3 TOSS SYSTEM START

3.1 General

System start procedure is the initialisation of a PTS Terminal computer for running an application program.
System start procedure comprises the following steps:

- Load the TOSS Monitor into memory
- Load the application into memory
- Read the configuration file and set up the required tables and buffers
- Pass control to the application or to CREBUG if this was included

The TOSS Monitor for a particular Terminal System must have been generated by the DOS utility SYSGEN. This results in a Monitor load module on disk.
Translator, CREDIT linker and linkage editor create an application load module on disk.
Generation of a configuration file is described in section 3.4.6 and 3.4.7.
These modules can then be copied to one or more cassettes by the DOS catalogued procedure $PCAS, or to TOSS formatted disk or flexible disk by $PDISC.

The use of SYSGEN, CONGEN, $PCAS and $PDISC is described in the DOS6800 System Software PRM M11.

3.1.1 *System load program SYSLOD*

To make the most economical use of core storage, a dynamic configuration procedure is included in the system software. This procedure SYSLOD is a software module which is linked to the TOSS Monitor.
After loading of the Monitor by the initial program loader, control is passed to SYSLOD. SYSLOD then loads the application load module from disk, cassette or flexible disk. After that SYSLOD reads the configuration file, and performs monitor and application configuration.
SYSLOD then passes control to the application or to CREBUG.
Configuration by SYSLOD must always take place, it is not possible to load an already configurated Monitor.

3.1.2 *Monitor configuration*

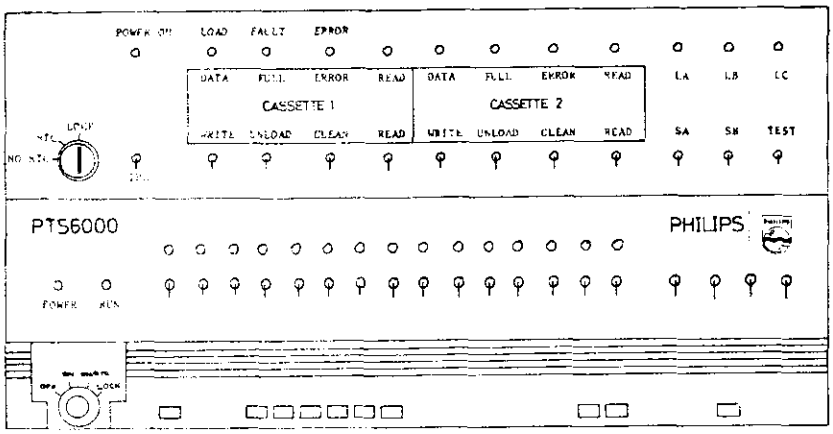
Monitor configuration is the process whereby the general Monitor created by SYSGEN is adapted to the specific hardware environment.
Monitor tables and buffers are built and the Monitor is supplied with pointers to the tables. In the case of PTS 6813 with MMU, the MMU tables are set up and an extra number of I/O buffers is reserved. Buffers for Data Communication and data management are also generated during system configuration.
If segmented application, the segment and page tables controlling the segments and pages in run time, are generated.

3.1.3 *Application configuration*

After the monitor configuration process, SYSLOD starts the application configuration. Terminal control areas, terminal work blocks, user work blocks, swappable work blocks, terminal stacks and data-set buffers are generated. In case of a system with MMU, the MMU tables are extended with references to the data division and the application. After application configuration, SYSLOD starts the system by queueing all tasks in the dispatcher queue and gives control to the interpreter.

3.2 Loading procedures

Monitor, application load module and configuration file can be loaded from cassette, disk or flexible disk.
Application and configuration data should reside on the same device type. If loading from disk, Monitor and application load module should be on the same volume.
Loading is controlled from the SOP or from the full panel and SOP if present.



System Operator's Panel

Full Panel

3.2.1 *SOP lamps*

During loading, the following SOP lamps indicate:

- SOP lamp 1 an application is loaded by SYSLOD
- SOP lamp 2 input error
- SOP lamp 3 memory overflow
- SOP lamp 4 format error
- SOP lamp 5 terminal ident error
- SOP lamp 6 user-or swappable work block error
- SOP lamp 7 MMU table overflow
- SOP lamp 8 illegal page size

3.2.2 *Program loading from cassette*

Monitor, application and configuration data can be loaded from the same or from different cassettes.

The procedure for loading from cassette is:

- Ensure that power is switched on at the Terminal computer and at each peripheral device.
- Ensure that the real time clock is on.
- Place Monitor cassette in a cassette drive.
- Press SOP switch IPL
  or where a full panel, press RST, MC, IPL in that order.
- Press SOP switch 1 for the left-hand cassette drive
  or SOP switch 2 for the right-hand cassette drive.
- Wait for loading indicator SOP lamp 1 to turn off.
  If the next module to be loaded is not on the same cassette, this cassette is now unloaded. Place the next cassette in a cassette drive and press the corresponding SOP switch.

– Turn key to Lock position.
– When loading is completed the application or CREBUG will be started.

3.2.3 *Program loading from disk*

Monitor and application must be loaded from the same volume.
The configuration file may be on the same or on a different disk or on flexible disk.

The procedure for loading from disk is:

– Ensure that power is switched on at the Terminal computer and at each peripheral device.
– Ensure that the real time clock is on.
– Place the Monitor disk on a disk drive, press the START button and wait for the READY lamp to light up.
If loading from the fixed disk, a disk cartridge must still be mounted.
– Press SOP switch IPL
or where a full panel, press RST, MC, IPL on the full panel.
– Press SOP switch 3 for loading from fixed disk,
or SOP switch 4 for loading from cartridge disk.
– If there is more than one application program on the disk, all SOP lamps will light up. Select the application to be loaded by pressing the corresponding SOP switch. Monitor and application are now loaded.
If during system generation (SYSGEN) was specified that there is only one application program on the disk, Monitor and application are now loaded automatically.
– If the configuration file is on the same volume as the Monitor and the application, SYSLOD automatically reads in the configuration data.
– If the configuration file is not on the same volume as the Monitor and the application, the eight leftmost SOP lamps will light up. Press the SOP switch corresponding to the device from which the configuration file must be read:

SOP switch 3 fixed disk
SOP switch 4 cartridge disk
SOP switch 5 flexible disk 0 multiplex channel
SOP switch 6 flexible disk 1 multiplex channel
SOP switch 7 flexible disk 0 programmed channel
SOP switch 8 flexible disk 1 programmed channel

– Turn key to Lock position.
– When loading is completed the application or CREBUG will be started.

3.2.4 *Program loading from flexible disk*

Monitor and application must be loaded from the same volume.
The configuration file may be on the same volume or on a fixed or cartridge disk.

The procedure for loading from flexible disk is:

– Ensure that power is switched on at the Terminal computer and at each peripheral device.
– Ensure that the real time clock is on.
– Place the flexible disk in disk drive 0 or 1 and wait for the READY lamp to light up.

- Press SOP switch IPL
  or where a full panel, press RST, MC, IPL on the full panel in that order.
- Select the flexible disk drive from which is to be loaded by pressing the corresponding SOP switch:

  SOP switch 5 flexible disk 0 multiplex channel
  SOP switch 6 flexible disk 1 multiplex channel
  SOP switch 7 flexible disk 0 programmed channel
  SOP switch 8 flexible disk 1 programmed channel

- If there is more than one application load module on the flexible disk, all the SOP lamps will light up. Select the application to be loaded by pressing the corresponding SOP switch. Monitor and application are now loaded.
  If during system generation was specified that there is only one application on the flexible disk, Monitor and application are loaded automatically.
- If the configuration file is not on the same volume as the Monitor and the application, the eight leftmost SOP lamps will light up. Press the SOP switch corresponding to the device from which the configuration file must be read:

  SOP switch 3 fixed disk
  SOP switch 4 cartridge disk
  SOP switch 5 flexible disk 0 multiplex channel
  SOP switch 6 flexible disk 1 multiplex channel
  SOP switch 7 flexible disk 0 programmed channel
  SOP switch 8 flexible disk 1 programmed channel

- Turn key to Lock position.
- When loading is completed the application or CREBUG will be started.

3.3 Program file layout

3.3.1 *Program file on cassette*

A cassette with the TOSS Monitor load module, the application load module and the configuration file is created with the DOS utility $PCAS. Application data may optionally be put behind these modules on the same cassette.

Layout of the program cassette:

TM	IPL MONITOR SYSLOD	TM	APPLICATION	TM	CONGIF FILE	TM	APPL DATA	TMTM

The different modules may also be on different cassettes. The loading program SYSLOD checks after every module if there is a double tape mark (TM).
In that case the cassette is unloaded and loading continues when the next cassette has been mounted and the corresponding SOP switch is pressed.
If no double tape mark is found after a module, loading continues from the same cassette.
Application data is not read by SYSLOD but has to be read by the application itself.

3.3.2 *Program file on disk*

A disk or flexible disk containing the TOSS Monitor load module, the application load module and the configuration file is created with the DOS utility $PDISC.
The disk or flexible disk must be TOSS formatted. The load modules should be on an L file, configuration data and, optional, application data on an S file.
The configuration file need not be on the same volume as Monitor and application.

3.4 Configuration file

SYSLOD performs configuration at load time, according to configuration data for the specific environment. These data are read from a configuration file which was first created under DOS.
The configuration file can be copied to cassette by $PCAS or to an S file on a TOSS formatted disk by $PDISC.
The configuration file should be on the same medium type as the application.

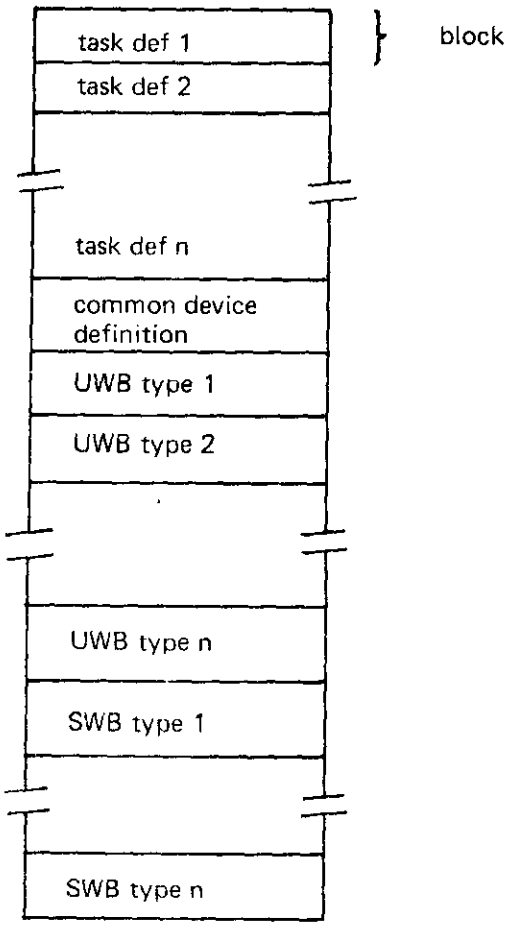
Requirements for the configuration file on disk:

- record length 9 and blocking factor 40 must be specified during CRF.
- File name should answer the format:

  $xxxx:nn
  where: x = alfanumeric character
  nn = 2 numerics representing the SOP switch number of the application (01 to 10).

The configuration file can also be created as an UF file on disk by standard DOS utilities and then moved to the load medium by $PCAS or $PDISC.

3.4.1 *Configuration file layout:*



3.4.2 *Task definition block layout*

Task definition block should have the following format:

DESCRIPTION	FORMAT	REMARKS	EXAMPLE
Block type	T;	Task block	T;
Number of tasks	nn;		06;
Task ID start value	TID=xn;		TID=F3;
Terminal class	TCL=xn;		TCL=F3;
Task level	LEV=nn;		LEV=60;
Number of terminal device classes	nn;	00 if no terminal device classes	03;
Terminal device class	TCD=Tnn;		TDC=T01;
Line connection	LC=nn{L/R};	number of channel unit, local or remote	LC=12L;
	TDC=Tnn;	repeat TDC and LC for each term. dev. class in the task	TDC=T02;
	LC=nn{L/R};		LC=02R;
			TDC=T03;
			LC=06L;
Number of special device classes	nn;	00 if no special device classes	02;
Special device class	SDC=Snn;	repeat SDC for each special dev. class in the task	SDC=S01;
			SDC=S02;

x = alphabetic character
n = numeric

3.4.3 *Common device definition block layout*

DESCRIPTION	FORMAT	REMARKS	EXAMPLE
Block type	C;	Common block	C;
Number of terminal device classes	nn;	00 if no terminal device classes	02 ;
Terminal device class	TDC=Tnn;		TDC=T02 ;
Line connection	LC=nn L; R	number of channel unit, local or remote	LC=03L;
	TDC=Tnn;		TDC=T03;
	LC=nn L; R	repeat TDC and LC for each term. dev. class.	LC=02R;
Number of special device classes	nn;	00 if no special device classes	03;
Special device class	SCL=Snn;	repeat SCL for each special dev. class	SCL=S04; SCL=S05; SCL=S06;

3.4.4 *User work block type definition block layout*

DESCRIPTION	FORMAT	REMARKS	EXAMPLE
Block type	U;	User work block	U;
Number of UWB types	nnn;		003;
Name of UWB	xxx;		UB1;
Number of blocks	nnn;	Number of blocks of this type	003;
		Repeat name and number for each UWB type	UC1; 005; UD1; 002;

3.4.5 *Swappable work block type definition block layout*

DESCRIPTION	FORMAT	REMARKS	EXAMPLE
Block type	S;	Swappable work block	S;
Number of SWB types	nnn;		002;
Name of SWB	xxx;		SB1;
Number of blocks	nnn,	Number of blocks of this type	003;
		Logical name and number for each SWB type	SC1; 004;

nn; number of tasks, in the same terminal class.

TID=xn; task identifier
Within every task definition block the first task has the specified ID, the next task has the specified value added by one and so on.
The CREDIT debugger must have a TID=TB.

TCL=xn; Terminal class to match, relates to the task identifier defined in the TERM statement in the data division. When TID and TCL have different names, the monitor task tables and application terminal control areas are configured with task identifiers according to the TID start value. TCL indicates that the current value in the TERM statement will not be used, when different from the start value in TID.

LEV=nn; priority level of the task is defined. Application tasks run normally on priority level 60, only the CREDIT debugger runs on a higher priority, level 55.

nn; number of terminal device classes used within this terminal class. Terminal device classes are defined at system generation time (SYSGEN). Devices contained in these terminal device classes are connected to a channel unit local terminals (CHLT) or channel unit remote terminals (CHRT).

TDC=Tnn; the terminal device class name as defined at system generation time, must be specified.

LC=nn L; line connection, local (L) or remote (R), to which the devices mentioned in the previous terminal device class (TDC=Tnn) are connected. Line connections are also added by one for each new task within this terminal class.

nn; number of special device classes used within this terminal class. Special device classes are defined at system generation time (SYSGEN).
These devices are not connected to a channel unit local (CHLT) or channel unit remote (CHRT).

SDC-Snn; special device class name as defined at system generation time (SYSGEN), must be specified.

C; is the start of the common device definition block.

. nn; number of terminal device classes used common by all terminal classes.

. TDC=Tnn; the terminal device class name as defined at system generation time, must be specified.

. LC=nn L; line connection, local (L) or remote (R), to which the devices mentioned in the previous terminal device class (TDC=Tnn) are connected. Line connections are also added by one for each new task within this terminal class.

nn; number of special device classes containing devices to be used as common device. (For all tasks in the application.)
SCL-Snn; the special device class name as defined at system generation time (SYSGEN) must be specified.
U; is the start of the user work block definition block.
nnn; number of different user work block types.
xxx; name of the user work block, corresponding to the one defined in the data division.
nnn; number of copies wanted, of the previously defined user work block.
S; is the start of the swappable work block definition.
nnn; number of different swappable work blocks types.
xxx; name of the swappable work block, corresponding to the one defined in the data division.
nnn; number of copies wanted, of the previous defined swappable work block.

The line connections are also added by one for each new task within a group. The terminal device class ID and the special device class ID are specified by SYSGEN.

3.4.6 *Generating a configuration file on cassette*

The procedure for generating a configuration file on cassette without using $PCAS is shown in the following example:

First insert a cassette in one of the cassette drives, then key in the following sequence:

```
(i)    ASG   /E1,TY10
(ii)   RDA   /OA
(iii)  T;
       02;
       TID=B0;
       TCL=B0;
       LEV=60;
       02;
       TDC=T01;
       LC=1L;
       TDC=T02;
       LC=4L;
       01;
       SDC=S01;
(iv)   C;
       00;
       02;
       SCL=S02;
       SCL=S03;
(v)    U;
       002;
       UB1;
       004;
       UC1;
       005;
(vi)   S;
       001;
       SB1;
       005;
(vii)  :EOF
(viii) REW   /03
(ix)   WEF   /03
(x)    PCH   /0A
(xi)   WEF   /03
(xii)  ULD   /03
```

Explanation:

(i) Assign file code E1 (source input) to console typewriter.
(ii) Read data from the source input device (typewriter) and transfer to temporary disk file /0A.
(iii) Key in data for the task definition block. (May be repeated for other terminal classes.)
(iv) Key in data for the common device definition block.
(v) Key in data for the user work block type definition block.
(vi) Key in data for the swappable work block type definition block.
(vii) End of the read data command.
(viii) Rewind the cassette (unit number 1).
(ix) Write a tape mark on the cassette.
(x) Write the contents of the temporary disk file /0A to the cassette (inclusive a tape mark).
(xi) Write a second tape mark on the cassette.
(xii) Unload the cassette.

3.4.7 *Generation of a configuration file on a DOS disk*

The procedure for generating a configuration file on a DOS disk is shown in the following example. With the catalogued procedure $PCAS the configuration file can be copied to cassette.

```
(i)   ASG   /E1,TY10
(ii)  RDA   /20
(iii) T;
      02;
      TID=B0;
      TCL=B0;
      LEV=60;
      02;
      TDC=T01;
      LC=1L;
      TDC=T02;
      LC=4L;
      01;
      SDC=S01;
(iv)  C;
      00;
      02;
      SCL=S02;
      SCL=03;
(v)   U;
      002;
      UB1;
      004;
      UC1;
      005;
(vi)  S;
      001;
      SB1;
      005;
(vii) :EOF
(viii) KPF /20,CONFIG
(ix)  $PCAS C=CONFIG
```

Explanation:

(i) Assign file code E1 (cource input) to console typewriter.
(ii) Read data from the source input device (typewriter) and transfer to temporary disk file /20.

(iii) Key in data for the task definition block.
(May be repeated for other terminal classes.)
(iv) Key in data for the common device definition block.
(v) Key in data for the user work block type definition block.
(vi) Key in data for the swappable work block type definition block.
(vii) End of the read data command.
(viii) Keep the file as library file.
(ix) With $PCAS the configuration data can be written to cassette

(For details see PRM DOS6800.)

3.4.7 *Errors during system run time*

For indicating the type of error, causing a program halt, the following SOP lamps are lit before the execution of the program is stopped.
SOP lamps are numbered from 1–11, number 1 being the left-most lamp.

SOP lamps lit					Significance
7	8	9	10	11	
			x	x	No currency buffer available
		x		x	Illegal interrupt
		x	x	x	Stack overflow
	x			x	Instruction not accepted SST, OTR or INR not accepted due to hardware error
	x		x	x	No blocks available
	x	x		x	Invalid instruction (trap)
	x	x	x	x	Requested LKM processor missing
x				x	Data management (SYSGEN) error