PART 2 DRTM CONFIGURATION

The Disc Real Time Monitor consists of a number of modules, from which the user can select those that are required for his application and configure his own monitor. The main parts of the monitor are:

- Nucleus, consisting of monitor modules running at level 48
- System Interrupt Modules, running at levels between 0 and 47
- System Programs, running at levels between 49 and 63, of which the System Command Language (SCL) module is the most important one.

All modules are centered around the dispatcher, a monitor routine which determines which routine is to be executed next, on the basis of priority levels.

NUCLEUS

The nucleus of the DRTM consists of a set of monitor modules which must be memory resident and run at level 48, and of a number of system tables.

Usually, the modules at level 48 are a continuation of an interrupt at a higher priority level: an interrupt is accepted at e.g. level 5, a branch is made to a module at level 48 and there the interrupt is processed. This enables other interrupts to be accepted again.

Monitor Modules

Monitor modules running at level 48 are the dispatcher and a number of service routines to handle certain monitor requests. The dispatcher's function is to decide which is the next routine or program which must be executed.
The monitor request handlers perform the following functions:

- Physical I/O (LKM1)
- Activate (LKM12)
- Exit (LKM3)

- Set Event (LKM18)
- Get Buffer/Release Buffer (LKM 4/5)
- Wait for an Event (LKM2)
- Switch inside a Program (LKM13)

They all run at level 48 and thus make use of the A15 stack just as interrupt routines.

System Tables

The following tables are part of the DRTM nucleus:

- T:CVT, Communication Vector Table, contains parameters concerning the memory configuration, such as memory size, stack base address, dispatcher address, etc.
- T:LKM, monitor request table, containing the addresses of the monitor request handlers.
- T:RMAC, contains the addresses of the memory resident monitor request handlers.
- T:PCT, Program Control Table, is actually a pool of PCT's, one for each user program. Each table consist of 18 words of information relevant to the program to which it refers.
- T:SLT, Software Level Table, consist of 15 entries, corresponding to priority levels 40 to 63. It is used by the dispatcher to find the current PCT connected to a given level.
- T:FCT, File Code Table, contains the links between logical file codes and the disc files or peripheral devices to which they have been assigned.
- T:SWP, containing the PCT addresses of swappable programs.
- DWT, Device Work Table, one for each peripheral device, containing the necessary parameters on the device and its logical handling.
- T:DCT, Disc Control Table, supplements the DWT for the disc I/O driver and facilitates communication between Data Management, disc driver and monitor. There is one DCT for each disc and DAD.
- T:LFT, Logical File Description Table, contains the information necessary for an I/O operation on a logical file. One table is assigned to each file.
- Timer Management Tables.
- T:DP (Pool of DAD Control Tables (DADCT))
- T:BTP (Pool of Bit Tables (BTB))

SYSTEM INTERRUPT MODULES

This set of modules consists of a number of routines which service certain hardware interrupts. The set can be extended with user-written interrupt routines.

- Power Failure Routine (I:PFAR)handles the power failure interrupt and normally operates on the highest priority level (0).
- LKM Handler (I:LKM) handles the LKM interrupt when a monitor request is given, then branches to the requested handler. Also handles the interrupt which occurs when an illegal instruction code is used. Normally operates on priority level 1.

- Real Time Clock Interrupt Routine (I:RTC), which handles the real time clock.
- I/O interrupt routines, which handle the interrupts coming from the various standard peripheral devices.
- User interrupt routines processing the interrupt from external user devices.

Note: Not-recognized interrupts, i.e. interrupts for which no routine has been included at system generation time or loaded at initialization time, will cause a system hang-up.

SYSTEM PROGRAMS

Various programs, running at levels between 49 and 63, may be included in the system. They may be either memory resident or read only.

System Loader

Used to load memory resident, read only, swappable or background programs and interrupt routines. For the read only, swappable and background programs the loader is activated by the dispatcher. For memory resident programs it is called by the System Command Language.

System Command Language (SCL)

To process the various DRTM control commands, different routines may be activated. These routines are read only and need not be memory resident. They can be connected to any level from 49 to 61 in case file code /EC (control command input) is assigned to the typewriter. When /EC is assigned to another device, the SCL routines must be connected to a level between that of D:OCOM +1 (see below) and 61. Output from the SCL takes place on file code /EF, which must always be the typewriter.

Control Panel Programs

A number of programs is required to process the control panel interrupt and the operator commands. These programs may be connected to any level between 49 and 61, but the choice of level must be made very carefully:

- D:CTPN reads the operator commands and activates the routine required to process a specific command.
  D:CTPN must be memory resident and it is recommended to connect it to a high level, i.e. 49. This will not be a serious objection because it is a relatively short program and its main function is to read an operator command with 'implicit wait'.
- D:OCOM processes various commands, especially RY and RD, and prints error messages. It is recommended to use it as a read only program.
  It may be connected to any level from 49 to 60, but if a higher-priority read only program is running, D:OCOM will not get control. Thus the system may be locked, if such a program accidentally requires operator intervention as a result of an error upon a request for I/O, since D:OCOM may not be loaded.
  Therefore, D:OCOM must be connected to a priority level higher than those of all other read only programs requesting I/O on devices on which retry operations are possible. For example, if D:OCOM is connected to level 55, read only programs at levels 49 to 55 must not use devices such as line printer or punched tape equipment, but disc files only.

For these reasons, it is recommended to connect D:OCOM to priority level 49.

- D:DUMP is an optional program which is used in the system when the dump feature is required. It is a read only program. Since it makes use of the line printer (on which retry operations are possible) it must be connected to a level between that of D:OCOM +1 and 61.

Data Management

This program performs the logical input/output for disc devices. It may be connected to any level between 49 and 61 and can be used by any user program. If the priority level of the user program is higher than that of Data Management, the request is recorded in the Activate queue and serviced as soon as Data Management becomes the program with the highest priority in the system.
The Data Management program must be memory resident and it is recommended to connect it to level 49.
The system itself does not make use of disc logical I/O, so if the user wishes he can remove this module from the monitor to save memory space.

User Service Calls

These are routines handling certain monitor requests made by user programs, such as Assign File Code, Delete File Code, Connect a Program to a Level, etc.
They may be memory resident or read only and can be connected to any level from 49 to 61. It is recommended to use level 49.
If they are memory resident, they can be called by any user program. If they are read only, however, they can be called only by user programs of a lower priority level.
At system generation time, the user can select the monitor request handlers which he requires for his programs. They may be individually declared as read only or memory resident.
Giving a monitor request of which the corresponding request handler has not been included, may result in system hang-up.

Disc Allocation Program

The disc allocation program, which may be connected to any level from 49 to 61, handles the allocation and deallocation of granules for disc files.
It must be memory resident and will be connected to level 49.
The Disc Allocation Program is called by Data Management and by the monitor requests Assign File Code and Delete File Code. This implies that, if it is declared as a read only program, it must be of a higher priority than those programs.

Time Handler

This program is activated by the real time clock and handles all the programs connected to this clock or to one of the timers. It must be declared memory resident, because it is activated with every real time clock interrupt.

Note: For all system read only programs, the start address must be the first word of the program.

DISPATCHER

All monitor modules are centered around the dispatcher, a module which runs at level 48 and divides central processor time between programs according to their priority level.

When an interrupt has been handled, the dispatcher determines which program must be started and prepares its activation by loading its start address and register contents from the save area (a 16-word area in front of the program or in the Read Only Save Area).
For hardware interrupt levels and level 48, control will be returned to the interrupted routine. For software levels (>48), the dispatcher will compare the priority levels of the programs which are active and not waiting for an event, in order to find the one with the highest priority. The dispatcher finds this information in the Program Control Tables, the addresses of which it looks up in the Software Level Table. If the highest priority program is not the interrupted program, it will receive control after the relevant data of the interrupted program (P-register, PSW, registers A1 to A14) have been stored in its save area.

If the interrupted program was also the one with the current highest priority, it will be restarted.

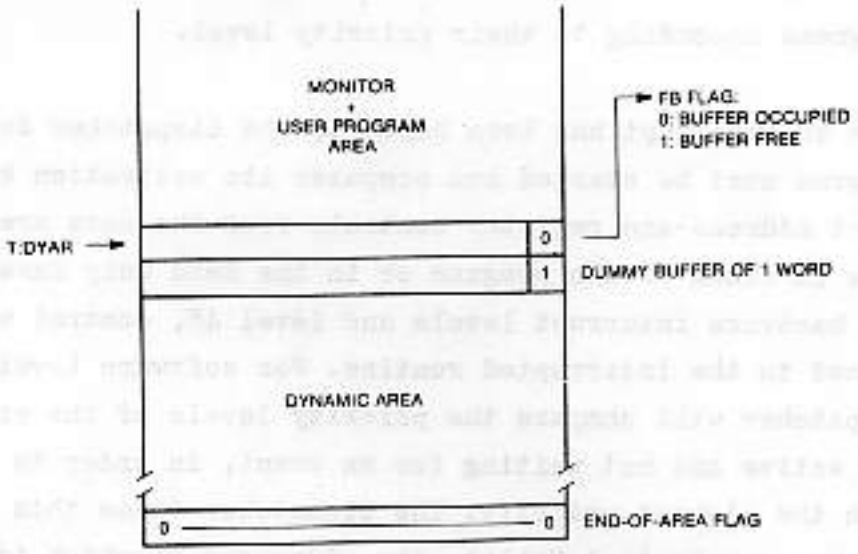
MEMORY ORGANIZATION

The memory layout has already been described in Part 1, Chapter 2. In this section the internal management of the various partitions will be discussed.

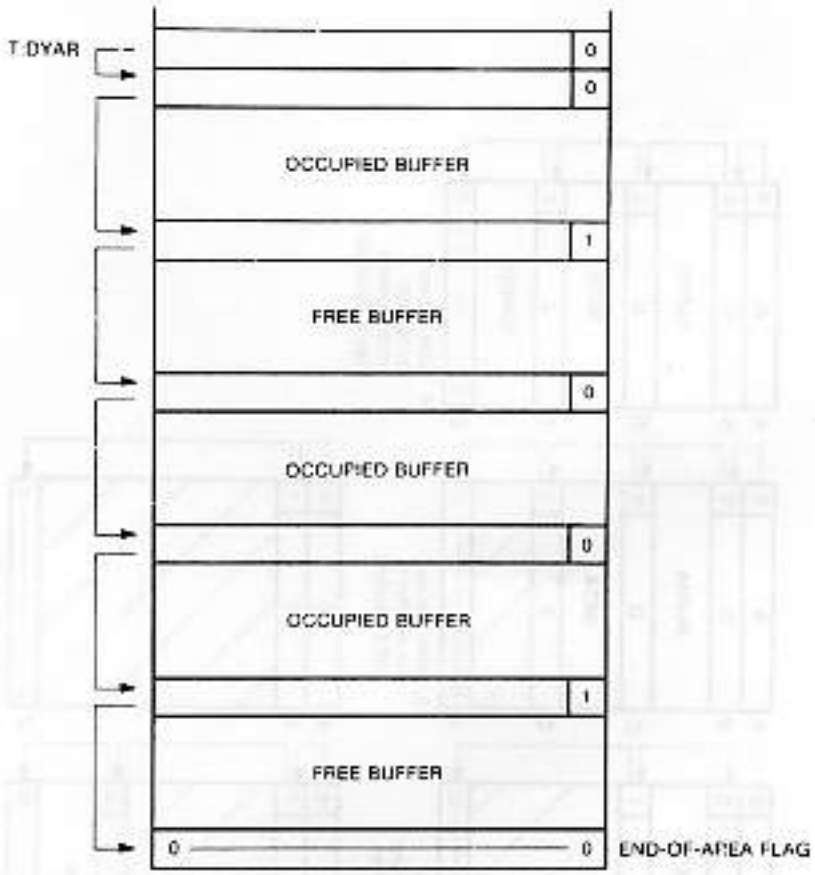
Dynamic Allocation Area

The area of memory which remains after the Initial Program Loader has loaded the monitor and user programs, is formatted by INIMON as are all other memory paritions. The minimum size of the area,

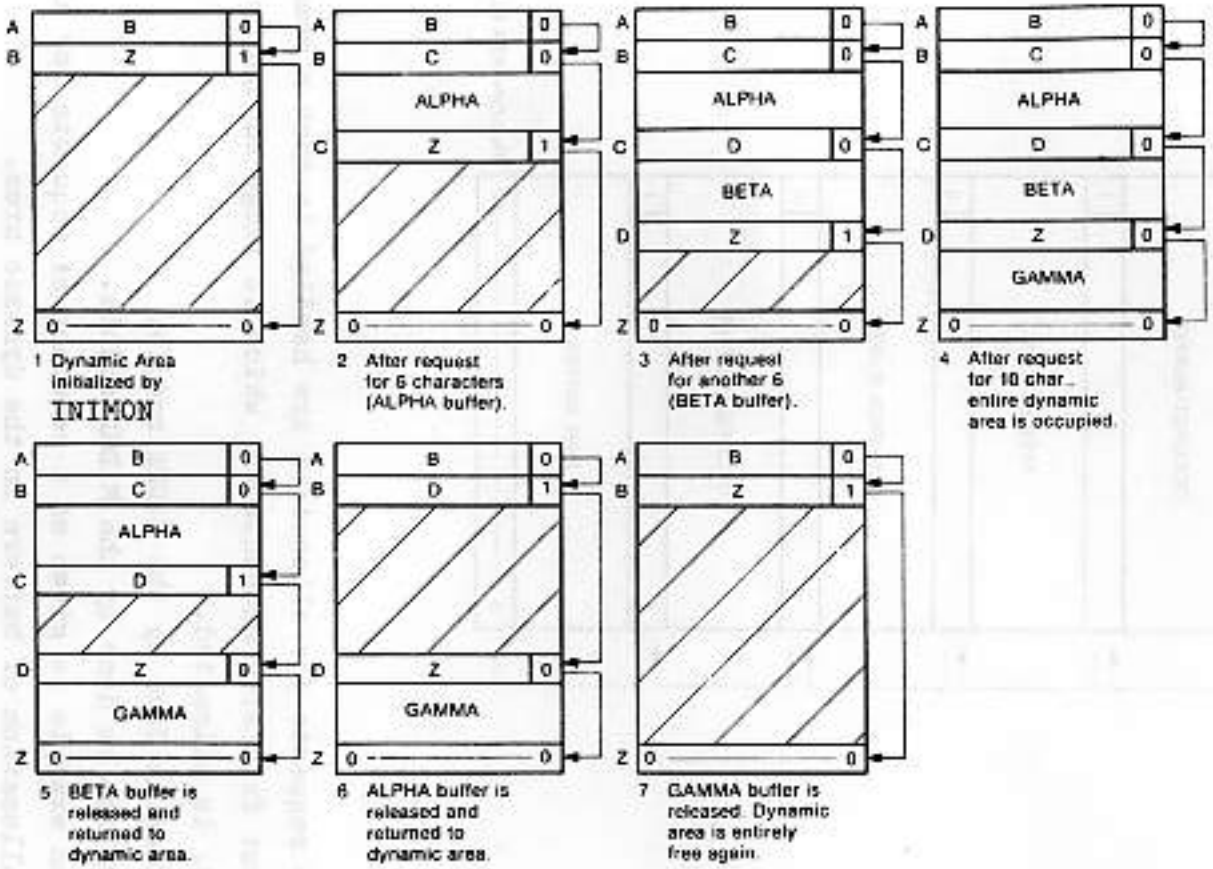
to be specified at system generation time, is 704 words. INIMON gives the following layout to this area, which is to be used for dynamically allocating buffer space to monitor or user programs:



Blocks of memory space can be requested either by the system itself, or by the user through a Get Buffer monitor request. When a buffer is allocated, a buffer guide of one word is set in front of it, in which bit 15 is set to 0, to indicate that the buffer is allocated. If a request is sent for deallocation of this buffer, either by the system or by the user (Release Buffer monitor request), this bit is set to 1, to indicate that the buffer is free again. After a number of such requests, the dynamic area might look as follows:



Any new requests for allocation are handled in such a manner that the first free area encountered, which is large enough for the request, is allocated.
Allocation is done by the M:DMA module.
Deallocation is done by the M:DML module.
Below an example is given of a sequence of requests for allocation and deallocation of buffers in the dynamic area.

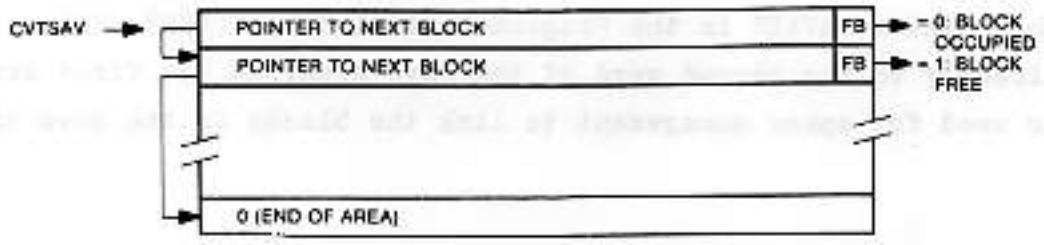

A
B
Z
B
0
1
0
0
1 Dynamic Area initialized by
INIMON
C
ALPHA
2 After request for 6 characters (ALPHA buffer).
D
BETA
3 After request for another 6 (BETA buffer).
GAMMA
4 After request for 10 char., entire dynamic area is occupied.
5 BETA buffer is released and returned to dynamic area.
6 ALPHA buffer is released and returned to dynamic area.
7 GAMMA buffer is released. Dynamic area is entirely free again.

Swap Area

This area, if used, is located behind the memory resident area. Its length is defined at system generation time and this size will be put in location CVTSWP of the Communication Vector Table (T:CVT). After loading, this word will point to the first word of the Swap Area, which is initialized by INIMON.
The size of the Swap Area must be at least as long as the longest swappable program used. To optimize the swapping operations, however, it is recommended that this be a multiple of 200 words.

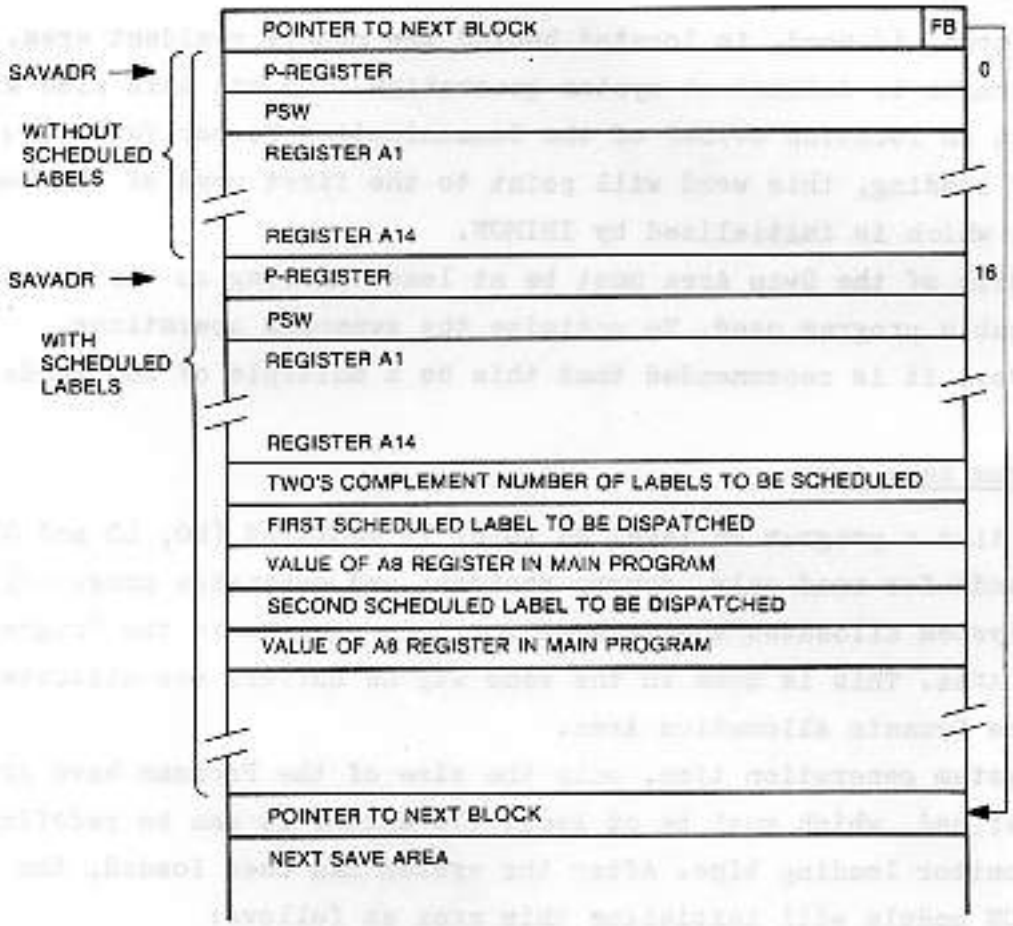
Program Save Area

Each time a program on level 49 to 61 is declared (RO, LD and SW commands for read only, memory resident and swappable programs), the system allocates a save area for this program in the Program Save Area. This is done in the same way as buffers are allocated in the Dynamic Allocation Area.
At system generation time, only the size of the Program Save Area is defined, which must be at least 170 words. It can be redefined at monitor loading time. After the system has been loaded, the INIMON module will initialize this area as follows:



For one program's save area a block is allocated of 16 words or, if scheduled labels are used in the program, of 16+1+16+2n words, where n is the maximum number of scheduled lables allowed in queue (FILLAB, see Part 1, Chapter 4) at the same time. (The number 16 consists of: P-register, PSW, register A1 to A14).

This area is filled as follows:



The pointer SAVADR in the Program Control Table (PCT) will point directly to the second word of the Save Area, as the first word is used for space management to link the blocks in the save area.

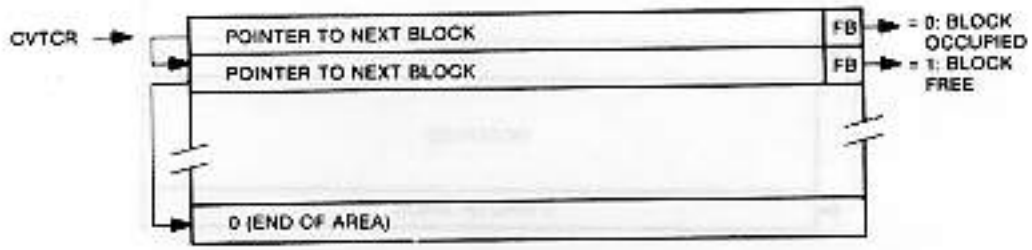
Read Only Area

This area is located behind the Program Save Area in memory. Its length is defined at system generation and must be at least 1088 words. The word CVTRDO in the Communication Vector Table (T:CVT, see below) contains this length at system generation. After loading, the monitor initialization program INIMON will store the

beginning address of the Read Only Area in the location CVTRDO.

Memory Resident Area

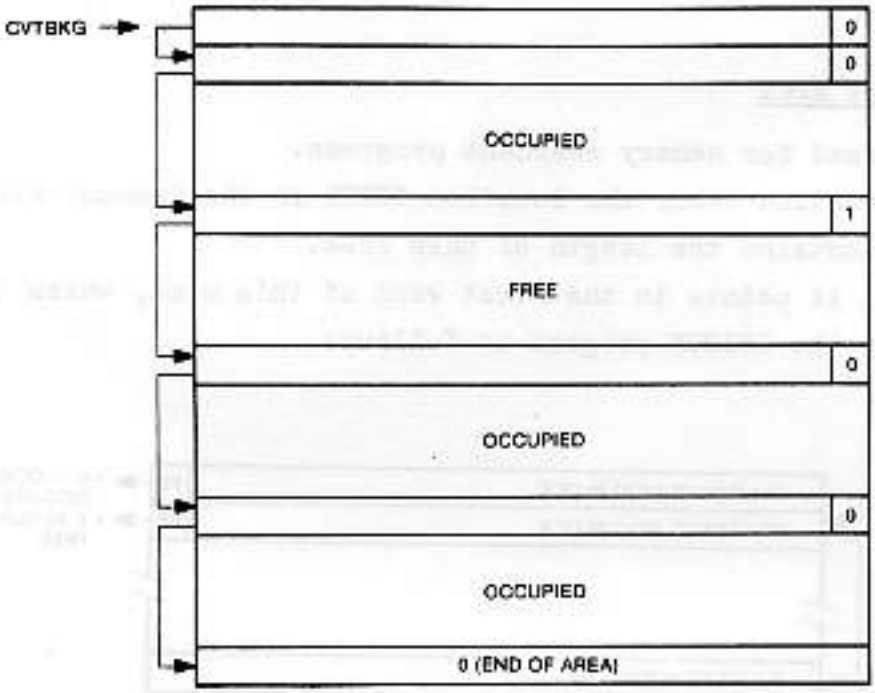
This area is used for memory resident programs.
At system generation time, the location CVTCR in the Communication Vector Table contains the length of this area.
After loading, it points to the first word of this area, which is initialized by the INIMON program as follows:



Memory space is allocated to programs in the same way as in the Dynamic Allocation Area, i.e. each program or memory block is preceded by a pointer, containing the address of the next one and an FB bit.

Background Area

Any memory space remaining after the above mentioned areas is implicitly designated as Background Area. The size must be at least equal to the size of the longest background program used. This area is organized in the same way as the Dynamic Allocation Area:



The difference with the Dynamic Allocation Area is, of course, that here blocks are not allocated for buffers or work areas, but for complete programs.
The location CVTBKG in the Communication Vector Table contains the beginning address of the Background Area.

FILE FORMAT

Apart from the user files, which are either sequential or random as described in Part 1, the system makes use of two other file formats, especially during initialization of the system and during the loading of the different user programs:

Load Module Format

When a program is created, e.g. under the Disc Operating Monitor, the result of the assembly and link edit process is a file of the load module type (LM; see Disc Organization in Part 1). All executable programs are recorded on disc permanently in this format, which is relocatable.

Core Image Format (D:CI File)

Every time the system is started from scratch, a special file, called D:CI file is created on disc, to optimize the loading speed for read only and swappable programs. This is done at monitor initialization time by the INIMON module.
At first this file contains all the system read only programs, in absolute format. It is expanded with all the read only and swappable programs declared by the user through the SCL commands RO and SW. The system can create this file when the beginning addresses of the Read Only Area and Swap Area are known, i.e. when the partitioning of the memory has been processed. Then the system can relocate all the necessary load module files and copy them onto the D:CI file. All this is done by the monitor segment D:USV3, which is first called by INIMON and after that by the System Command Language SCL when an RO or SW command is processed.
The size of the D:CI file must be declared at System Generation (see Appendix A to Part 1).

SWAPPING

For each swappable program, two parameters control the swapping process:

- TS
- SEC

Only if they are both 0 a swapping operation may take place.
TS indicates the time slicing and is decremented by one every 100 milliseconds.
SEC (swap event count) is incremented for each disc I/O and for each other I/O operation if the E option had been selected at declaration time (see SW system command). It is also incremented during the processing of any monitor requests performed by possibly disc-resident progs. It is decremented at the end of an I/O operation as well as at the end of a monitor request for which it had been incremented.
When both TS and SEC are 0, a swap-out followed by a swap-in is started only if a program of equal or higher priority than the current one is waiting to be swapped.
A program exit forces TS to 0 and in this case, when its SEC becomes 0, the program will be overwritten by the next swappable program, if any.
During wait requests, TS is managed as follows:

- explicit wait: if bit 15 of A8 is set to 1, TS is reset to 0.
- with non-disc I/O operations, if the control unit was busy or if the wait bit was set and if bit 7 of A7 was also set, TS is reset to 0.
- put in wait on dynamic allocation area overflow: no action.

Note:

- All Get Buffer requests issued by swappable programs must be issued with bit 0 in A7 set.
- When a program is put in wait because a device has already been attached to another program, TS is reset to 0.

SYSTEM TABLES

Some of the monitor modules provide links between the user programs and system programs. These links are in the form of tables, some of fixed length, some of variable length. The tables provide for all communication between user and system software, especially between memory resident and non-resident programs. These tables must be created at system generation time as they define the main characteristics of the system.
The tables may be located anywhere in the first 16k of memory. One table, which contains pointers and information about all the others, has this address stored in a fixed location:

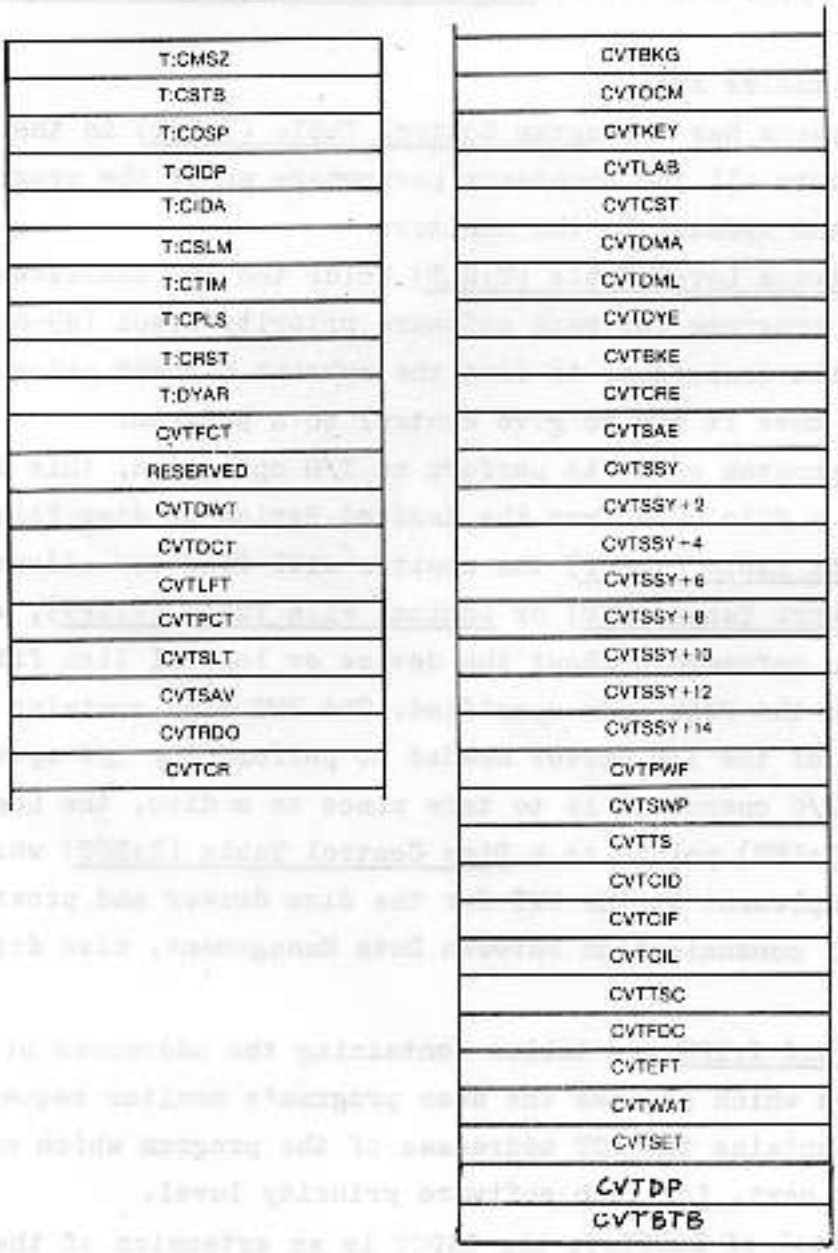
- location /82: address of T:CVT (Communication Vector Table).

The other tables are:

- Each program has a Program Control Table (T:PCT) in the monitor area, where all the necessary parameters about the program are stored and updated by the monitor.
- The Software Level Table (T:SLT) holds the PCT addresses of the current programs for each software priority level (49-63), to enable the dispatcher to find the current highest priority program in case it has to give control to a program.
- When a program wants to perform an I/O operation, this is done through a file code from the desired device or disc file. In the File Code Table (T:FCT) the monitor will find the address of a Device Work Table (DWT) or Logical File Table (T:LFT), which contains parameters about the device or logical disc file related to the file code specified. The DWT also contains the address of the I/O driver needed to perform the I/O operation. If the I/O operation is to take place on a disc, the Logical File Table (T:LFT) points to a Disc Control Table (T:DCT) which acts as a complement to the DWT for the disc driver and provides a means of communication between Data Management, disc driver and monitor.
- T:RMAC and T:LKM are tables containing the addresses of the routines which process the user program's monitor request.
- T:SWP contains the PCT addresses of the program which must be swapped next, for each software priority level.
- T:DP (Pool of DADCTs); the DADCT is an extension of the Disc Control Table
- T:BTP (Pool of BTBs); the BTB is an extension of the DADCT.

Communication Vector Table (T:CVT)

This table provides a useful link between system components which are memory resident and those which are not. It contains the addresses of system tables and common subroutines. By addressing them, in read only system programs, according to their relative positions in the CVT, an external reference from a non-resident to a memory resident system component may be avoided.



T:CMSZ gives the machine's memory size. It is calculated by the system at loading time. If it is 0, the memory size is 32k.

T:CSTB contains the base address of the system stack, as defined at system generation time.

T:CDSP contains the address of the dispatcher.

T:CIDP contains the status word of the Idle Task.

T:CIDA contains the P-register value of the Idle Task.

T:CSLM contains the maximum number of scheduled labels allowed at the same time in one program.

T:CTIM contains the address of the Real Time Clock tables.

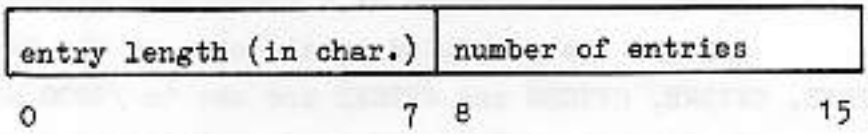
T:CPLS contains the number of pulses of the timer during one 20 msec period (PR): 1 for the 50Hz standard clock.

T:CRST contains the non-standard clock reset value.

T:DYAR contains, at system generation time, the length of the dynamic allocation area. After the system has been loaded it contains the address of the first word of this area.

CVTFCT contains the address of the File Code Table (T:FCT), i.e. of its first word. This word contains the length of the file code table, in characters.

CVTDWT contains the address of the first entry in the Device Work Table (DWT). The word (CVTDWT)-2 contains the entry length and the number of entries in the DWT:



CVTDCT points to word 0 (DCTHD) of the first entry in the Disc Control Table (DCT).

CVTPCT points to the first word of the first entry in the Program Control Table (PCT) Pool. This first word contains the name of the first program in the PCT Pool.

CVTSAV contains, at system generation time, the length of the Program Save Area. After the system has been loaded, it points to the first word of this area.

CVTRDO contains, at system generation time, the length of the Read Only Area. After the system has been loaded, it points to the first word of this area. The Read Only Area is preceded by two reserved words used by the loader.

CVTCR contains, at system generation time, the length of the Memory Resident Area. After the system has been loaded, it points to the first word of this area.

CVTBKG need not be set at system generation time. After system loading, it will contain the address of the first word of the Background Area.

CVTOCM contains the address of the 72-character buffer which is used to read an operator message. The buffer must be located in the first 16k of memory, because the sign bit is used to indicate whether the buffer is being processed (=1) or free (=0) to accept an operator message. If the sign bit is set to 1 and the operator presses the control panel interrupt button, it will be ignored by the system.

CVTKEY is set to zero at system generation time and is used by the 'Read an Unsollicited Key-in' monitor request to build the chain of requests from user programs for reading the unsollicited key-in.

CVTLAB contains the address of a dispatcher subroutine which manages the scheduled label queue in the Save Area.

CVTDYE, CVTBKE, CVTCRE and CVTSAE are set to /8000 at system generation time. They are used to indicate overflow (sign bit = 0) or not (=1) in the Dynamic Allocation Area, Background Area, Memory Resident Area and Program Save Area. In case of overflow, these event bits are used to put a requesting program in wait state.

CVTSSY is a status word containing various DRTM flags. The sign bit and bit 1 are used by the Dump program when the HD command is given.

Bit 2 = 1: the automatic restart routine stops after a power failure to allow the operator enough time to perform any manual operations required.

Bit 2 = 0: no halt in the automatic restart routine.

Bit 3 = 1: the system is ready to run. This bit is reset to 0 at system generation time. After completion of loading and initialization, it is set to 1 to indicate that all system read only programs are now recorded in the D:CI file, in core image format.

Two other status words are also available for user programs. They can be used by any application either to contain table addresses or any information about the application. These words are all reset to zero at system generation time.

CVTPWF Power failure/automatic restart option, set by system generation. If it contains 0, the option has not been selected and the system will issue a halt in case of power failure. If the option has been selected, this word contains the address of the automatic restart routine.

CVTSSY + 2 contains the address of the M:XNAM routine (search PCT).

CVTSSY + 4 contains the address of the BTIMER chain pointer (B:POIN)

CVTSSY + 6 contains the address of the CTIMER chain pointer (C:POIN)

CVTSSY + 8 contains the address of the RTIMER chain pointer (R:POIN)

CVTSSY + 10 contains the address the release timer block routine (F:BLK)

CVTSSY + 12 and +14 are status words reserved for the user for inter-program communication.

Bit 15 is used to denote a power failure which has not been processed.

CVTSWP At system generation time this word contains the size of the swap area (at least 2 characters). After initialization it contains the beginning address of this area. The area is preceded by 2 reserved words used by the loader.

CVTTS This word contains a value used as the standard or default value of the time slice for swappable programs. It is set either at system generation or by the command TS.

CVTCID Contains the file code of the disc which contains the D:CI file. It is set at system generation time.

CVTCIF Contains zero at system generation time. After initialization it points to the first free sector of the D:CI file.

CVTCIL At system generation time this word contains the size (in number of granules) of the D:CI file. After initialization it points to the last sector of the D:CI file.

CVTTSC This word is the time slice counter of the current swappable program (reset at zero at system generation time). When this value reaches zero, the program is swapped out and another swappable program is activated.

CVTFDC, CVTEFT } 2 words used by EDFM

CVTWAT contains the entry point in the Wait routine

CVTSET contains the entry point in the Reset routine

CVTDP contains the address of the first word of the DADCT Pool.

CVTBTP contains the address of the first word of the BITAB Pool.

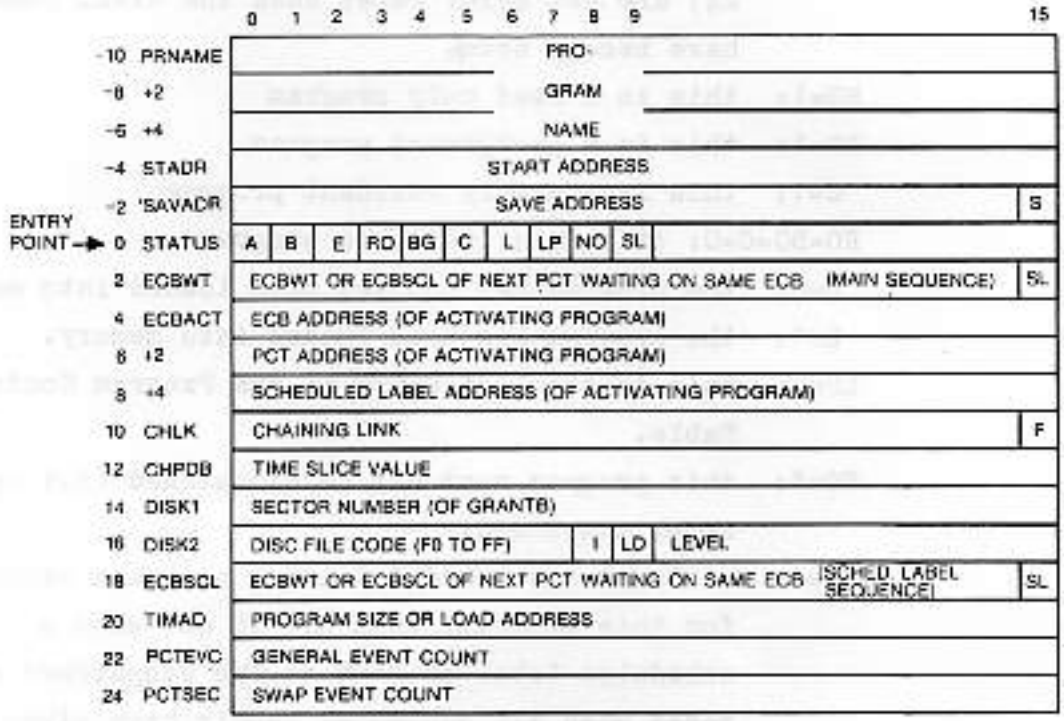
Program Control Table (T:PCT)

This table contains all relevant information about any program and relates to all the activities taking place in the system at a specific moment. There is one 18-word entry for every program, some of them for system programs, some, reserved at system generation time, for use by the System Command Language for the processing of control commands. When a program is deleted, the corresponding PCT entry is released.

The address of T:PCT is stored in the Communication Vector Table (T:CVT), to facilitate access by the System Command Language for creating a PCT entry.

At system generation time the number of PCT entries must be specified. This number cannot be modified. Reference to the PCT must be made through the label indexed by the PCT address, i.e. the word STATUS in the PCT.

One PCT entry has the following layout:



where:

PRNAME: is a 3-word block, containing the program name in ASCII left justified and filled with blanks if less than 6 characters long.

STADR: is the start address of the program.

SAVADR: Address of the save area of the program:
If S=0, it is the address of the main program save area.
S=1, it is the address of the scheduled label save area.

STATUS: gives the status of the program. This word is the entry point of the PCT block.
If A=0: the program is in active state (set by Activate (M:ACT)).
A=1: the program is in inactive state (set by Exit (M:EXIT), when Event Count is zero).
B=0: this PCT entry is free for allocation to a program.
B=1: this PCT entry has already been allocated to a program.

E=0: the program has not made its exit.

E=1: the program has made its exit (set by Exit (M:EXIT), when the event counts (words 22 and 24) are not zero; reset when the event counts have become zero).

RO=1: this is a read only program

BG=1: this is a background program

C=1: this is a memory resident program

RO=BG=C=0: this is a swappable program

L=0: the program has not yet been loaded into memory.

L=1: the program has been loaded into memory.

LP=1: this is the last entry in the Program Control Table.

NO=1: this program must not be dispatched (not operable, suspended).

SL=1: at least one scheduled label has been recorded for this program. This bit is set when a scheduled label is sent to the dispatcher and reset when all scheduled labels have given their exit.

The program cannot become inactive until the counter (PCTEVC) has become zero again (see bit E)

ECBWT: ECBWT or ECBSCL (SL bit set in last case) of the next PCT waiting on the same ECB as the one on which the main program waits (see T:EVT table)

ECBACT: A block of three words to enable the activating program to synchronize itself with this program, i.e. the program activated by it. The block contains:

- ECB address
- PCT address
- scheduled label address

The activating program can synchronize itself by giving a Wait monitor request after the Activate request or by giving a scheduled label at activation time. When the activated program is completed, the label will be scheduled. However, if a Wait request has been given, the program itself must send a Set Event monitor request. This is not

done implicitly upon exit of the program.

CHLK: Chaining link (see below).

CHPDB: The left character of this word contains the time slice value for this program. It is set by the control command SW (Declare a Swappable Program). Default value = CVTTS (see Communication Vector Table).

DISK1: Contains the relative sector number of the first sector of this program on disc. If it is a background program, it points to the sector GRANTB of the load module. If it is a core image program, it points to the first sector of the program within the D:CI file.

DISK2: Bits 0 - 7 contain the file code of the disc (F0 to FF) on which this program is located.

Bit I = 0: All pending I/O operations of the current swappable program must be completed before swapping the program.

Bit I = 1: Only the pending disc I/O request must be completed before swapping the program (ECB and buffer areas used by non-disc devices are outside the swap area).

Bit LD = 1: This is a background program which is being loaded.

Bits 10 - 15 contain the level to which this PCT is connected.

ECBSCL: ECBWT or ECBSCL (SL bit set in last case) of the next PCT waiting on the same ECB as the one on which the scheduled label waits (see T:EVT table).

TIMAD: Contains the memory space required to load the program, if this is a background program, or the beginning address of the program if it is loaded into the memory resident area. It then points to the first word of the memory block into which the program is loaded, i.e. the block pointer.

PCTEVC: Event Count used for Exit Control.

PCTSEC: Swap event count used for swap control. The program may be swapped out only when this word reaches the value zero.

Use of Event Counts

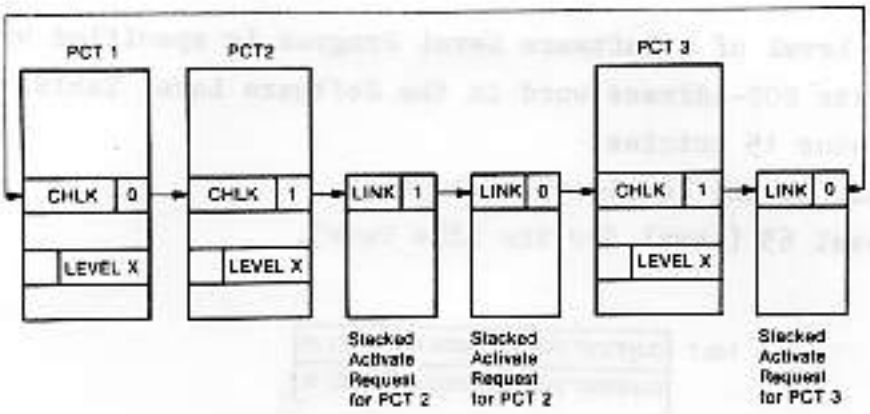
Generally speaking, these words are incremented whenever an action such as an I/O operation, a scheduled label, execution of a non-resident monitor request, is requested by the user and decremented at completion of the action.
For the event counts PCTEVC and PCTSEC, these are the following actions:

- PCTEVC: - incremented by IORM
  decremented by ENDIO
  - incremented by I:LKM if scheduled label requested
  decremented by M:LAB (M:DISP) when scheduling occurs
  - incremented by I:LKM when a non-resident monitor request is requested
  decremented by the monitor request handler itself when the request has been processed

- PCTSEC: - incremented by IORM if bit I of DISK 2 is set or if it is disc I/O or bit 7 for a non-disc I/O order is not set
  decremented by ENDIO under the same conditions
  - incremented by I:LKM when a possibly disc-resident monitor request is requested
  decremented by the monitor request handler itself when the request has been processed.

Chaining Link

If, in this word, F = 0, it indicates a link to another PCT. If F = 1, it is a link to a stacked Activate request.
All PCT blocks connected to the same priority level and the stacked Activate requests for them are linked together in a chain as follows:



- PCT's are inserted into the chain each time a monitor request 'Connect to a Level' is given for a program on the same level. If there is only one PCT, the chain loops on itself.
- Activate requests are stacked each time an Activate request is given for a program which is already active: an entry is created (by dynamic memory allocation) and inserted into the PCT chain after the PCT block itself. The entries for these stacked Activate requests hav the following layout:
  - Level 49 stacked entry:

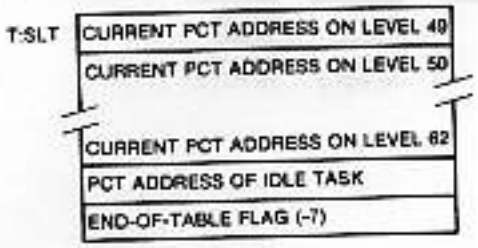
LINK	F	
ADDRESS OF CALLED ROUTINE		(A3)
PCT ADDRESS OF ACTIVATING PROG.		(A5)
SCHEDULED LABEL ADDRESS		(A6)
PROGRAM NAME ADDRESS		(A7)
ECB ADDRESS		(A8)

These blocks are created by the Activate monitor request handler (M:ACT) and erased by the Exit monitor request handler (M:EXIT).

Software Level Table (T:SLT)

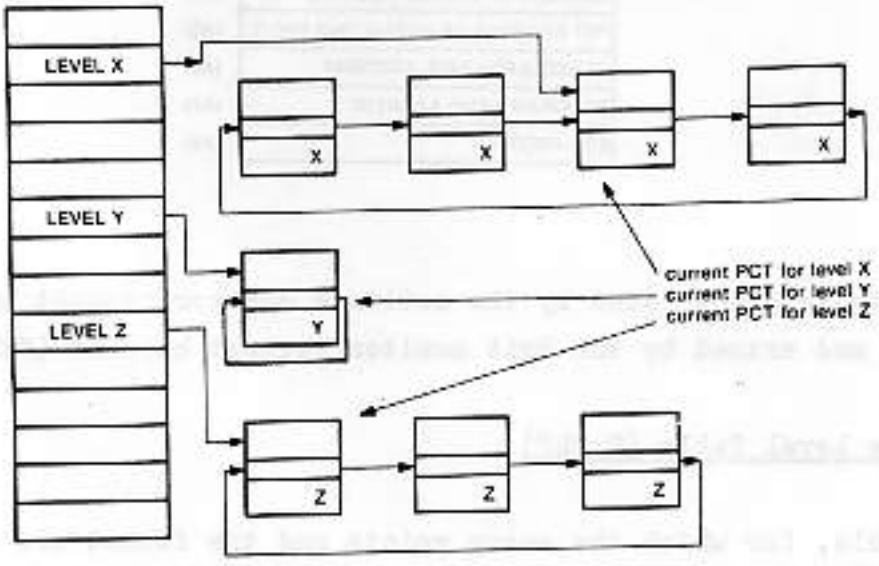
This table, for which the entry points and the format are defined in the monitor module T:SYS, is used by the dispatcher to find the current PCT for a given level.

The priority level of a Software Level Program is specified by the position of its PCT-address word in the Software Level Table.
The SLT contains 15 entries:
- one for each of the levels 49 through 62
- one for level 63 (level for the Idle Task).



Example of PCT-SLT Organization

The links between SLT and PCT entries are created by the 'Connect a Level' monitor request handler (M:CNLV) and deleted by the 'Disconnect a Level' request handler (M:DNLV), when there is only one PCT in the chain. Within the chain itself the links are changed by the Switch Inside a Level request handler (M:SWTC).



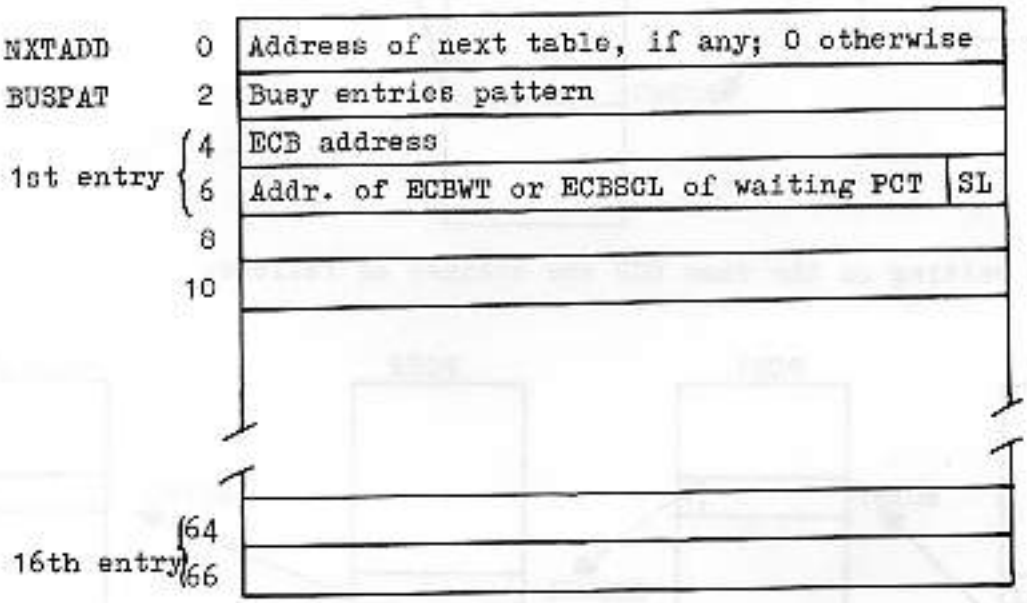
Event Tables (T:EVT)

This table contains up to 16 entries of 2 words each. These entries are created when a program for the first time wants to wait for one specific event (explicit or implicit wait) and they are released when the last program waiting for that event is restarted (by a Set Event request). The entries in the table are shared by both system and user programs.

Only one table is delivered as standard in a separate module. If it becomes necessary, the user may expand it with more such tables.

This standard module is called M:EVTB and its entry point is called T:EVT, referred to in the M:WAIT and M:RSEV modules.

The entry point T:EVT must correspond to the word NXTADD of the first table:



NXTADD : Address of the word NXTADD of the next event table, if there is any; otherwise, this word contains 0.

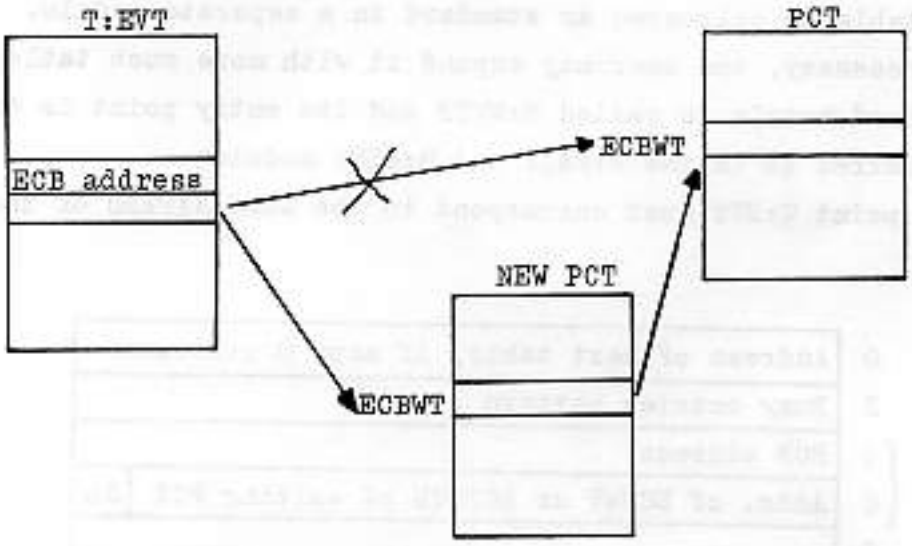
BUSPAT : Pattern of the busy entries in the table, each bit corresponding to one entry in the table (bit 0 = 16th entry, bit 15 = 1st entry). A set bit indicates that the corresponding entry is free.

Each entry is 2 words long:

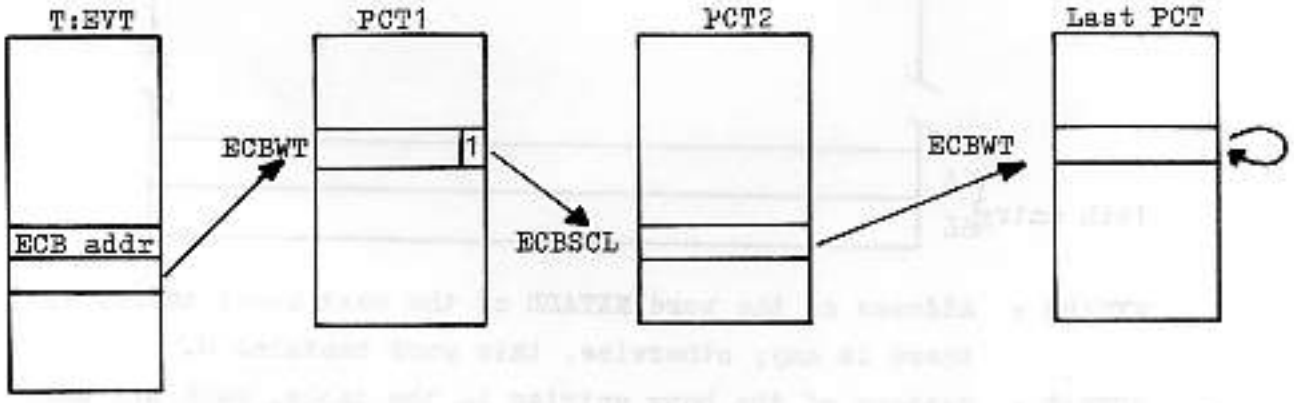
- the first word contains the address of the ECB on which the program is waiting
- the contents of the second word depends on the sequence type of the

waiting program: if it is the main program sequence it contains the address of the word ECBWT of the PCT of the program, if it is a scheduled label sequence, it contains the address of the word ECBSCL of the PCT (in this case bit 15, SL, is set).

As soon as another program wants to wait on the same ECB, the link between the T:EVT and the first PCT is broken and the new PCT is inserted into the chain:



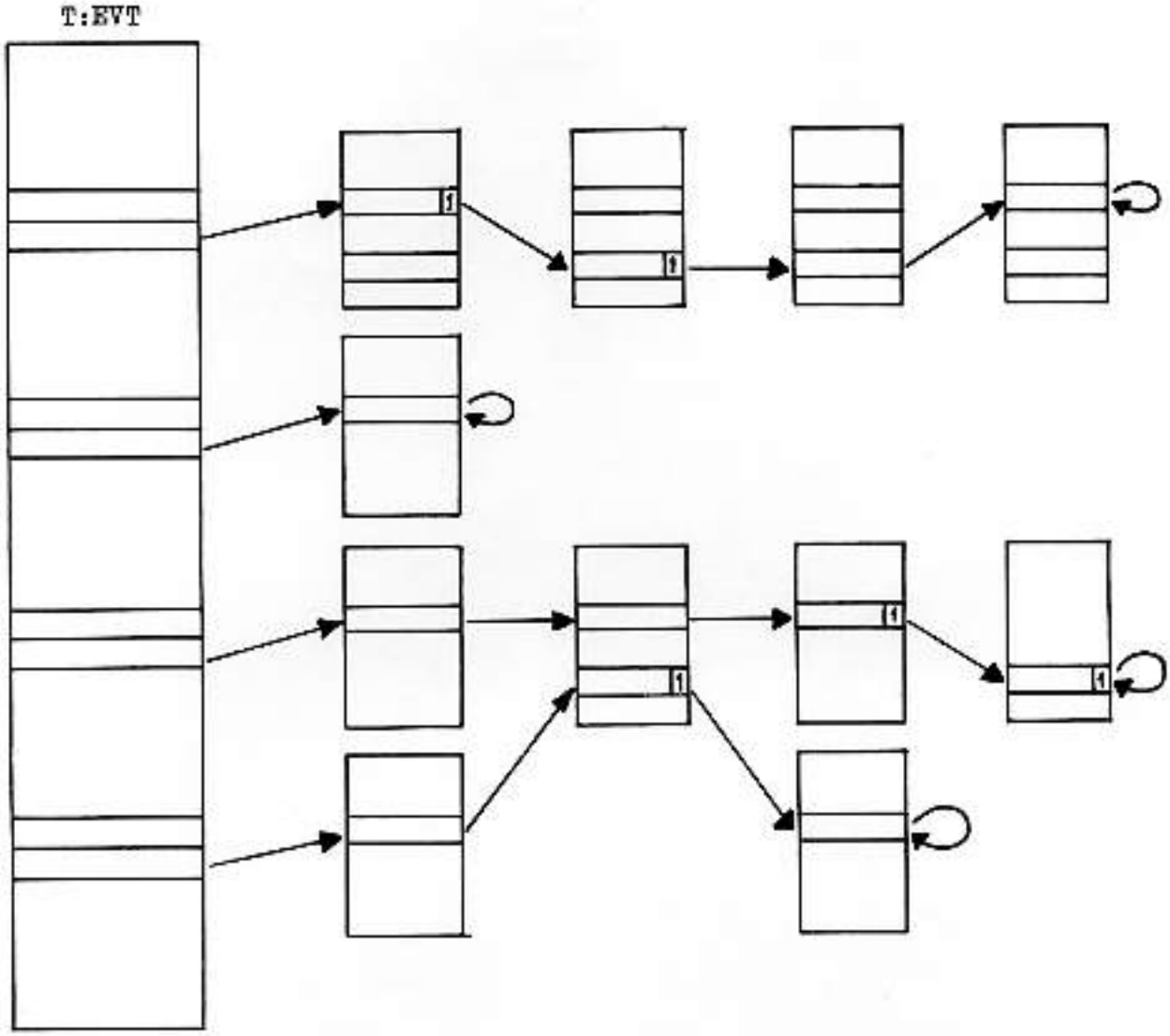
All PCTs waiting on the same ECB are chained as follows:



Event Table Overflow: when no more free entry is left in the Event Table or Pool of Event Tables, a branch is made to SYSAB, with register A2 = 6.

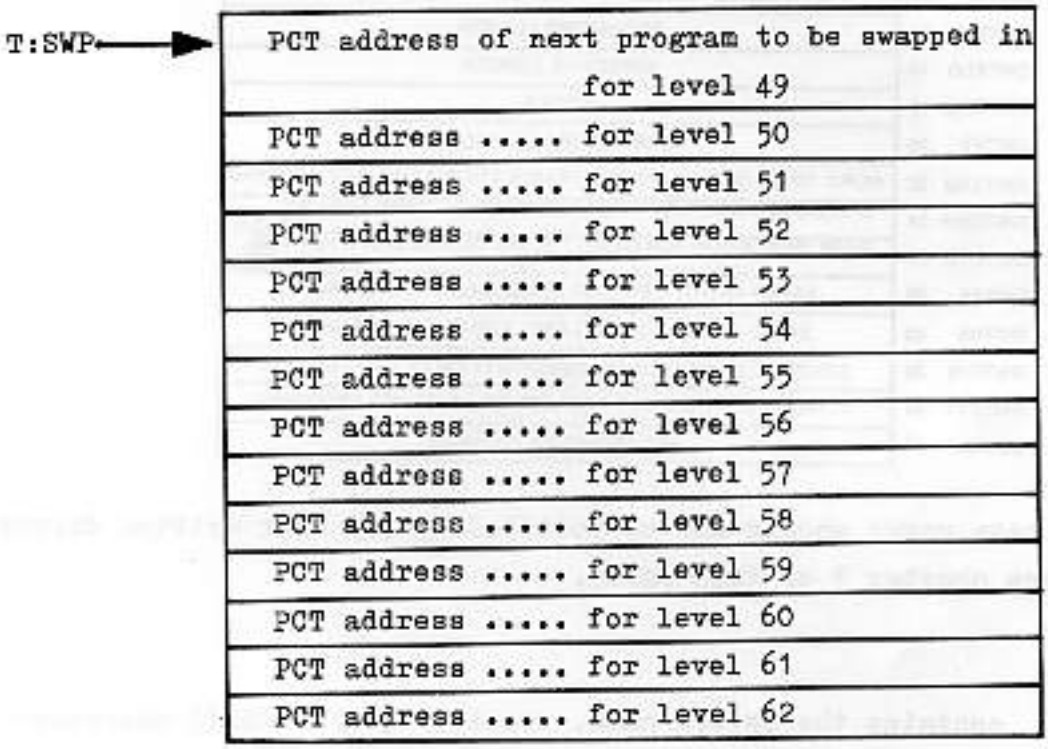
Note: Due to the response time of the requests Wait for an Event and Set Event, it is not recommended to extend the pool of event tables excessively.

Example of T:EVT - PCT organization:



Swapping Table (T:SWP)

When a time-slice has elapsed for one program, the dispatcher (M:DISP) needs to know the next program to be swapped in. It is found in this table where these programs are recorded (one per level). If there is no program to be swapped in, the entry in T:SWP is set to zero.

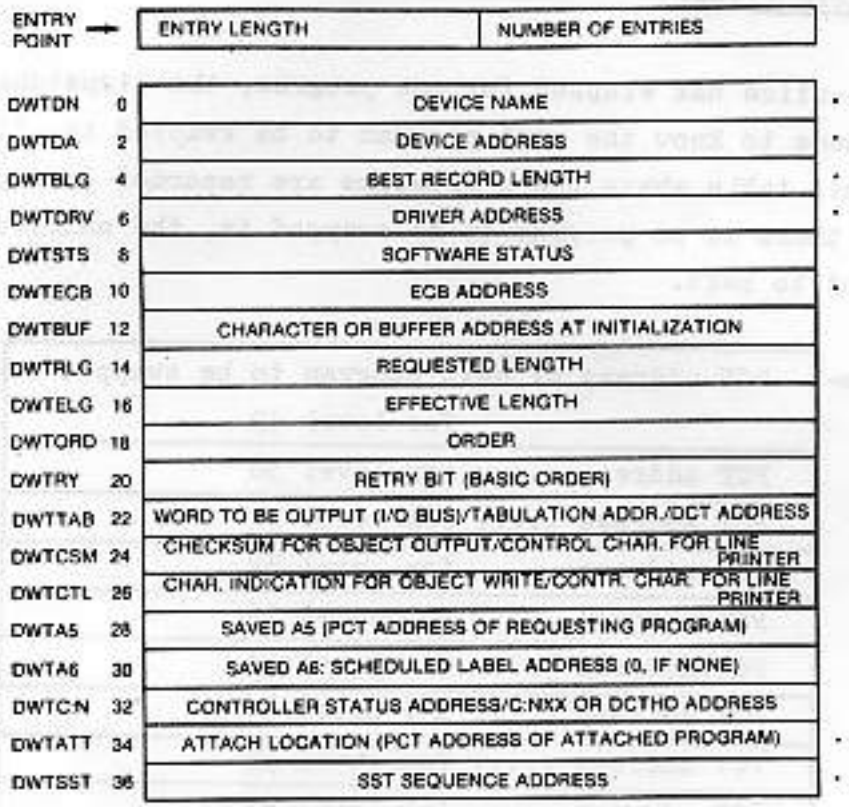


Device Work Table (DWT)

The DWT is a monitor module, which consists of fixed length 19 word entries, one entry being built for each peripheral device. Note that the ASR is considered as consisting of three different devices and, consequently, occupies three entries in the DWT: one each for the typewriter, the tape reader and the tape punch.

T:DWT points to the first entry of the table which is preceded by a word which contains, in bits 0 - 7, the length of one entry in characters, and in bits 8 - 15, the number of entries in the DWT.

An entry in the DWT has the following layout:



(*: These words should not be modified by any user written drivers; see chapter 3 of this part).

where:

WORD 0 contains the device name, consisting of 2 ASCII characters.

WORD 2 contains a binary value giving the physical address of the device.

WORD 4 contains the recommended maximum record length for this device. This number depends on physical device limitations, the transfer rate of the I/O processor and on compatibility with other devices:
typewriter, tape punch, tape reader, card reader: 80.
cassette tape: 256. magnetic tape: 4095. Line printer: 80 or 136.

WORD 6 contains the address of the I/O driver belonging to this device.

WORD 8 Software status, used to record any I/O error during data

transfers on the I/O bus or to register the specific basic I/O operation to be performed for an order for Tape Cassette or Magnetic Tape (see Part 1, Chapter 11)

WORD 10 contains the address of the user's ECB.

WORD 12 contains the user buffer address, or the character address of the next character to be input or output.

WORD 14 Requested length of the I/O operation, given by the user.

WORD 16 Effective length, used to count the number of characters actually transferred during an I/O operation.

WORD 18 The order for the I/O operation, as specified by the user in the A7 register to define the particular I/O function to be performed.

WORD 20 is used to record that the user wants the standard recovery feature, even for Basic I/O orders.

WORD 22 Next character or word to be output, for devices on the I/O bus. For disc devices, this word points to the DCT (Disc Control Table) address.

WORD 24 For an object write order this word contains the checksum. For Line Printers, this word contains the last character, saved from the buffer.

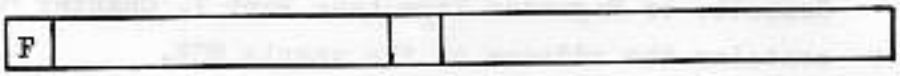
WORD 26 specifies, in case of 4 x 4 object input for devices on the I/O bus, whether the right of left character must be entered. In case of line printer output it is used to save the control code (first two characters of a buffer).

WORD 28 is used to save the program PCT address as recorded at initialization time by the IORM module. It is returned to the dispatcher at completion of the I/O operation.

WORD 30 is used to save a scheduled label address, as recorded at initialization by the IORM module. It is returned to the dispatcher at completion of the I/O operation. This word contains 0, if no label has been scheduled.

WORD 32 For disc, address of location DCTHD in the Disc Control Table. For non-disc devices, address of the control unit status word. With every DWT entry, a control unit status word is associated, indicating wether an I/O operation

can be initiated or not (busy/not busy). The sign bit indicates whether the device is free or not.
For non-disc devices, this word is used as follows:



F = 1 if the control unit is free; 0, if busy.
If busy, bits 7 to 15 are used as follows:
bit 7 = 1, set by M:RETR to indicate that the device is waiting for an RY or RD operator message. If RY is typed it is reset to zero; if RD is typed, it remains 1, to indicate to M:RETR not to output the RY flag in any following PU error message and return the status to the calling program.
Bits 8 to 15 contain the contents of the BOU lines, to be sent when a CIO instruction is received.

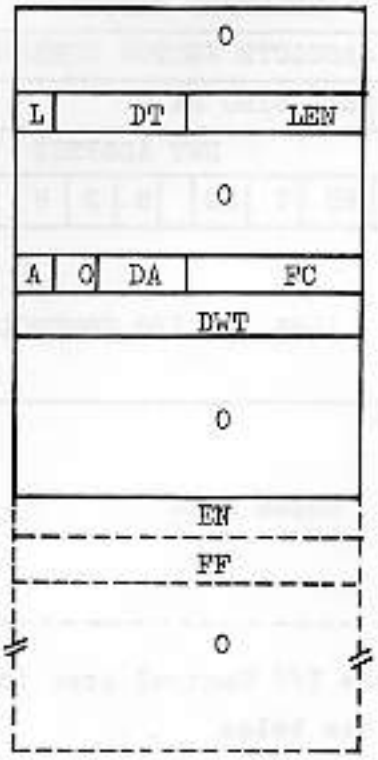
WORD 34 This word is used to record the PCT address of the program which has been attached to this device by means of a monitor request. If this device is not attached, this word contains the hexadecimal value /8000.

WORD 36 This word is used to find the SST (send status) sequence address relative to a device, in case of a throughput error on a device connected to the I/O bus.

T:DCT (Pool of DCTs: Disc Control Tables)

At system generation time, one entry must be created for each disc in the configuration. For it, the user specifies disc type, interrupt level and number of queue entries.
The queue at the rear of each DCT may be omitted if the user will not make use of DFM in accessing the disc.
The DCT length is variable, because the length of the queue is variable.
The entry point T:DCT points to the start address of the first DCT in the pool. The is not the first word of the DCT (DCT address 0 = DCTHD).

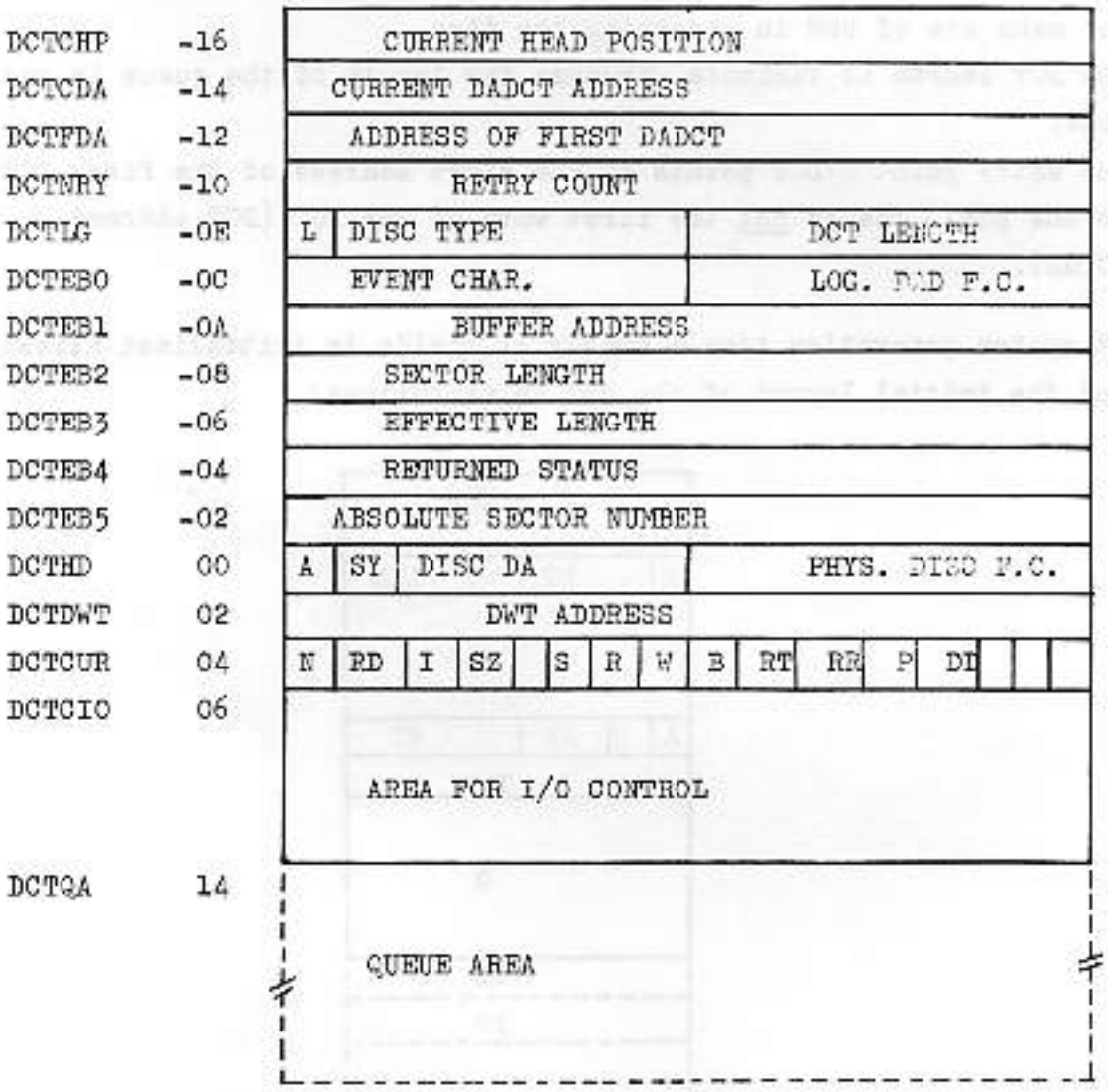
At system generation time a number of fields is initialized already and the initial layout of the DCT is as follows:



DCT (Disc Control Table)

The Disc Control Table is a complement to the DWT (Device Work Table) for the disc drivers. It facilitates communication between Disc File Management, disc driver and monitor.

Layout is as follows



Note: - Layout of the I/O Control Area is variable, depending on disc type. See below

- Queue area is present only if specified by user at system generation time. See below.

DCTCHP: current head position; used by the driver to check if a seek operation is necessary.
The value depend on the disc type:
X1215: 0-202
X1216: 0-
CDC 40M: 0-
CDC 80M: 0-

DCTCDA: address of the current DADCT.
Set by IORM via the DAD file code and used by the driver to calculate physical cylinder number physical sector and so on.
For physical disc access this word contains 0.

DCTFDA: first pointer of DADCT chain. (See DADCT).

DCTNRY: Number of times an I/O operation is to be retried before it is marked down as irrecoverable I/O error.
At the first erroneous I/O operation, the retry indicator is set and the retry count initiated.
The initial value depends on the type of operation:
- seek: 1
- read: 26
- write: 3

DCTLG: length of DCT (in characters)
Only the right hand character is used to specify the DCT length.
Bit 0 (L) is used as an end indicator:
L=1: last entry in T:DCT
L=0: not the last entry.
Bit 1 and 2 are not used.
Bit 3 and 4 indicate the disc type:
00: X1215 or X1216
01: CDC disc
Bits 5, 6 and 7 further specify the disc type:
X1215 or X1216: 000 = X1215
001 = X1216
CDC disc : 000 = 40M
001 = 80M.

DCTEB0 to DCTEB5 is the ECB used only by the Disc File Management routine to make a request to the physical disc driver:

DCTEB0: bits 0-7 : event character
bits 8-15: file code of the DAD on which the file is stored.

DCTEB1: address of the buffer (one sector) used by the physical disc driver to read or write a sector. For sequential access this is the address of the blocking buffer, for direct access of the user buff

DCTEB2: length of the buffer used to read or write a sector: 410 characters.

DCTEB3: effective length, returned by the physical disc driver: 410 characters.

DCTEB4: status of the request made for a module on level 49, as returned by a module on level 48 (disc I/O driver).

DCTEB5: relative sector address (relative to beginning of the DAD) of the sector to be read or written.

DCTHD : Base address of the DCT entry.
Bit 0(A) is a software busy indicator:
- 0 : device is active, request has been received.
- 1 : device is free.

Bit 1(SY): system disc indicator (set by INIMON):
- 1 : this DCT belongs to the system disc.

Bits 2-7 : physical device address.
Bits 8-15: physical disc file code (/F0 to /FF).

DCTDWT: Address of the DWT (Device Work Table) to which this DCT belongs.

DCTCUR: Indicators:
Bit 0(N) : operable indicator:
- 0 : disc operable (set at system generation)
- 1 : disc not operable (set by INIMON, when disc not ready).

Bit 1(RD): ready indicator:
- 1 : disc ready, new pack has been mounted, but not yet initialized.

Note: When bit N and/or bit RD is 1, the disc driver will refuse a request, even from a system

program. RD is set to 1 when the system receives an unexpected interrupt from the disc unit while the 'disc ready' bit in the status word is 1. When RD is 1, the program DISKINIT (part of D:USV1) will be called to reinitialize the disc by

- suppressing the file codes assigned to the files on the old disc,
- scratching the DAD administration of the old disc,
- resetting N and RD to 0,
- resetting bit A in DCTHD

Bits 2 to 7 indicate the operation to be performed:

Bit 2(I) : interrupt indicator:
= 1 : an order has been sent to a device and the system is waiting for the interrupt. Set to 1 after a CIO instruction, reset to 0 after execution of SST instruction.

Bit 3(SZ): seek to zero indicator:
= 1 : a seek to zero must be performed. Set only on detection of a seek error, either via hardware status or software identifier check.

Bit 4 : not used (must be 0).

Bit 5(S) : seek indicator:
= 1 : a differential seek must be performed. The contents of register A3 will be given in word DCTSK.

Bit 6(R) : read indicator:
= 1 : a read order must be sent to the disc.

Bit 7(W) : write indicator:
= 1 : a write order must be sent to the disc.

Note: the operation indicators SZ, S, W and R are initiated by the disc driver, depending on the requested order:

	SZ	S	R	W
read	0	0	1	0
read (+ seek)	0	1	1	0
write	0	0	0	1
write (+ seek)	0	1	1	1

The driver attempts to execute the operation indicated by the leftmost 1-bit. When the driver resets this bit after a successful SST instruction, it tries the next one and so on until all bits are zero.
When a seek error is detected, bits SZ, S and R are set and bit W remains unchanged.

Bit 8(B) : position lost indicator:
= 1 : the head position recorded in DCTCHP does not correspond to the actual head position, i.e. the following operation for this disc will be a seek to zero (Note: for CDC discs a seek operation is sufficient).

Bit 9(RT): retry indicator:
= 1 : the disc driver is trying to recover an error. This bit is initiated to 0 when an I/O event is prepared and set to 1 when an error indication is returned after an SST instruction.
This bit is used to check if the retry count in word DCTNRY must be incremented or decremented.

Bit 10(RR): drive busy indicator (only for X1215/X1216):
= 1 : a request is being processed on the other part of an X1215 or X1216 drive: do not start an operation on this part of the drive.

Bit 11(P) : prepared indicator:
= 1 : the DCT is ready to start the I/O operation (but was held up because the control unit was busy).

Bit 12(DD): disc structure indicator
= 1 : this pack has a DAD structure.

Bits 13 to 15 are not used.

DCTIO : Area for I/O control, where the layout depends on the disc type:

X1215/X1216:

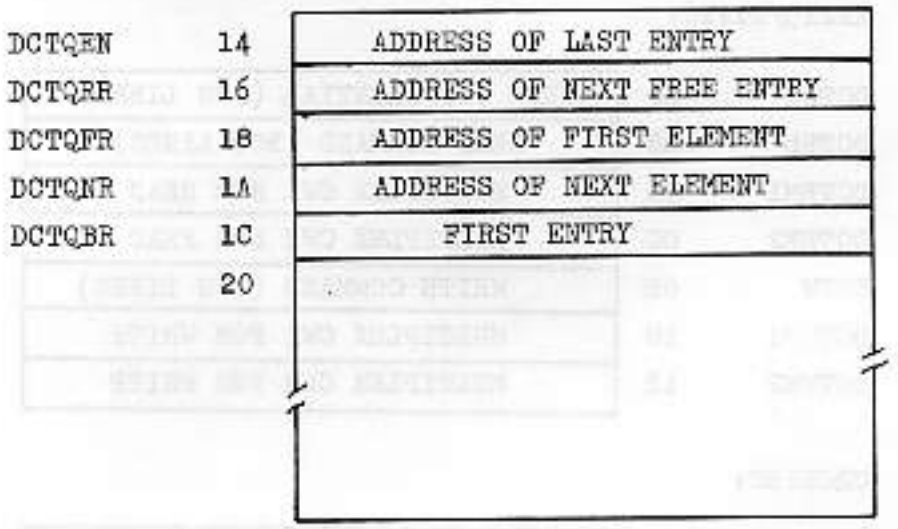
DCTSK	06	SEEK DIFFERENTIAL (BOU LINES)
DCTRD	08	READ COMMAND (BOU LINES)
DCTRM1	0A	MULTIPLEX CW1 FOR READ
DCTRM2	0C	MULTIPLEX CW2 FOR READ
DCTW	0E	WRITE COMMAND (BOU LINES)
DCTWM1	10	MULTIPLEX CW1 FOR WRITE
DCTWM2	12	MULTIPLEX CW2 FOR WRITE

CDCDISC:

06	SEEK TO ZERO
08	SEEK
0A	WRITE OR READ COMMAND
0C	TOTAL LENGTH CW FORMAT
0E	LENGTH CW FORMAT
10	BUFFER ADDRESS

DCTQA : Queue area, used only by M:DFM to queue the requests for files on this disc.
If DFM is not used on this disc, the user can omit specification of this area at system generation time.
The number of entries in the queue is also specified at system generation time.

The layout of DCTQA is:



where:

DCTQEN gives the address of the last entry in the queue area (i.e. the last 2 words of the DCT). Set at sysgen time.

DCTQRR gives the address of the next free entry in the queue area (initiated to DCTQBR).
Maintained by M:DFM.

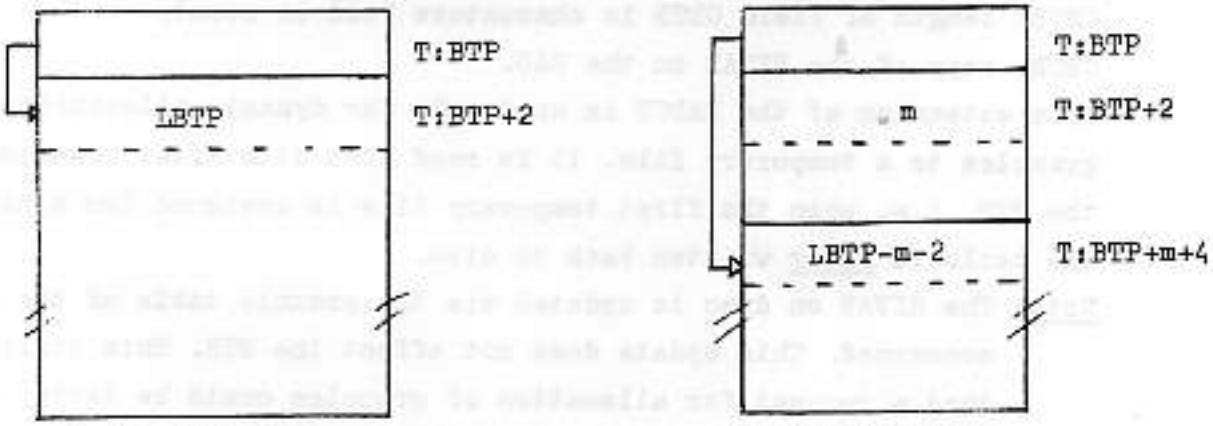
DCTQFR gives the address of the first element in the queue.
It is initiated at 0 and maintained by M:DFM.
When M:DFM is active with a file on this disc, the queue will always have at least one element.

DCTQNR gives the address of the next element in the queue. It is initiated at 0 and maintained by M:DFM. When it is 0, the current element is the only one.

DCTQBR is the first entry in the queue area.

T:BTP (Pool of BTBs=Bit Tables)

The length of this pool is specified at system generation time: 2+LBTB characters, where LBTP specifies the maximum total length of the BITABS which are kept in memory simultaneously, for administration of allocation for temporary files.
The first word of the pool contains information regarding the pool itself, the rest is available for BTBs. When a new temporary file is assigned for a DAD, an entry is taken from this pool to create its BTB. It is returned to the pool when the DAD is scratched from the system.
When no more entries are free in the pool, the system releases the "scratchable" BTBs (DADNTF(first character of DADCT word 04)=0).
The unused area of the pool is always located at the end, i.e. when an entry is released any entries following it are pushed up.
The layout of the T:BTP is as follows, where the first word points to the start address of the free area and the first word of the free area contains the remaining length, in characters:



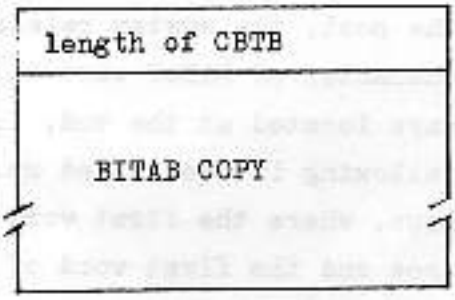
Initial State

Run-time State
(one BTB with length=m)

BTB (Bit Table)

The Bit Table is an extension of the DADCT, required only when temporary file(s) have been assigned to the DAD.
It remains linked to the DAD, even when no more temporary files are assigned for it. In this case however, it may be released when no more space in available in the T:BTP for a new Bit Table for another DAD; A BTB is also released when its DADCT is scratched from the system.

The BTB has a variable length and is part of the T:BTP where it has the following layout:



where:

LBTB: length of field CBTB in characters (odd or even).
CBTB: copy of the BITAB on the DAD.

This extension of the DADCT is used only for dynamic allocation of granules to a temporary file. It is read from disc after creation of the BTB, i.e. when the first temporary file is assigned for a DAD.
The table is never written back to disc.

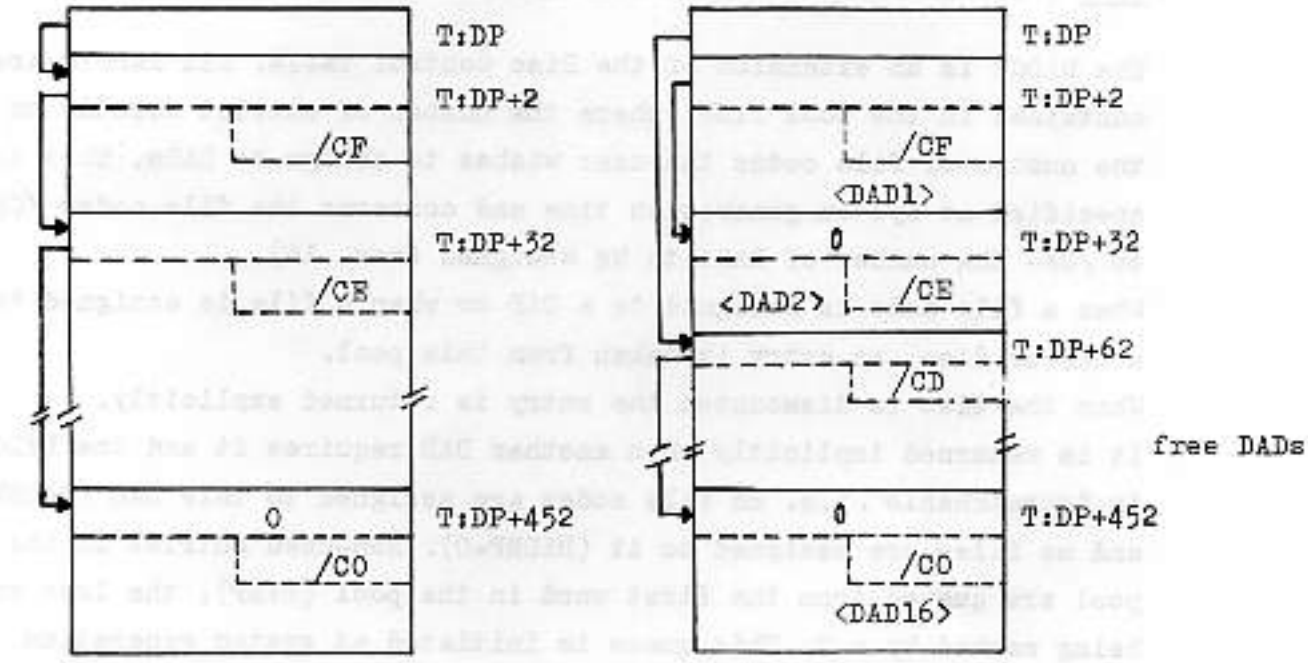
Note: The BITAB on disc is updated via the granule table of the file concerned. This update does not affect the BTB. This implies, that a request for allocation of granules could be denied even if there is still space on the DAD (resulting from a previous 'Delete File' command).

T:DP (Pool of DADCT = Direct Access Device Control Tables).

The DADCT is an extension of the Disc Control Table. All DADCTs are contained in the pool T:DP, where the number of entries depends on the number of file codes the user wishes to assign to DADs. This is specified at system generation time and concerns the file codes /CF to /CF- the number of DADs to be assigned (max. 16).
When a file code is assigned to a DAD or when a file is assigned to a DAD or disc, an entry is taken from this pool.
When the disc is dismounted the entry is returned explicitly.
It is returned implicitly when another DAD requires it and the DADCT is "scratchable", i.e. no file codes are assigned to this DAD (DADNFC=0) and no files are assigned to it (DADNF=0). Non-used entries in the pool are queued from the first word in the pool (T:DP), the last entry being marked by a 0. This queue is initiated at system generation time.
The relation between implicit DAD file code and DADCT is defined and fixed at system generation time, when the field DADIFC in the DADCT is set.

Note: The implicit DAD file code appears in the LFT only when a file has been assigned to that DAD or when a file code has been assigned to it.

The layout of T:DP (for NDAD=16) is as follows:

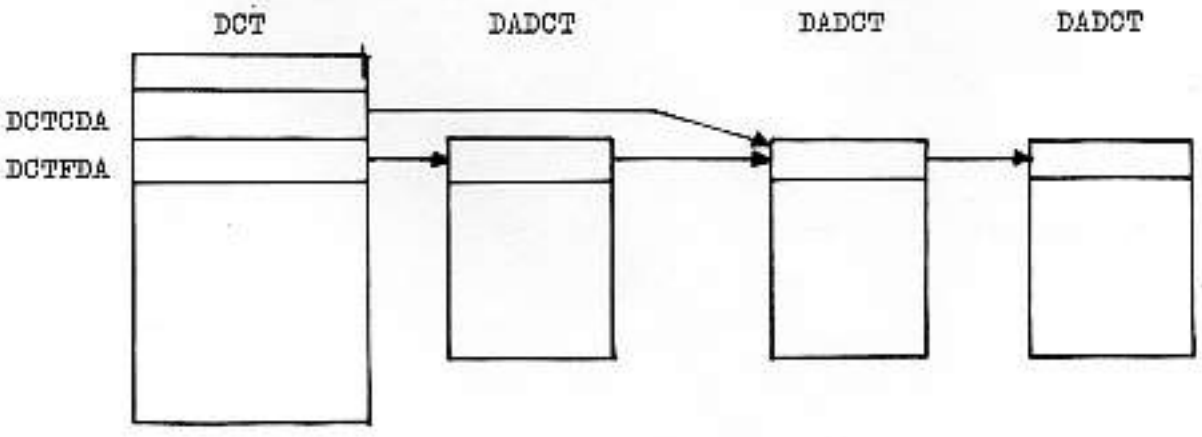


Initial State
(as defined at sysgen)

Run-time State
(DAD1 and DAD2 are assigned on the same disc pack).

DADCT

The Direct Access Device Control Table is an extension of the DCT. One DCT may be extended by a number of DADCTS linked by a forward chain starting in the DCT (DCTFDA). The DCT also contains the address of DADCT on which I/O is currently taking place (DCTCDA).



All DADCTs are located in the DADCT-pool T:DP and have the following layout:

DADLNK	00	FORWARD CHAIN-LINK	
DADFC	02	# OF ASS.	DAD FILECODE
DADNF	04	# OF TEMP. FILES	# OF PERM. FILES
DADDCT	06	DCT ADDRESS	
DADNAM	08	DAD NAME	
	0A		
	0C		
DADINT	0E	NUMBER OF INTERLACES	
DADSPT	10	NUMBER OF SECTORS/TRACK	
DADSLG	12	SECTOR LENGTH	
DADNBC	14	NUMBER OF CYLINDERS	
DADFCL	16	NUMBER OF FIRST CYLINDER	
DADSPG	18	NUMBER OF SECTORS/GRANULE	
DADSTA	1A	STATUS	DAD NUMBER
DADBTB	1C	BTB ADDRESS	

Where:

DADLNK: pointer to the next DADCT belonging to the same DCT (word DCTFDA)
If 0, this is the last DADCT in the chain.

DADFC : Is split in two characters:
DADNFC: contains the number of file codes assigned to this DAD. If it contains 0, no explicit assign has been made.
DADIFC: implicit file code for this DAD; related to the entry number in the DADCT pool, independent of the DAD, but fixed at system generation time.
M:DFM requires this field to put the file code in the ECB for physical read/write operations.

DADNF : is split in two characters:
DADNTF: Number of temporary files assigned to this DAD.
It is incremented each time a temporary file is assigned for this DAD, and decremented when one is deleted.
This word is checked to determine if a BTB can be scratched

in the BTB pool.

DADNPF: Number of permanent files assigned for this DAD. It is incremented each time a permanent file is assigned for this DAD and decremented when one is deleted. These two characters, DADNTF and DADNPF are checked to determine if a DADCT can be scratched.

DADDCT: Address of the DCT of which this DADCT is an extension.

DADNAM: Name of the DAD (6 ASCII characters).

In four situations, the DAD name is involved, depending on the following factors:

- a disc has a DAD structure : D
  a disc does not have a DAD structure : ¬ D
- assign is done by disc number + DAD name : N
  assign is done by disc number only (first DAD): ¬ N

Thus:

	N	¬ N
D	1.	3.
¬ D	2.	4.

1: DAD name is set at time of assignment and checked against the VTOC.

2: assign is refused.

3: DAD name is read from the VTOC

4: DAD name is left as six blanks.

DADINT: Number of interlaces on this DAD.

Fixed per disc type as follows:

X1215: 3

X1216: 3

CDC 40M: } Fixed at Premark.

CDC 80M: }

DADSPT: Number of sectors per track in this DAD.

Fixed per disc type as follows:

X1215: 16

X1216: 16

CDC 40M: 39

CDC 80M: 39

FL 250K: 496

DADSLG: Sector length on this DAD.
Fixed at 410 characters per sector.

DADNBC: Number of cylinders on this DAD.
maximum, per disc type:
X1215: 200
X1216: 400
CDC 40M: 404
CDC 80M: 808
As sector length is at present fixed at 410 characters per sector, the maximum is:
X1215: 200
X1216: 400
CDC 40M: 103
CDC 80M: 103

DADFCL: Cylinder number of the first cylinder of this DAD.

DADSPG: Number of sectors per granule.
Fixed for all disc types at 8 sectors per granule.

DADSTA: DAD number is the sequence number of this DAD on the disc-volume. If 0, it is the first DAD on the volume.

DADBTB: address of the BTB, located in the BTB pool (T:BTP).
This word is used only if a temporary file has been assigned for this DAD.

Logical File Description Table (T:LFT)

This table contains the information required by the system for performing an I/O operation on a logical file.
The table is of fixed length, with one entry for each file. The number of entries is fixed at system generation time and cannot be modified.

LFTORD	X	LENGTH							ORDER					
LFTEAD	USER ECB ADDRESS													
LFTREC	USER RECORD AREA ADDRESS													
LFTLGT	USER REQUESTED LENGTH													
LFTPCT	A5 (PCT ADDRESS)													
LFTLAB	A6 (SCHEDULED LABEL ADDRESS)													
LFTMD1	A	P	S	O	U	C	T	R	SE	FM	E	W	CD	
LFTMD2	ASCNT													
LFTDCT	D:CT ADDRESS													
LFTBOT	ABSOLUTE ADDRESS OF SECTOR GRANTB OF THIS FILE													
LFTSRC	RELATIVE NUMBER (0 ton) OF CURRENT SECTOR													
LFTSAC	ABSOLUTE ADDRESS OF CURRENT SECTOR													
LFTBAD	BLOCKING BUFFER ADDRESS								NBR. OF SECTORS OF DIR. ACC. FILE					
LFTBDS	DISPLACEMENT OF NEXT RECORD IN BLOCKING BUFFER													
LFTBUF	CURRENT BUFFER ADDRESS (FOR CURRENT OPERATION)													
LFTSEC	CURRENT SECTOR TO BE READ OR WRITTEN													
LFTORC	CURRENT ORDER TO BE PERFORMED													
LFTSTC	CURRENT STATUS													
LFTSVD														
LFTSVS														
LFTSLU														
LFTSLB														
LFTSLC														
LFTSLT														
LFTSLR														
LFTLK1														
LFTLK2														
LFTATT														

LFTORD: X: 0, if this is the current entry.
1, if this is the last entry.
LENGTH is the length of one entry, in characters.
ORDER is the I/O order given by the user in ECB word 0.

LFTEAD: contains the address of the user ECB.

LFTREC: is the address of the user record area, where the Data Management routine must read or write the logical record.

LFTLGT: is the requested length as specified in the user ECB. If the requested length is greater than 8 sectors (i.e. 3200-8=3192 characters) incorrect length status will be returned and the record will be truncated at the first 3192 characters. In read mode, if the requested length differs from the record length given when the record was written, the shortest length will be moved from the disc to the record area. If this is equal to the requested length, no incorrect length status will result. Otherwise, when the record is truncated, incorrect length status will be set in the ECB. To avoid this, the user must read with the maximum record length in his request, and write with any length less than 3192 characters.

LFTORD: is used to save the I/O order given in the monitor request.

LFTPCT: contains the address of PCT of requested program.

LFTLAB: contains the scheduled label address. If no scheduled label is used, it contains 0.

LFTMD1 and LFTMD2 are two mode words defining the status of the LFT:

LFTMD1:

A=1: LFT inactive, free for processing a request.
=0: LFT busy. A user request has already been recorded and is not yet terminated.

P=1: file is write-protected.
S=1: source file.
O=1: object file.
C=1: load module.

} These bits are always 0 under DRTM

U=1: undefined file; contains user data.

T=1: temporary file. This bit is set when the file code is assigned. It is used to find out whether the file can be extended and, for Data Management, to read the GRANTB to find the next granule address in case of sequential access.

R=1: random access is used for this file.

SE=1: sequential access is used for this file.

FM=1: the last write operation for this file was a write EOF, so the file is closed.
This bit is not modified by read and rewind operations. When an assignment is made for a catalogued file, this bit is also set to 1.

E=1: the current entry is already in queue. 0, if not.

W=1: write request.

0: read request.

CO=1: consecutive file.

0: non-consecutive file.

LFTMD2:
Bits 0 to 3 contain the number of file codes assigned to this LFT. At system generation time, it is initialized with the value 0. It is incremented each time a file code is assigned to this LFT and decremented when the assignment of a file code changes. Thus, when ASCNT becomes zero again, LFT is free to be used by the assign routine. Bits 4 to 15 give the number of sectors of the file, if it is a consecutive one.

LFTDCT: contains the address of the D:CT of the disc on which this file is stored.

LFTBOT: contains the address of the GRANTB sector of this file (second sector of first granule). It is initialized when an assign request is given and remains unchanged until the file code is deleted.

LFTSRC: is the relative sector number within the file.

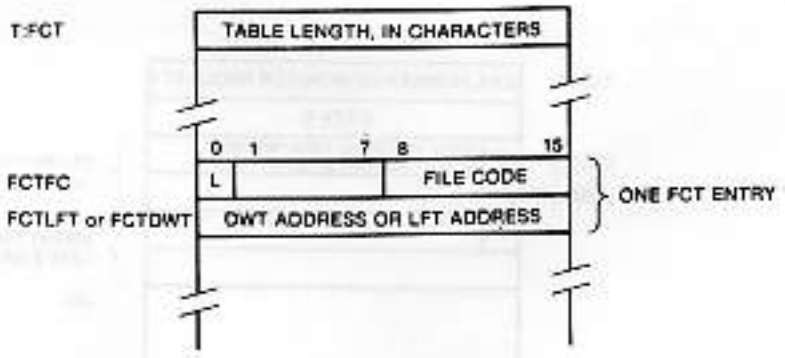
LFTBAD: contains, for sequential files, the current blocking buffer address. For random access files, it contains the number of sectors in the file.

The words LFTBUF to LFTLK2 are used as work area by the logical I/O routines for performing their operations. LFTBUF, LFTSEC, LFTORC and LFTSRC are used by a subroutine which performs the physical part of an I/O operation and handles the request queue. The other words are used as work area.

LFTATT: is the Attach/Detach flag.
If the value is /8000, the file is not attached. If it is not /8000, it contains the PCT address of the program to which the file is attached.

File Code Table (T:FCT)

The file code table establishes the connection between a file code and a physical device or a logical file assigned to that file code. The table is of fixed length and created at system generation time. The user must at that point declare all the file codes used by the system, i.e. /EF, /E0, /02, /F0 to /FX. The others can be assigned when they are needed, but they may also be declared at system generation time.



Each entry consists of two words:

FCTFC: contains the file code, in bits 8 to 15.
L=1 if it is assigned to a logical disc file
L=0 if it is assigned to a physical device.

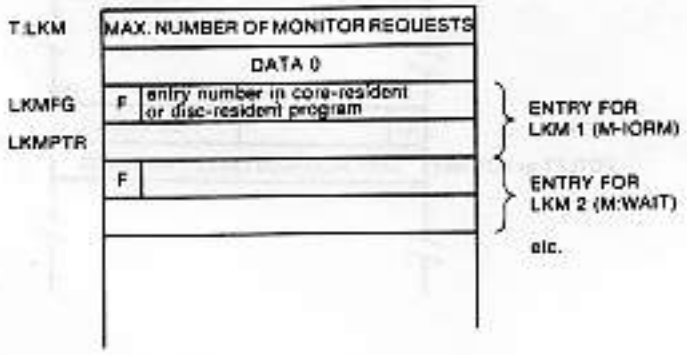
If L=1, the second word of the entry contains the address of the file description table LFT and is called FCTLFT.

If L=0, the second word contains the address of the Device Work Table (DWT) and is called LFTDWT.

- For spare entries in the table, File Code is reset to zero. It can then be used for new file code assignments.
- The second word of an entry is reset to zero to indicate an assignment to NO device. In this case, bit L of the first word is also reset to zero.

Monitor Request Address Table (T:LKM)

This table contains the addresses of the routines which handle the different monitor requests. When a monitor request is made, control first goes to the I:LKM interrupt routine, which then finds the corresponding monitor request handler via this table, in accordance with the DATA number specified in the request. The table is generated at system generation time.
A table entry (each entry occupies two words) reset to zero indicates that the corresponding monitor request is not included in the system.



There must be as many 2-word entries as the number in the location T:LKM specifies (not including the first 2 words).

The words in the entries have the following meaning:

-LKMFG: F=0: the monitor request is processed by a possibly disc-resident program running at level 49 to 62.
The contents of LKMFG must be passed to the routine handling the monitor request via register A3, which is used to identify the entry point in the called program. The sign bit of A3 is set:
- to zero, if the request comes from the main program sequence;
- to one, if the request comes from a scheduled label sequence.

F=1: the monitor request is processed by a memory resident routine running at level 48.

LKMPTR: If F=1, this word points to the start address of the routine processing the monitor request at level 48.
If F=0, the monitor request is processed at a level equal to or greater than 49. In this case, LKMPTR points to the program name (PRNAME) in the Program Control Table of the program which must process this monitor request. If the monitor request is read only, the program is either D:USV1, D:USV2 or D:USV3. If it is memory resident, LKMPTR points to D:RMAC. In the latter case, the table T:RMAC must be updated with the appropriate entry point. (see also below).

Resident Monitor Request Table (T:RMAC)

In the DRTM a number of monitor requests are handled by routines running at level 48:

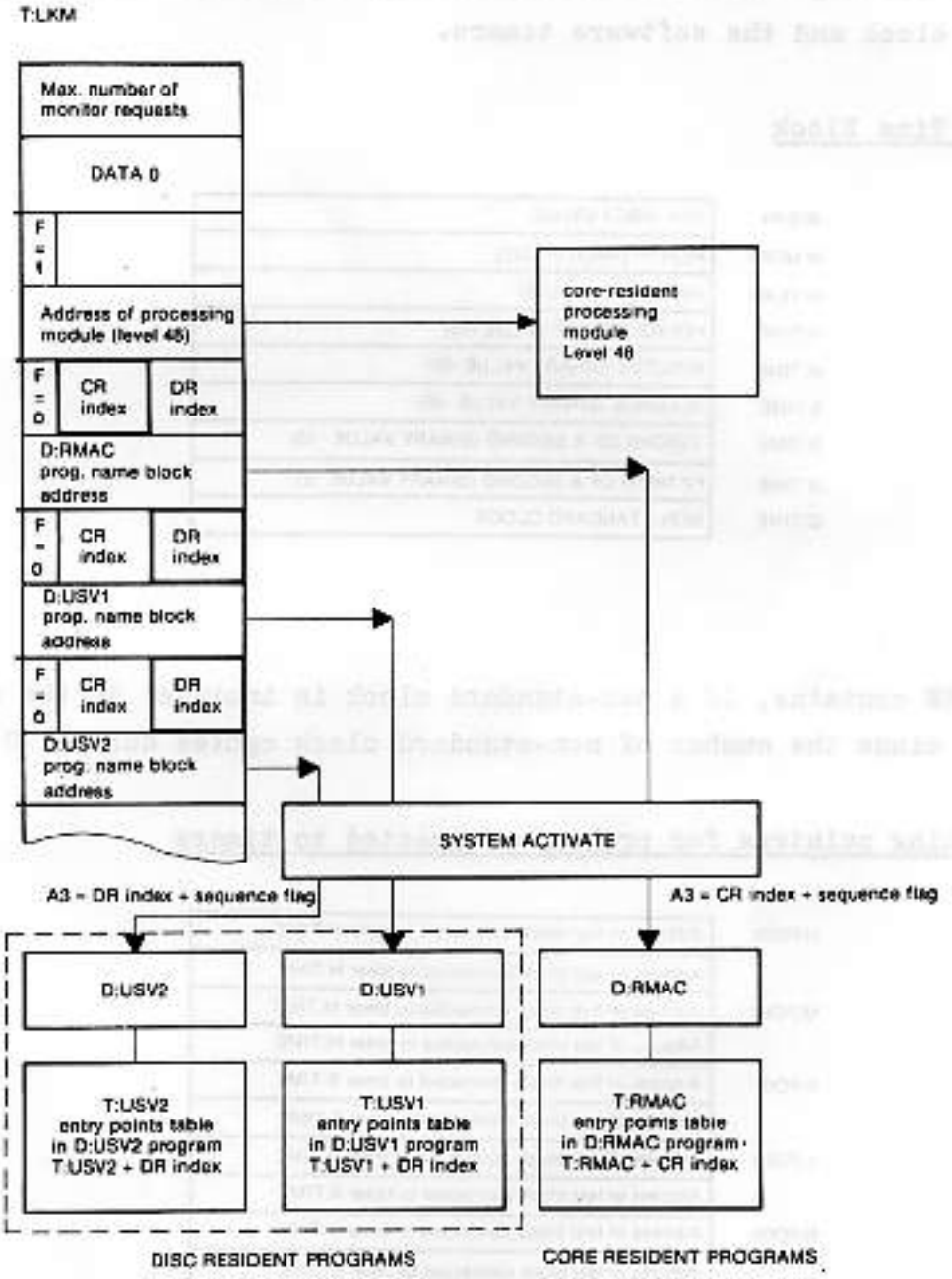
- I/O
- Wait for an Event
- Activate
- Exit
- Get Buffer
- Release Buffer
- Set Event
- Switch inside a Level

The other monitor requests are handled by system programs running at a level equal to or greater than 49. If these monitor requests have been declared read only, they are processed by the system programs D:USV1, D:USV2 and D:USV3.
If they, or any number of them, have been declared as memory resident, they will be processed by a memory resident program D:RMAC. This program is link-edited with the monitor. A table must be created for the D:RMAC program to be able to branch to the correct entry point when one of these monitor requests is encountered in a program. This table must contain the start addresses of the selected memory resident monitor request handlers. The length of this table is variable, depending on the number of monitor requests declared memory resident, but each address must be stored in a specific location in the table:

T:RMAC		
	0	DATA M:DFM (Data Management)
	2	DATA M:NDLG (Data Management)
	4	DATA D:CNTM (Connect Timer)
	6	DATA D:DNTM (Disconnect Timer)
	8	DATA D:GTIM (Get Time)
	10	DATA D:ATDV (Attach Device)
	12	DATA D:DTDV (Detach Device)
	14	DATA D:CNLV (Connect Level)
	16	DATA D:DNLV (Disconnect Level)
	18	DATA D:WGT (Wait for given Time)
	20	DATA ASGPRO (Assign File Code)
	22	DATA DELPRO (Delete File Code)
	24	DATA KEYPRO (Read Key-In)
	26	DATA CANKEY (Cancel Key-In)

Thus, if the monitor request Disconnect a Level is required to be memory resident and any (or none) of the requests preceding it, but not the requests Assign, Delete, Read Key-in, the table must be 9 words long. If only disc logical I/O is required, only the first two words have to be generated.

Note: A similar table is required for the programs D:USV1, D:USV2 and D:USV3. It is described in the paragraph dealing with these programs.


T:LKM
Max. number of monitor requests
DATA 0
F = 1
Address of processing module (level 48)
F = 0
CR index
DR index
D:RMAC prog. name block address
F = 0
CR index
DR index
D:USV1 prog. name block address
F = 0
CR index
DR index
D:USV2 prog. name block address
core-resident processing module Level 48
SYSTEM ACTIVATE
A3 = DR index + sequence flag
A3 = CR index + sequence flag
D:USV2
D:USV1
D:RMAC
T:USV2 entry points table in D:USV2 program T:USV2 + DR index
T:USV1 entry points table in D:USV1 program T:USV1 + DR index
T:RMAC entry points table in D:RMAC program T:RMAC + CR index
DISC RESIDENT PROGRAMS
CORE RESIDENT PROGRAMS

TIMER MANAGEMENT TABLES

The following tables are connected with the management of the real time clock and the software timers.

Real Time Block

W:DAY	DAY (ASCII VALUE)
W:MONT	MONTH (ASCII VALUE)
W:YEAR	YEAR (ASCII VALUE)
H:TIME	HOURS (BINARY VALUE -24)
M:TIME	MINUTES (BINARY VALUE -60)
S:TIME	SECONDS (BINARY VALUE -60)
B:TIME	TENTHS OF A SECOND (BINARY VALUE -10)
A:TIME	FIFTIETH OF A SECOND (BINARY VALUE -5)
C:TIME	NON-STANDARD CLOCK

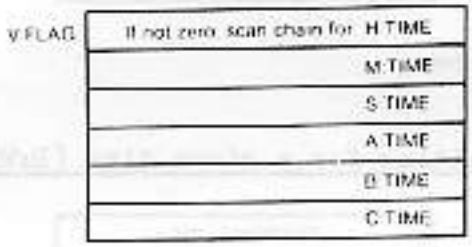
C:TIME contains, if a non-standard clock is included in the system, zero minus the number of non-standard clock cycles during 20 msec.

Chaining pointers for programs connected to timers

H:POIN	Address of first block connected to timer H:TIME
	Address of last block connected to timer H:TIME
M:POIN	Address of first block connected to timer M:TIME
	Address of last block connected to timer M:TIME
S:POIN	Address of first block connected to timer S:TIME
	Address of last block connected to timer S:TIME
A:POIN	Address of first block connected to timer A:TIME
	Address of last block connected to timer A:TIME
B:POIN	Address of first block connected to timer B:TIME
	Address of last block connected to timer B:TIME
C:POIN	Address of first block connected to timer C:TIME
	Address of last block connected to timer C:TIME
R:POIN	Address of first block connected to abs. time HH MM SS
	Address of last block connected to abs. time HH MM SS

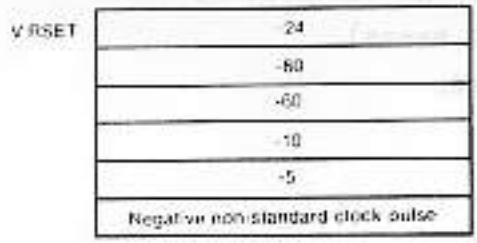
V:FLAG

This is a table built by the I:RTC module.
When a chain of programs connected to a timer must be scanned, I:RTC sets a flag which is to be detected by the timer management module M:DCK2 and reset by it.
If one of the values in the table ≠ 0, the chain connected to that timer must be scanned:



V:RSET

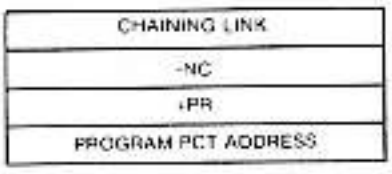
This block contains the values necessary to reset the real time block values.



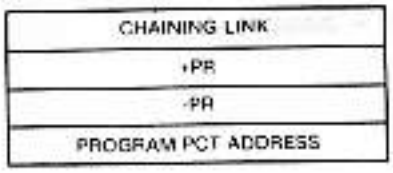
Blocks built by Connect Program to Timer request (D:CNTM module)

Standard Block

Format before first activation of the connected program



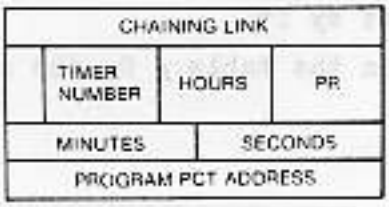
Format after the first activation



NC: number of timer cycles before the first program activation

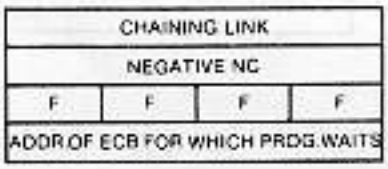
PR: number of timer cycles between two activations of the same program.

Block for programs connected to absolute time



T.N. is the timer number.

Block for programs waiting for a given time (D:WGT module)



Timer Numbering

- With declaration of the 20 msec. clock:
  - 0 - clock timer (20 msecs)
  - 1 - tenths of seconds
  - 2 - seconds
  - 3 - minutes
  - 4 - hours
- With declaration of a non-standard clock:
  - 0 - non-standard clock
  - 1 - 20 milliseconds
  - 2 - tenths of seconds
  - 3 - seconds
  - 4 - minutes
  - 5 - hours

Note: If a Connect Timer request is given (via the System Command Language or in a user program) before the SCL command Set Clock (SC) has been given, the numbering accepted by SCL is the second one.

Input/Output operations involve a number of monitor modules, aside from the Device Work Table, Disc Control Table, Logical File Table and File Code Table already mentioned in the previous chapter.
The most important of these are the I/O drivers, one for each type of peripheral.
Every I/O monitor request is handled by a driver. The user can, if he wishes, write his own drivers and insert them into the system. Such user-written drivers must interface with certain tables and modules which make up the I/O system of the monitor. The necessary information which must be taken into account is described in the following paragraphs.

TABLES

Three tables are used by the I/O system to know the necessary details about the peripheral devices used:

- File Code Table, which enables the user to assign a file code (a logical number) to a device and use this file code in programming. See Chapter 2 of this part.
- Device Work Table: contains all parameters about the peripherals which are necessary for the I/O system to know. See Chapter 2 of this part.
- Controller Status Table or Disc Control Table: one for each device control unit. The CST is a free format table of unspecified length in which the user can put information for use by his I/O driver for non-disc devices. The I/O system knows one word of this table (referred to by word DWT+32) which contains the status of the control unit. The first bit of this status word is reset to 0 (=busy) as soon as an I/O operation for a device is started and set to 1 when it is terminated (1=free).
  For disc, Disc Control Tables are created. The layout is given in Chapter 2.

I/O SYSTEM

The physical I/O system can be divided into four main parts:

- The I/O request module (IORM)
- Driver
- End of I/O module (ENDIO)
- Service routines (COMIO and M:RETR)

IORM

When the user has given an I/O monitor request (LKM 1), this module will receive control, via the I:LKM module and T:LKM table.
First this module

- checks the validity of the request
- increments the event count (PCT)
- computes the Device Work Table address
- checks whether the device control unit is busy or free (via DWT+32). If the unit is busy, the user is put in wait.
- checks whether the device is busy or not. If it is busy, the user is put in wait.

Then IORM sets some parameters in the DWT and in the controller status table:

- controller is set to busy (bit 0 = 1)
- retry flag in controller status (bit 7) is set for non-disc devices
- any scheduled label parameters are set in DWT+28, DWT+30. DWT+24 and DWT+26 are set to zero
- the user Event Control Block is initialized, i.e. the left character of ECB 0 and ECB+8 (status) are set to zero
- the I/O order (A7) is analyzed which may result in initialization of some DWT location.

At the end of this process a branch is made to the I/O driver concerned, the address of which is found in DWT+6.

Driver

On entry into a driver, these registers must contain the following parameters:

A4: User order without wait bit
A7: User I/O order
A8: User ECB address
A6: DWT address.

The DWT must contain:

Words 0 to 8: not modified.
Word 10: User ECB address
Word 12: User buffer address
Word 14: Requested length (may be non-significant)
Word 16: Zero
Word 18: I/O order (without wait or retry bit; may be non-significant, e.g. skip orders)
Word 20: Retry bit
Word 22: not significant.

The first part of the I/O driver performs the I/O initialization. An exit to the C:WAIT module is made with the user order (with wait bit!) in A7 and the user ECB address in A8.
The second part of the driver is made up by the interrupt routine:

- interrupt sequence generated by SYSGEN:

  for single unit controllers:
  ```
  STR A1, A15
  STR A8,A15
  LDKL A6,DWT address
  ```

  for multi-unit controllers:
  ```
  STR  A1,A15
  STR  A8,A15
  LDKL A6,controller status or
          DCTDH (DCT) address
  ```

- the exit from the interrupt routine is made to R:TURN (in ENDIO module) if the I/O is not yet finished or to R:TUR4 (in ENDIO) if it is finished. The exit to R:TUR4 must be made with A2 containing the I/O status and A6 containing the DWT address.

- for Retry procedures a branch must be made:

  ABL M:RETR

  In this case, the calling sequence is:

  A6: DWT address

  A2: Status to be printed

  A1: Retry flag (0, if retry)

  A3: Hardware order (contents of register with which CIO Start will be sent).

Note: The I/O processor control parameters must have been reinitialized.

ENDIO

The calling sequence to be used by user I/O drivers for this module is:

A6: DWT address

A2: I/O software status (see Part 1, chapter 11)

    ABL R:TUR4

Moreover, ENDIO must find the following parameters set in the DWT:

- effective length (word 16)
- ECB address (word 10)
- Scheduled label parameters (words 28, 30).

Then, the following functions are performed by the ENDIO module:

- controller status is updated (bit 0 is reset to 0: free)
- the event count is decremented
- the event character in the user ECB is updated (bit 0=0)
- the effective length is set in the user ECB (word 3)
- the software status is set in the user ECB (word 4)
- a branch is made to the dispatcher (M:DISP), with register A5 containing the PCT address and A6 the scheduled label address, if any.

COMIO

This routine has two functions: executing I/O hardware instructions and building the I/O processor control words and putting a requesting program in Wait State. Below the calling sequences for the functions are listed.
Note, that in these sequences the instructions INH - MSR - ABL must be given in the order specified:

- CIO Start: A6: DWT address
  A2: hardware order
  A3: return address
  INH
  STR A1,A15
  STR A8,A15
  ABL S:TIO
  Return: - condition register set
  - inhibit mode.
- CIO Stop: A6: DWT address
  A3: return address
  INH
  STR A1,A15
  STR A8,A15
  ABL H:LTIO
  Return: - condition register set
  - inhibit mode.
- OTR: A6: DWT address
  A3: Return address
  A1: word to be output
  INH
  STR A1,A15
  STR A8,A15
  ABL O:TRIO
  Return: - condition register set
  - inhibit mode.

- INR: A6: DWT address
A3: return address
INH
STR A1,A15
STR A8,A15
ABL I:NRIO
Return: - condition register set
- inhibit mode
- word or character to be input: in A1.

- SST: A6: DWT address
A3: return address
INH
STR A1,A15
STR A8,A15
ABL S:SST
Return: - condition register set
- inhibit mode
- status in A2.

- the I/O processor control words are built as follows:
A1,A 2: two-word contents of control word
(the user must give this as:
- length (in 2's complement) + 4 control bits
- ending buffer address
This is converted by routine M:TEX to
- length (in 2's complement) + 4 control bits
- beginning buffer address.
A6: : DWT address
A3 : return address
INH
STR A1,A15
STR A8,A15
ABL M:TEX

The requesting program is put in Wait, if this is necessary, as follows:

A7: User I/O order
A8: User ECB address
ABL C:WAIT

M:RETR

This module is involved when the operator wants to retry an I/O operation.
The calling sequence is:
A1: Retry flag (0, if I/O must be retried)
A2: Status to be printed
A3: Request (hardware order for CIO Start)
A6: DWT address
ABL M:RETR

The I/O processor control words must have been reinitialized.
Then the following actions are taken:
- activate D:OCOM with message formatted in buffer
- if the I/O operation must not be retried, branch to ENDIO with the value /8000 in register A2
- if the I/O operation must be retried, test whether bit 7 in the control unit status is set:
  a) if not, set bit 7 and go to the dispatcher
  b) if set, (i.e. M:RETR must be called after an RD command by the operator), reset bit 7 and then branch to ENDIO with the value /8000 in register A2.

When a Retry message is sent by the operator:
(D:OCOM is normally connected to level 49)
- compute CIO and execute it
- reset address of buffer in DWT
- reset effective length to zero in DWT
- reset DWT words 24 and 26 to zero
- reset the effective length and status in the user ECB to zero
- for RD messages: do not reset bit 7 of the control unit status, but EXIT
- for RY messages: reset bit 7 of the control unit status, then EXIT.