0 000000 000000	0 000000 000000	47															
48	098DB(00066)	0 000000 000000	0 000000 000000	5 800058 821D39	5 80003C C22578	0 000000 000000	0 000000 000000	0 000000 000000	48															
49	098E2(0006D)	0 000000 000000	5 800058 811066	5 80001C 923D00	5 80003D 619CBB	0 000000 000000	5 800021 61E735	5 800006 B26351	49															
50	098E9(00074)	5 800032 A44F95	0 000000 000000	5 800003 0266F9	5 800046 2281A9	5 800016 A26ADC	5 800020 D26E15	5 800004 D27026	50															
51	098FO(0007B)	5 80000F A26447	0 000000 000000	5 800009 E27281	5 800004 B26633	0 000000 000000	5 800058 82C905	0 000000 000000	51															
52	098F7(00082)	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	5 800022 8338D6	52															
53	098FE(00089)	5 80004B 23E396	5 800001 35DAB3	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	0 000000 000000	53															
54	09905(00090)	5 800005 9300DF	5 800006 230745	5 800003 E30AF9	0 000000 000000	0 000000 000000	0 000000 000000	5 80001C 42E554	54															
55	0990C(00097)	0 000000 000000	THRU 09914(0009F)																					55
56																					56			
57																					57			
58	INPUT: MD 338D6 FOR 100																				58			
59	338D6(00000)	0 000000 000004	3 488000 000000	0 00FFFF F020D2	0 000000 000000	6 800000 000000	5 800000 635884	7 088000 000000	59															
60	338DD(00007)	0 000000 000000	0 000000 000000	0 000000 000000	5 000000 400000	0 000000 000000	0 000000 0011F0	0 000300 052000	60															

stack vector

P18 vector 093F0

not log 3

msid

in process stack

1	338E4(0000E)	0	000000	000000	0	000000	000000	0	000000	000000	5	270000	841742	3	F84917	A08011	3	000000	002000	3	C8A000	408002	1
2	338EB(00015)	3	000249	D88C9A	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	84000B	4C267B	5	C00001	EC267B	2
3	338F2(0001C)	7	088600	20E003	7	088800	50E003	0	000000	000000	5	E10002	94917A	7	888800	70E003	7	088801	00E003	7	088801	00E003	3
4	338F9(00023)	6	800000	004800	3	C08000	804010	3	000201	A0A003	0	000000	00000F	5	C00022	8338D6	1	488001	00000E	0	000000	000000	4
5	33900(0002A)	0	000000	000000	5	C0FFFF	F338EA	0	000000	000000	0	800000	204800	0	000000	000014	0	000000	000000	0	000000	000000	5
6	33907(00031)	0	000000	000000	0	000000	000000	0	000000	000000	7	088000	000000	6	A00A06	A84CD8	3	488001	40C012	3	000806	684CD8	6
7	3390E(00038)	4	024FFF	E20000	7	088800	C12003	3	488001	40C004	3	000001	00E003	0	000000	000000	3	C08000	804003	3	000400	60E003	7
8	33915(0003F)	0	000000	000004	0	9F40A0	83020	0	000000	000000	0	000000	000001	0	000000	000004	7	088224	308C72	3	C08000	804008	8
9	3391C(00046)	3	000624	784C72	0	000000	000001	0	000000	00003D	0	000000	000000	1	488003	600003	4	000000	320000	7	088803	784CDD	9
10	33923(0004D)	0	024FFF	E20000	0	000000	000038	0	000000	000036	3	488003	61000B	3	020803	784CDD	0	000000	00003D	0	000000	000001	10
11	3392A(00054)	3	C08000	804004	3	020200	F12003	0	000000	000003	5	C00008	34917A	6	000000	000000	0	005500	000004	7	088000	088CCE	11
12	33931(0005B)	3	488005	408007	3	000828	984BCE	0	000000	000002	0	000000	000000	3	C08000	804004	3	000213	C88CCE	6	000000	000000	12
13	33938(00062)	6	000000	000000	6	000000	000000	5	C00008	34917A	0	000000	000000	1	488001	40000C	6	A0040C	F84C88	3	488001	40C009	13
14	3393F(00069)	3	00020C	E84C88	3	488001	40C002	3	000400	80E003	5	E40000	5C267B	5	EC0000	5AB239	0	000000	00001B	3	C08000	804005	14
15	33946(00070)	3	000400	50E003	0	000009	000400	0	AB23E0	0D3038	0	000000	000003	0	000000	00000E	0	000000	000004	7	088224	308C72	15
16	3394D(00077)	3	C08000	804008	3	000A25	B84C72	0	000000	000014	0	FFFFFF	FFFFFF	0	000000	000000	0	000000	00033A	0	000000	00007F	16
17	33954(0007E)	0	000000	000000	5	C00034	7338D6	3	C08000	804009	3	000807	184CB9	0	CD5A64	665473	1	570A55	C95CFO	0	000000	014C6B	17
18	3395B(00085)	0	00063F	626206	0	0800DC	000002	0	000000	000000	0	000000	000260	0	000000	00027E	0	000000	001FCA	0	000000	00095D	18
19	33962(0008C)	0	000000	000004	0	000000	000000	0	400000	4E90AD	0	000000	00DF53	0	000000	000000	0	000000	000000	0	000000	000000	19
20	33969(00093)	0	000000	000000	5	C0000A	41D6E1	7	088400	289288	7	088400	C89288	7	088A04	789288	3	000102	80E070	20	000000	000000	1st Card
21	33970(0009A)	3	000006	40DF53	5	C0005D	E0095D	3	C888FB	F0401C	3	020A34	885288	5	C0001D	20A917	5	C00000	833956	5	C0005D	E0095D	21
22	33977(000A1)	0	000000	00027E	0	000000	0007CD	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	00000A	22
23	3397E(000A8)	0	000000	000000	3	000807	184CB9	0	000000	000088	0	000000	002100	0	000000	000088	0	000000	0338D6	0	000000	033972	23
24	33985(000AF)	3	C08000	80401C	0	000000	033AFE	0	000000	000034	5	E03397	300000	0	000000	00002D	0	000000	000004	0	000000	000000	24
25	3398C(000B6)	0	000000	000010	0	000000	000004	0	000000	000000	0	000000	000000	3	C08000	80400B	3	000001	C85287	5	C00015	30A996	25
26	33993(000BD)	0	000000	000000	0	000000	000000	5	C00000	80A9A5	6	000000	000000	0	000000	000000	0	000000	000000	0	000000	000003	26
27	3399A(000C4)	0	000000	000008	6	000000	000000	0	000000	000000	1	000000	002004	0	000000	000003	0	FE0000	000000	0	000000	00002D	27
28	339A1(000CB)	0	000000	00000A	0	000000	000000	0	030000	0FFAF9	5	C00005	BOE6CE	0	000000	000000	0	000000	00001F	0	000000	000001	28
29	339A8(000D2)	0	000000	000000	2	000000	000001	2	000000	000000	3	C08000	80400D	3	200238	D04BD1	1	48800C	80000B	0	000000	000000	29
30	339AF(000D9)	0	000000	000000	0	000000	000000	3	F84917	A08006	3	200820	484BCA	0	000000	000000	0	000C19	334C03	0	000000	000000	30
31	339B6(000E0)	0	000000	0000FB	0	000000	000000	5	E00000	30A64A	0	000000	0000FB	0	000000	000004	0	00063F	643E18	0	000000	000057	31
32	339BD(000E7)	0	000000	000001	5	C00008	30FA93	0	000000	000000	5	E00000	408C1D	0	000000	000057	0	000000	000057	0	000000	000008	32
33	339C4(000EE)	0	000000	000020	5	E00000	808CDB	0	000000	000000	0	000000	000000	0	400000	000000	5	E00001	D4EA87	0	000000	000000	33
34	339CB(000F5)	0	000000	000000	0	400000	000000	0	000000	000006	0	000000	000000	0	000000	000000	0	000000	0020D2	0	000000	0AE2D9	34
35	339D2(000FC)	0	000000	0A6FDF	0	000000	0C0454	5	EOAE2D	400000	5	COFFFF	FO0000										35
36																							36
37																							37
38	INPUT: MD 20F8 FOR A6																						38
39	020F8(00000)	3	000145	001448	3	408000	000000	0	00FFFF	F020D2	0	000000	000000	6	800000	000000	5	800001	A4353D	0	000000	000000	39
40	020FF(00007)	0	000000	000000	3	408000	000007	5	880144	8B2226	5	800039	009875	1	408000	800763	5	80FFFF	F00000	6	000000	000000	40
41	02106(0000E)	0	000000	000000	7	008000	084C36	7	008000	005204	5	080144	84FA40	7	008000	084CD8	7	008602	484CD2	0	000000	000000	41
42	0210D(00015)	7	008020	984BCA	7	008000	084C8A	7	008000	004C33	7	008000	084CD0	6	000000	000000	7	008000	384BCD	7	008000	084C89	42
43	02114(0001C)	7	008000	085288	7	008001	5051A8	7	008007	804CF6	7	0086A7	9050D3	7	008000	084D02	0	000000	000000	7	008000	004E68	43
44	0211B(00023)	7	008000	005045	7	408024	484BD6	0	000000	000060	7	00800A	404EC8	7	008000	005043	7	008000	004CDC	5	080001	040002	44
45	02122(0002A)	5	080004	040003	5	890004	5D8567	5	090004	04006E	5	090000	840073	5	880004	0B570D	5	880000	CC0869	7	00841C	60513B	45
46	02129(00031)	7	0086B4	305050	7	0082AE	90504A	7	008A07	7050F7	7	008ACF	1050E0	7	00824D	085118	7	008000	0050D9	7	008000	004C3E	46
47	02130(00038)	7	008000	084E25	7	008000	005200	7	008009	205209	7	00803F	A84BBC	7	008000	005202	7	008000	084BE3	7	008A0A	A04CFO	47
48	02137(0003F)	0	000000	000000	7	00800B	484BC2	7	008000	084CDD	7	008000	084CB9	7	008000	004E26	7	008000	004E27	7	008000	084BC0	48
49	0213E(00046)	7	008076	E84BBF	7	008066	884BBF	7	008000	004E51	7	408A00	184C15	7	008A0D	484BBC	7	008003	E84BE3	7	008004	684BC2	49
50	02145(0004D)	7	008000	0051AA	7	008000	004C31	7	008000	084CD1	0	0005D9	DOEF86	7	00800B	B84BBD	7	008000	00512A	7	008000	084C3D	50
51	0214C(00054)	7	008816	104CF9	7	008000	0051A7	7	008000	005042	0	000000	014000	7	008027	684BCE	7	008005	684BBD	7	008009	C84BCE	51
52	02153(0005B)	7	008801	B84C90	7	008000	004C91	7	008000	005088	7	008404	E051FA	7	008000	C85092	7	008000	C8508F	7	008000	0051FO	52
53	0215A(00062)	7	008000	084C3B	7	008000	004E81	5	80000C	841510	7	008000	004C97	7	008000	084C3C	7	008800	B04E63	7	008000	004E65	53
54	02161(00069)	7	008000	005058	7	008000	005059	7	008000	004CF3	5	080001	040007	5	880004	0FB90C	7	008A02	604CB1	7	008005	904CB3	54
55	02168(00070)	7	008000	084BCD	7	008000	084C8E	7	008000	004CC4	7	008000	004EE3	7	008000	004EE2	7	008000	004EE5	7	008801	204EE8	55
56	0216F(00077)	7	008000	004C8F	7	008200	105044	7	408000	004FOA	7	008000	084C8B	7	008000	004EE6	7	408002	B04FO4	7	008804	004EE0	56



1	02192(0009A)	5	000000	000000	7	008000	004E38	7	008000	004E2B	7	00841F	985053	7	008033	A84BCA	0	000000	000000	7	00801C	984BCA	1
2	02199(000A1)	7	008045	484BCA	7	008046	284BCA	7	008000	005127	7	00804A	984BCA	7	008021	784BCB							2
3																							3
4																							4
5	INPUT: MD 5DAB3 FOR 13 <i>stack 8A - on seg dictionary</i>																						
6	5DAB3(00000)	3	000001	28400E	3	48A000	000000	0	00FFFF	F020D2	0	000000	000000	3	C08000	804003	5	880000	DA3335	7	08A401	AOA003	5
7	5DABA(00007)	3	800002	80F403	5	880000	5AB239	1	408000	80009D	1	408000	800098	1	408000	800066	1	408000	800068	1	408000	800099	7
8	5DAC1(0000E)	1	408000	80000E	1	408000	80009E	1	408000	80000A	5	070000	00008C	5	070000	000093							8
9																							9
10																							10
11	INPUT: MD 9F403 FOR 2B <i>seg 3</i>																						
12	9F403(00000)	3	FF5006	BDB21E	3	9C2A17	5007BA	3	A2A012	B05007	3	AGB204	9C2A03	3	Bd6004	Ad962A	3	(B220D3)	A31003	3	100483	5002B8	11
13	9F40A(00007)	3	A3A260	08AE50	3	08ABA3	AE6005	3	500CAF	ABBEFF	3	024FFF	E20000	3	B20495	B47002	3	BAA240	OFAE60	3	06AE60	07AE60	12
14	9F411(0000E)	3	08ABB2	15ABB0	3	ABA3AE	5009AB	3	A3A240	OFBOBF	3	000800	C12003	3	B4A260	08FFAE	3	6009AE	6007AE	3	6008AB	B215AB	13
15	9F418(00015)	3	BOABAE	600AAE	3	6007AE	6008AB	3	B215AB	500CAF	3	BOABAE	600B50	3	0CAFAB	AE600C	3	ABA3B0	BOBOB0	3	BEFFFF	FFFFF	14
16	9F41F(0001C)	3	04000B	400000	3	B20595	B4BOBF	3	000600	20E003	3	BFFFFF	FFFFF	3	000800	50E003	3	BOBOBF	FFFFF	3	000800	70E003	15
17	9F426(00023)	3	BFFFFF	FFFFF	3	000801	00E003	3	BFFFFF	FFFFF	3	000801	00E003	3	B4BEFF	FFFFF	3	800000	204800	3	B20695	B4A220	16
18	9F42D(0002A)	3	00FFFF	FFFFF																		17	
19																							18
20																							19
21	INPUT: MD 93F0 FOR 37D <i>PIB vector Cdex CB</i>																						
22	093F0(00000)	3	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	20
23	093F7(00007)	0	000000	000000	5	800008	309C09	5	800008	30BA72	5	800008	31D32F	5	800008	31D3B6	0	000000	000000	0	000000	000000	21
24	093FE(0000E)	0	000000	000000	THRU 0940F(0001F)																		22
25	09410(00020)	5	800008	30A675	0	000000	000000	0	000000	000000	5	800008	30CB95	0	000000	000000	5	800008	30CEEC	0	000000	000000	23
26	09417(00027)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	800008	30E851	0	000000	000000	24
27	0941E(0002E)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	800001	00D161	25
28	09425(00035)	0	000000	000000	0	000000	000000	5	800008	3102C2	0	000000	000000	5	800001	00D92B	0	000000	000000	0	000000	000000	26
29	0942C(0003C)	0	000000	000000	THRU 09439(00049)																		27
30	0943A(0004A)	5	800001	00E3AC	5	800008	30F4E7	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	28
31	09441(00051)	5	800008	30F073	0	000000	000000	0	000000	000000	5	C00008	3101A1	0	000000	000000	5	800008	30FA0C	5	800008	30FA93	29
32	09448(00058)	0	000000	000000	0	000000	000000	0	000000	000000	5	800008	319A31	5	800001	00FDFF	0	000000	000000	5	800001	01C851	30
33	0944F(0005F)	0	000000	000000	0	000000	000000	0	000000	000000	5	C00008	31B788	5	C00008	31B831	0	000000	000000	0	000000	000000	31
34	09456(00066)	0	000000	000000	0	000000	000000	5	C00008	32164C	5	800008	3224F1	0	000000	000000	0	000000	000000	0	000000	000000	32
35	0945D(0006D)	0	000000	000000	5	C00008	3107E9	5	800008	30F3D4	5	800001	01C7D3	0	000000	000000	5	C00008	310349	5	C00008	3244BA	33
36	09464(00074)	5	C00008	3147CF	0	000000	000000	5	C00008	3242D3	5	C00008	31EC3F	5	C00008	324427	5	800001	0243E8	5	800008	3263C0	34
37	0946B(0007B)	5	800001	026D8F	0	000000	000000	5	800008	3271FA	5	800001	02661F	0	000000	000000	5	C00008	327C7F	0	000000	000000	35
38	09472(00082)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	C00008	34917A	36
39	09479(00089)	5	C00008	3417DD	5	800001	0183D7	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	37
40	09480(00090)	5	800001	0300CB	5	800008	3306BE	5	800001	030AE5	0	000000	000000	0	000000	000000	0	000000	000000	5	800008	33FEB4	38
41	09487(00097)	0	000000	000000	THRU 09498(000A8)																		39
42	09499(000A9)	5	800008	32B636	5	800001	02DFF6	5	800008	310414	5	800001	030AB7	0	000000	000000	0	000000	000000	0	000000	000000	40
43	094A0(000B0)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	800008	32AF97	5	800001	0368E9	5	C00008	335486	41
44	094A7(000B7)	0	000000	000000	0	000000	000000	5	800001	03691F	5	C00008	32B552	5	C00008	3291D3	5	800008	3383C2	5	800001	036DC0	42
45	094AE(000BE)	0	000000	000000	THRU 09506(00116)																		43
46	09507(00117)	5	C00008	34D34D	5	C00008	33560D	5	800008	341CE5	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	44
47	0950E(0011E)	0	000000	000000	THRU 09526(00136)																		45
48	09527(00137)	5	800008	33E275	5	800001	02C3F1	5	C00008	30FF57	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	46
49	0952E(0013E)	0	000000	000000	THRU 0953C(0014C)																		47
50	0953D(0014D)	5	800008	31264C	5	800001	0529D5	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	48
51	09544(00154)	0	000000	000000	THRU 09553(00163)																		49
52	09554(00164)	5	800008	341B6A	5	800001	02D5E0	5	C00008	356A05	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	50
53	0955B(0016B)	5	C00008	340448	5	800001	05CC4E	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	51
54	09562(00172)	0	000000	000000	THRU 095D1(001E1)																		52
55	095D2(001E2)	5	800008	317543	5	800001	04E626	5	800008	34F5E5	5	800008	340FE6	5	800001	02D6B3	0	000000	000000	0	000000	000000	53
56	095D9(001E9)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	800008	3346A9	5	800001	0510F5	54
57	095E0(001F0)	5	C00008	33300E	0	000000	000000	0	000000	000000	5	C00008	340AE1	5	800008	3334DE	0	000000	000000	0	000000	000000	55
58	095E7(001F7)	0	000000	000000	THRU 096D3(002E3)																		56
59	096D4(002E4)	5	C00008	3492CF	5	C00008	35A2EC	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	57
60	096DE(002EB)	0	000000	000000	THRU 0971E(0032E)																		58

PIB  
for  
88

1	0971F(0032F)	5	C00008	35CA22	5	C00008	34EAB5	5	800008	342555	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	1
2	09726(00336)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	800008	30E647	5	800001	02FE18	0	000000	000000	2
3	0972D(0033D)	0	000000	000000	THRU 0976C(0037C)																			3		
4	<i>my P/B take attributes</i>																									
5	<i>Backup pool</i>																									
6	<i>msg</i>																									
6	INPUT: MD 4917A FOR 83																									
7	4917A(00000)	3	C08000	804000	0	000000	000001	0	000000	00008A	0	000008	2300DC	0	000001	11DCDE	0	000000	000380	0	000000	00036A	0	000000	000000	7
8	49181(00007)	0	000000	000000	0	000000	000107	0	000000	000000	0	000000	000001	0	000000	000000	0	000000	000008	0	000000	000000	0	000000	000000	8
9	49188(0000E)	0	400C18	CDED05	0	000000	000000	0	000000	000000	5	800000	2B5990	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	9
10	4918F(00015)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	5	800000	80E9F3	0	000000	000000	0	000000	000000	10
11	49196(0001C)	0	000000	000000	0	000000	000000	5	800000	1C57AF	0	000880	000000	5	800000	6100ED	0	000000	000000	0	05B000	000000	0	000000	000000	11
12	4919D(00023)	0	000000	000000	0	000000	000000	0	3AFB63	EFCFDS	0	000000	000400	0	000000	000000	1	48A000	400002	2	800000	000001	0	000000	000000	12
13	491A4(0002A)	2	000000	031907	0	000000	000000	0	000000	000000	0	000000	AD00B2	0	000000	000000	3	020200	F12003	0	000000	000000	0	000000	000000	13
14	491AB(00031)	0	000000	000000	0	000000	000000	0	088020	000000	5	800000	436E0D	0	000000	000000	0	000000	000000	5	800001	9100B6	0	000000	000000	14
15	491B2(00038)	0	000000	000000	0	40005B	117000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	15
16	491B9(0003F)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000001	0	800200	000000	0	000000	000000	5	800001	E4EA87	0	000000	000000	16
17	491C0(00046)	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	17
18	491C7(0004D)	7	384050	788BCA	1	784917	A0004D	0	000044	000000	0	000000	000000	0	000019	334B9A	0	000000	000004	0	000019	334C95	0	000000	000000	18
19	491CE(00054)	0	000000	000000	0	000000	00FF8E	0	000000	006DF4	0	000000	017FFE	0	000000	000014	0	000000	004469	0	000000	000000	0	000000	000000	19
20	491D5(0005B)	0	000000	000000	0	000000	0E0404	0	000000	000000	0	000000	000000	0	000000	000000	0	0E0400	50E003	0	000000	000000	0	000000	000000	20
21	491DC(00062)	0	000000	000000	0	000008	681004	0	000000	000000	0	007FFF	FFFFFF	0	000008	000300	0	000000	00001D	0	000000	000043	0	000000	000000	21
22	491E3(00069)	0	000000	000000	0	000000	000000	0	007FFF	FFFFFF	0	000000	000000	0	000000	000000	0	000000	00001F	5	E00005	B093F0	0	000000	000000	22
23	491EA(00070)	0	000000	000070	0	000000	000000	0	000000	4180BC	0	000000	000000	0	000000	000032	0	010000	000000	0	000000	000000	0	000000	000000	23
24	491F1(00077)	0	08800A	OFOABD	0	000000	001194	0	800001	200200	0	000000	0000BC	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	24
25	491F8(0007E)	0	05B000	000000	0	0A0101	06D3C6	0	D7D9D6	C70000	0	000000	000000	0	000000	000093	0	000000	000000	0	000000	000000	0	000000	000000	25
26																										
27																										
28	INPUT: MD 36E0D FOR 4																									
29	36E0D(00000)	0	000000	000000	5	800000	63C2F8	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	0	000000	000000	29
30																										
31																										
32	INPUT: MD C57AF FOR 1																									
33	C57AF(00000)	0	04C4C9	E2D200	<i>message pool</i>																			33		
34																										
35																										
36	INPUT: MD E9F3 FOR 8																									
37	OE9F3(00000)	0	250305	06D3C6	0	D7D9D6	C706D6	0	C2D1C5	C3E303	0	D1E2C4	07D4C5	0	D4C4E4	D4D707	0	C5E7C1	D4D7D3	0	C50000	000000	0	000000	000000	37
38	OE9FA(00007)	0	FEDFED	FEDFED																				38		
39																										
40																										
41	INPUT: MD 3C2F8 FOR 6																									
42	3C2F8(00000)	0	002401	002100	0	00E2C5	C740C1	0	D9D9C1	E840C5	0	D9D9D6	D9407C	0	404DF0	FOFOFO	0	F1F3F0	FO5D5C						42	
43																										
44																										
45	INPUT: RELEASE																									
46																										
47																										
48																										
49																										
50																										
51																										
52																										
53																										
54																										
55																										
56																										
57																										
58																										
59																										
60																										

*See P 214  
for dump analysis*

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60

1		1
2		2
3		3
4		4
5		5
6		6
7		7
8		8
9		9
10		10
11		11
12		12
13		13
14		14
15		15
16		16
17		17
18		18
19		19
20		20
21		21
22		22
23		23
24		24
25		25
26		26
27		27
28		28
29		29
30		30
31		31
32		32
33		33
34		34
35		35
36		36
37		37
38		38
39		39
40		40
41		41
42		42
43		43
44		44
45		45
46		46
47		47
48		48
49		49
50		50
51		51
52		52
53		53
54		54
55		55
56		56
57		57
58		58
59		59
60		60