13.1 GENERAL

I/O drivers perform device control functions and (optionally) process data communications and data management I/O requests. The type of function to be performed is specified in register A7 by the requesting task. Examples are as follows:-

/00	Test Status	/05	Basic Write
/01	Basic Read	/06	Standard Write
/02	Standard Read	/0A	Random Read
/03	Numeric Read	/0B	Random Write

Further parameters are specified in an 'Event Control Block'.

The address of the event control block is placed in register A8 by the current task.

A separate driver is available for each type of device (for example boxes 23 to 33 in figure 12.1) and some devices may have more than one driver (for example a keyboard). The required drivers must be built into the monitor during system generation. Separate drivers are also available for performing device control (e.g. 35) at either local or local and remote work positions comprising one or more of the following devices:-

- * Keyboard
- * Teller Terminal Printer
- * General Printer
- * Numeric Display
- * Indicator Display/Keyboard Lamps
- * Video/Plasma Display

These devices must be connected to the Terminal Computer via a Channel Unit for Local Terminals (CHLT) or a Channel Unit for Remote Terminals (CHRT). Devices which are used remotely (i.e. via modems) must be connected to the CHRT.

For 6800 systems one of two drivers may be used to control devices attached to the CHLT or CHRT.
The situation is somewhat different for PTS 8000; see section 13.4, which also describes the driver switch module for DRAS01 and DRSL01 in PTS 6800 systems.

Driver DRLT01 is used to control devices attached to a CHLT. Driver DRRT01 is used to control devices attached either to a CHLT or a CHRT. That is, driver DRLT01 controls locally connected devices only, and driver DRRT01 controls both locally and remotely connected devices. Only one of these drivers is included in the Monitor during system generation.

Drivers DRLT01 and DRRT01 are normally entered only from the individual device drivers for the above devices (for example boxes 23 and 24). The only exception to this rule is the 'test remote line' function. In this case the driver DRRT01 is entered directly from the LKM processor (box 20).

Drivers DRLT01 and DRRT01 terminate by branching to the Dispatcher (box 38).

The drivers for the remaining devices not listed above (for example box 25) are self contained. That is, they do not enter any additional driver. These drivers all terminate by branching to the Dispatcher.

Separate drivers are also available for each type of communication line discipline. These drivers terminate by branching to the Dispatcher.

I/O drivers for devices and data communications are only included in the Monitor if they are specifically requested during system generation.

A separate driver, TIODM, is available for data management (box 22). This 'driver' activates a special data management task (box 42) to process data management requests. This task issues disk I/O requests. The data management driver is only included in the Monitor if data management is requested during system generation. A similar situation exists with file management (boxes 21 and 41).

13.2 DRIVER STRUCTURE

An I/O driver in TOSS consists in general of the following parts:-

* Activation of request
* Interrupt handler
* Procedure handler
* Termination of request
* Abortion of request
* Power on Initialization

Some of these are optional but the activation and termination parts, for instance, are always present.

In most cases the parts can all be regarded as interrupt handlers, internal (LKM) or external.

Only if the driver has to perform a lot of processing may a task be activated from the driver, for example SNA handling in the DC driver for IBM 3600 emulation.

The connection from the application program to the device (data-set) is done via a file code which is used in scanning the current task table (TTAB) for the corresponding device work table (DWT).

In some cases one DWT may serve several devices, the file code being then connected to a DWT plus an index (for example a magnetic tape controller for eight tape stations).

Only one request per DWT is handled by the driver, and the driver may simultaneously handle several requests for different DWT's. The DWT is also used as a queue element.

The driver always works with the DWT and the file code is never seen. The first part of the DWT is standard, the rest is driver dependent.

13.2.1 Activation of Request

When the application executes a monitor request for I/O (LKM DATA 1), control is first given to a common I/O request handler (TIO) which performs some standard functions such as looking up the DWT, checking if the device is free, queueing, etc.

The activation entry in the driver is found via the DWT (DWTADR). It is then called with all information necessary to perform the request passed in the DWT (DWTOR).

The activation part in the driver normally ends by starting up some hardware action, and a branch to the dispatcher (TDISP) is then made.

13.2.2 Interrupt Handler

The interrupt handler is called when a control unit is ready with a data or status transfer, and has raised an interrupt. The CPU saves the current program counter (P) and status word on the A15 stack and gives control to the interrupt handler, which is found via the appropriate entry in the interrupt vector table. Registers A1 thru A8 are saved by the driver.

After executing the proper I/O instructions, either a return to the interrupted program is made, or a branch to the procedure handler, or a call to the termination part.

13.2.3 Procedure Handler

The procedure handler can be viewed as a higher level interrupt handler. For example, in a data communications driver the interrupt handler assembles a complete message and passes it to the procedure handler.

After performing the necessary actions the procedure handler either returns to the interrupted program, or calls the termination part.

13.2.4 Termination of Request

When the request has been executed (for example a message received or transmitted), a monitor routine (TENDIO) is called. This routine checks if the device queue is empty. If not, than the first task in the device queue is put into the dispatcher queue. After this, the task which completed the I/O (with no wait) is put into the dispatcher queue.

The driver then exits to the dispatcher (TDISP) to return to the interrupted task, or to start the task that had its I/O terminated, depending on which has the highest priority.

13.2.5 Abortion of Request

If the application wants to abort a previously issued I/O request, a special monitor request is available. This may happen when, for example, three I/O requests with no wait are outstanding and the others may be aborted on completion of one.

The monitor request handler calls the driver at an abort entry point found via the DWT, and then calls TENDIO to terminate the request, indicating in the return code that it was aborted.

Only a few TOSS drivers support the abort request (for example keyboard, badge card reader, and data communications).

13.2.6 Power-On Initialization

After power-on or system load, each driver is called at a special entry point where recovery and initialization functions are performed. These entry points are contained in a table, PFTAB.
I/O operations which were active immediately before the break are repeated, except for keyboard operations (see SYSGEN).

13.2.7 Support Routines

One support function is available to drivers in the monitor. This is Start a Timer (SETTIM).

It is possible for the driver to start a timer which, after a specified time, will activate a specified entry point in the driver.
This code in the driver is treated as being part of a monitor task and the time is given in multiples of 100 ms.

13.2.8 Environment

* Driver Entry Points:-

  - ACTIVATE
  - ABORT (optional)
  - POWER-ON
  - INTERRUPT HANDLERS

* Driver External Points:-

  - DISPATCHER (TDISP)
  - ENDING OF I/O REQUEST (TENDIO)
  - TIMER FUNCTION (SETIMP)
  - QUEUE MONITOR TASK (QMJOB)
  - DEVICE WORK TABLES (DWT)

13.2.9 Register Usage Conventions

The dispatcher (TDISP) must be called with eight registers (A1-A8) from the current active task on the A15 stack. (The idle loop does not use any registers).

* The activation part may use any register (A1-A14).

* The other parts may use any registers that are saved at the interrupt and restored at return. (A1-A8 are automatically restored when calling TDISP).

13.2.10 I/O Request Flow

When a task performs an I/O operation, control is passed to the TOSS monitor (see figure 13.1).

The TOSS monitor starts the I/O operation (e.g. with wait) on the requested device. When the I/O is started, a check is performed to see if another task can be dispatched (within priority level).

After completion of the I/O, the task is scheduled for dispatching on a First-In-First-Out basis within priority. When the task is dispatched again, it will continue after the I/O instruction.

The detailed actions performed and the TOSS modules involved are as shown in figure 13.1. The numbers in the figure are explained below:-

- Task A0's code is interpreted (e.g. CREDIT code) and an I/O instruction (e.g. KI) is executed (1).

- The interpreter passes control to the TOSS monitor with the LKM DATA 1 instruction. Control is given to the TOSS module IHLKM. Because of the I/O operation, control will be passed to module TIO (2), which searches for the corresponding device worktable (DWT) and the activation address of the driver.

- Control is passed to the activation part of the driver (3). The device driver starts character transfer via the DRRT01 or DRLT01 driver and passes control to the dispatcher module TDISP (4).

- The next task within priority is dispatched, e.g. Task A1 (5). While Task A1's code is being interpreted, and A1 has not yet requested I/O, an interrupt is raised due to reception of a character via DRRT01 or DRLT01 (6).

- Normal interrupt actions are performed and control is passed to the interrupt handler in the device driver (7).

- After the character is stored in a buffer, the dispatcher module (TDISP) gets control (8) and the interpreter continues interpreting the code for Task A1.

- On completion of the I/O operation, the terminating part of the driver is activated, which calls the module TENDIO (9). This module checks if other tasks have outstanding requests for this device (device queue). If so, the queued task will be scheduled and then the task which completed its I/O will be scheduled. Control is then passed to the dispatcher module (TDISP).

Figure 13.2 shows the dispatching sequence in more detail.

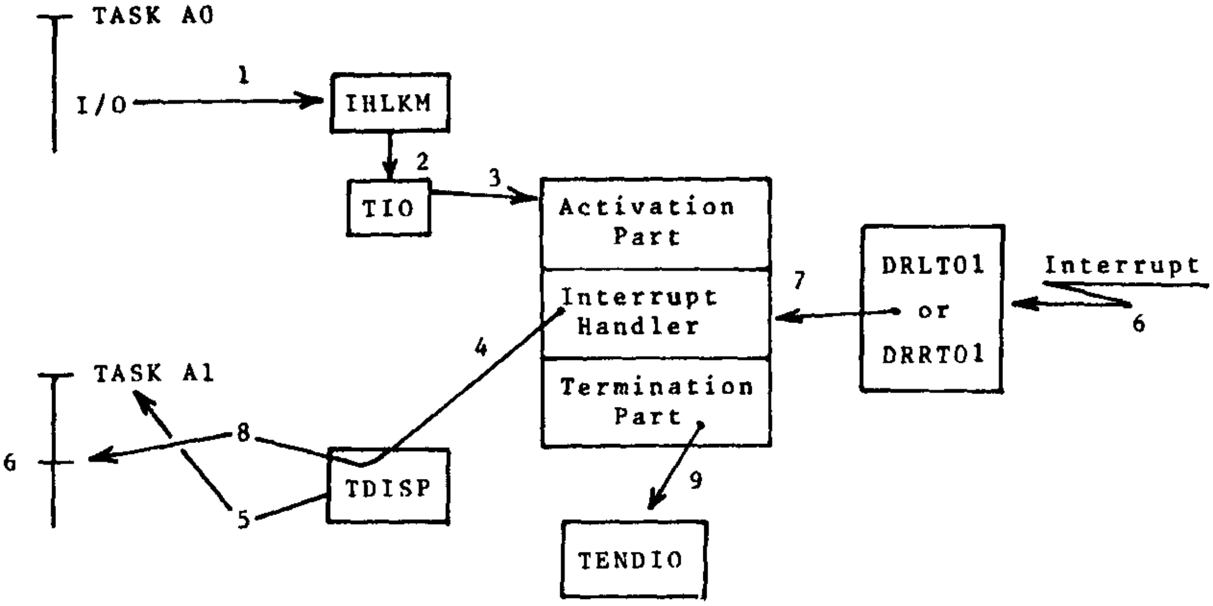

Figure 13.1. TOSS I/O Request Flow.

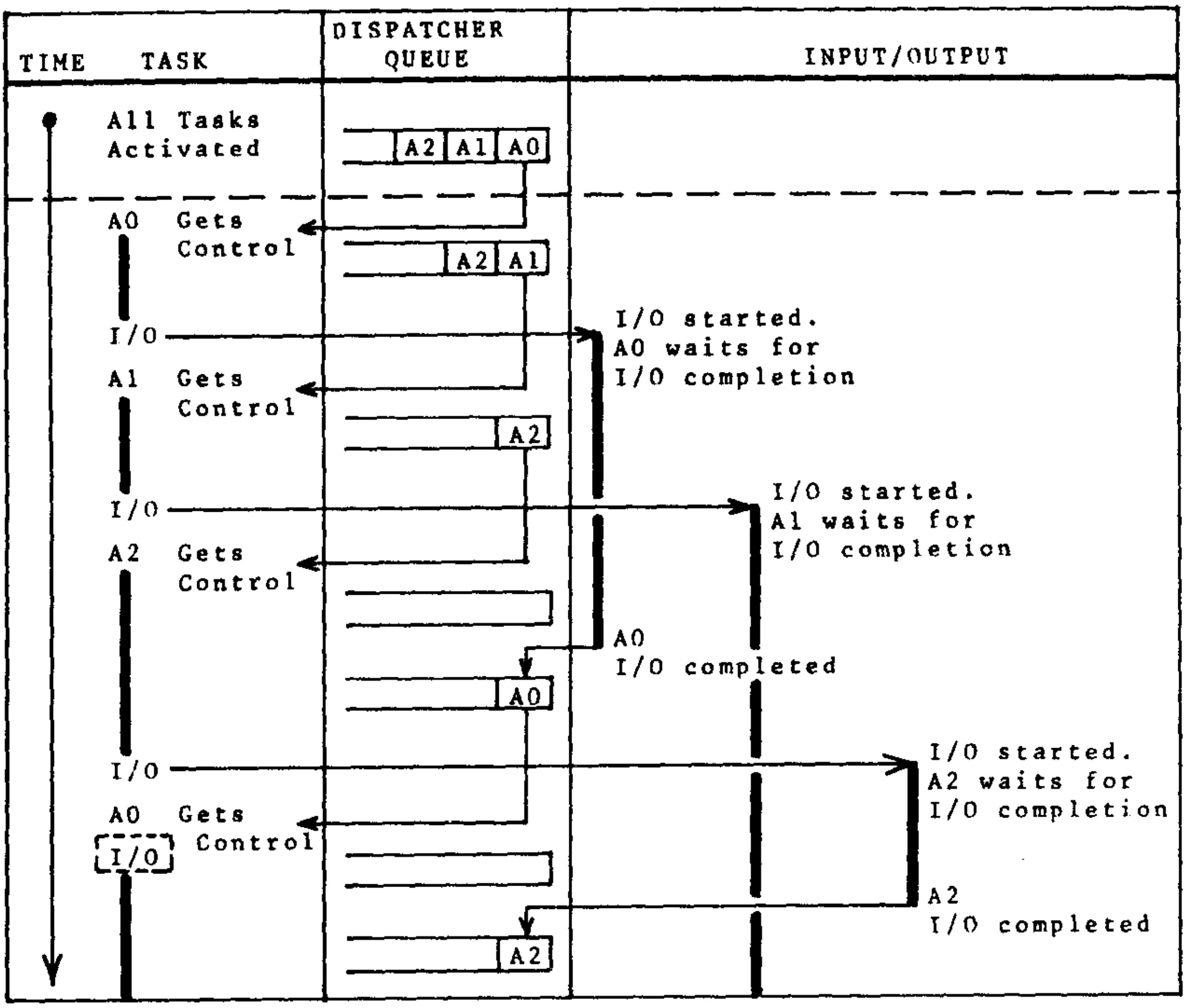

Figure 13.2. TOSS I/O and Dispatching.

13.3 DRIVER FOR LOCAL AND REMOTE TERMINALS (DRRT01)

13.3.1 General

The system software handling workstations either locally or remotely connected to a PTS computer is divided into two parts, the channel unit driver and the device drivers. See figure 13.3.

When only local workstations are configured, the driver for local workstations (DRLT01) should be part of the monitor. This driver has no interface to the application software, only to the device drivers.

When both local and remote workstations are configured, the channel unit driver for local and remote workstations (DRRT01) must be included in the monitor. In this case DRLT01 must be excluded. Only one of the drivers DRLT01 and DRRT01 can be present in the monitor. DRRT01 has only one point of connection to the application software, the test remote line function. DRRT01 has connections to:-

* Application software (test remote line)
* Device drivers
* CHRT/CHLT hardware

The main purpose of DRRT01 is to supervise the transmission of characters between device drivers and Channel Units. A message reporting the result of any transmission is sent to the device drivers by DRRT01.

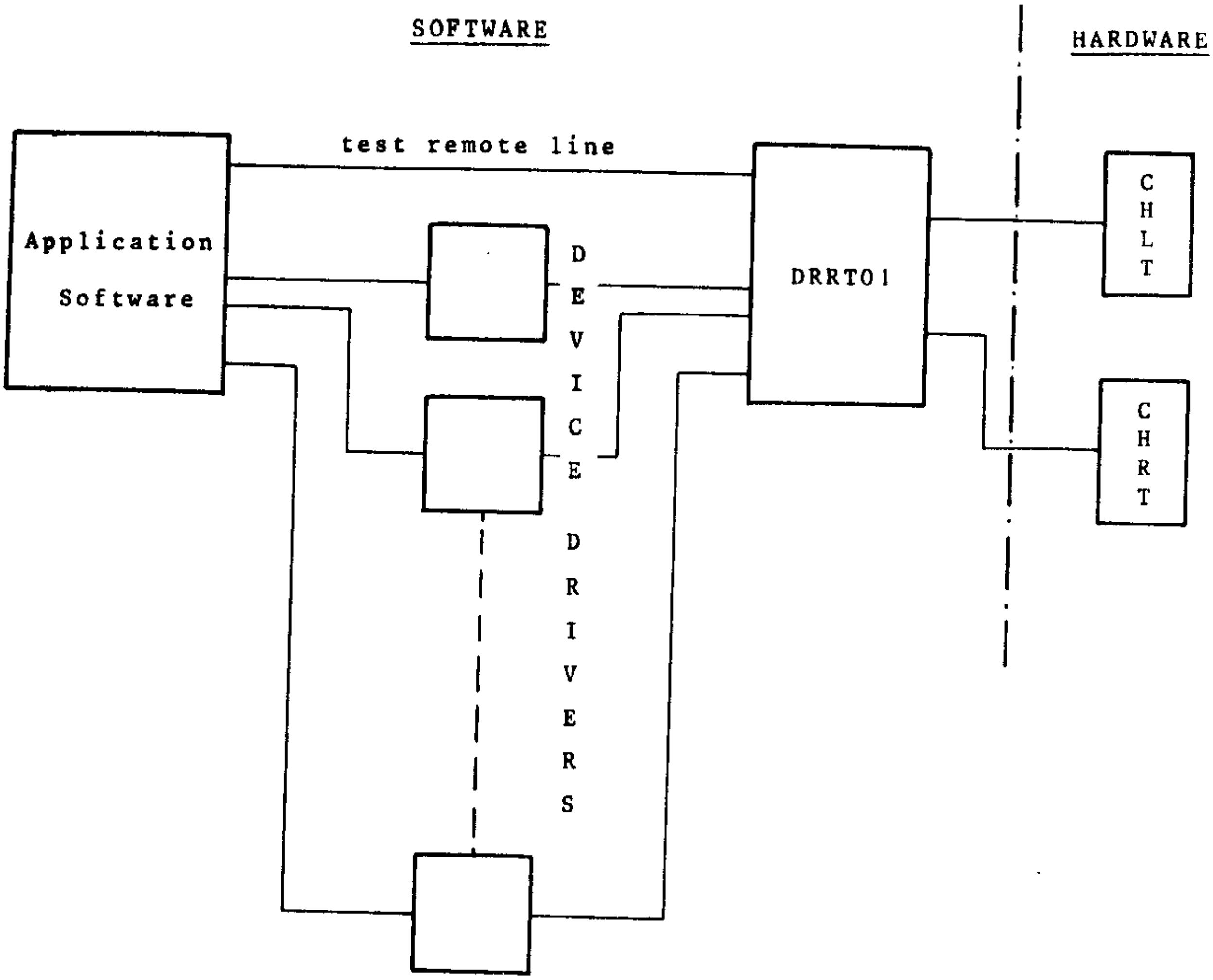

Figure 13.3. Driver Interfaces.

Figure 13.4 shows the system software actions carried out when an LKM request occurs. At activation the device driver orders DRRT01 to send or receive a character. DRRT01 executes the command and, when the interrupt comes (output control character), checks it and gives control back to the device driver with a message concerning the result of the input or output transfer. The device driver then orders input or output until all requested characters are transferred, or an error occurs.

The device work table (DWT) serves as a communication link between DRRT01 and device drivers.

See also figure 13.6 which shows control block relationships.

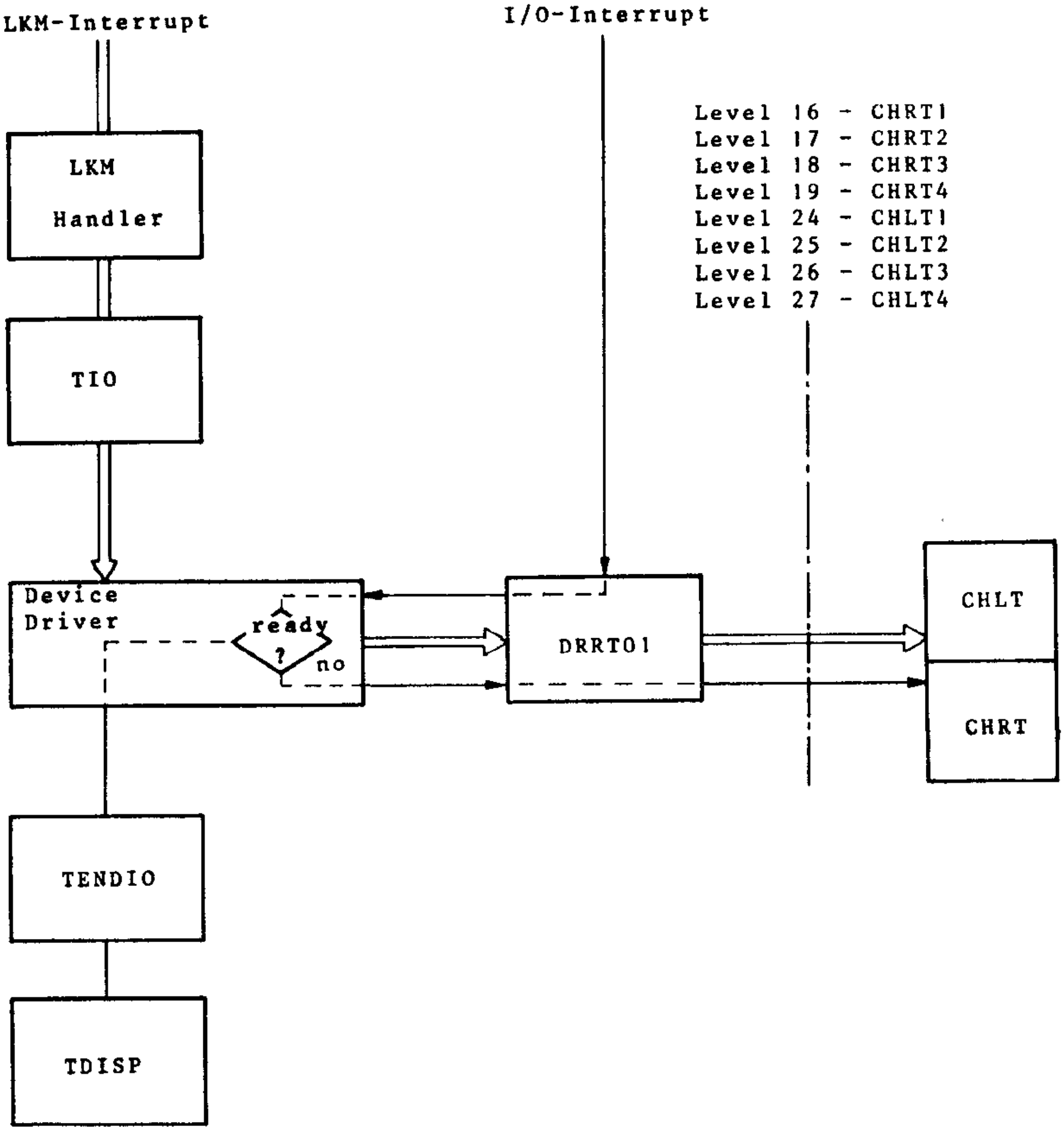

Figure 13.4. LKM Request Sequence.

The following abbreviations are used in the description:-

		Hex Codes (Right Byte)
DRRTO1	DRiver for local and Remote Terminals	
ACK	ACKnowledgement of transmitted character	07
NAK	Negative AcKnowledgement of transmitted char	05
OER	Hardware time-out on output transmission	00
DRI	Data Request Immediate (printers) .... Bit A = 1.	00,04,08,0C
DRD	Data Request Delayed (printers) .... Bit A = 0.	00,04,08,0C
SER	Power-on character from selector unit	03
STD	Status input (e.g. printer voucher status) .. Bit A = 0.	00,04,08,0C
OBC	Output Block Control function	08,0A
ABC	Acknowledgement of a Block transmission incl. information about LRC and VRC control in the selector unit .... Note: For OBC and ABC the left byte is used for LRC.	0C,0E
DIN	Data from INput device.	
SYN	Output for SYNchronisation of modem	55

Figure 13.5 is a copy of figure 2.9, repeated here for reference.

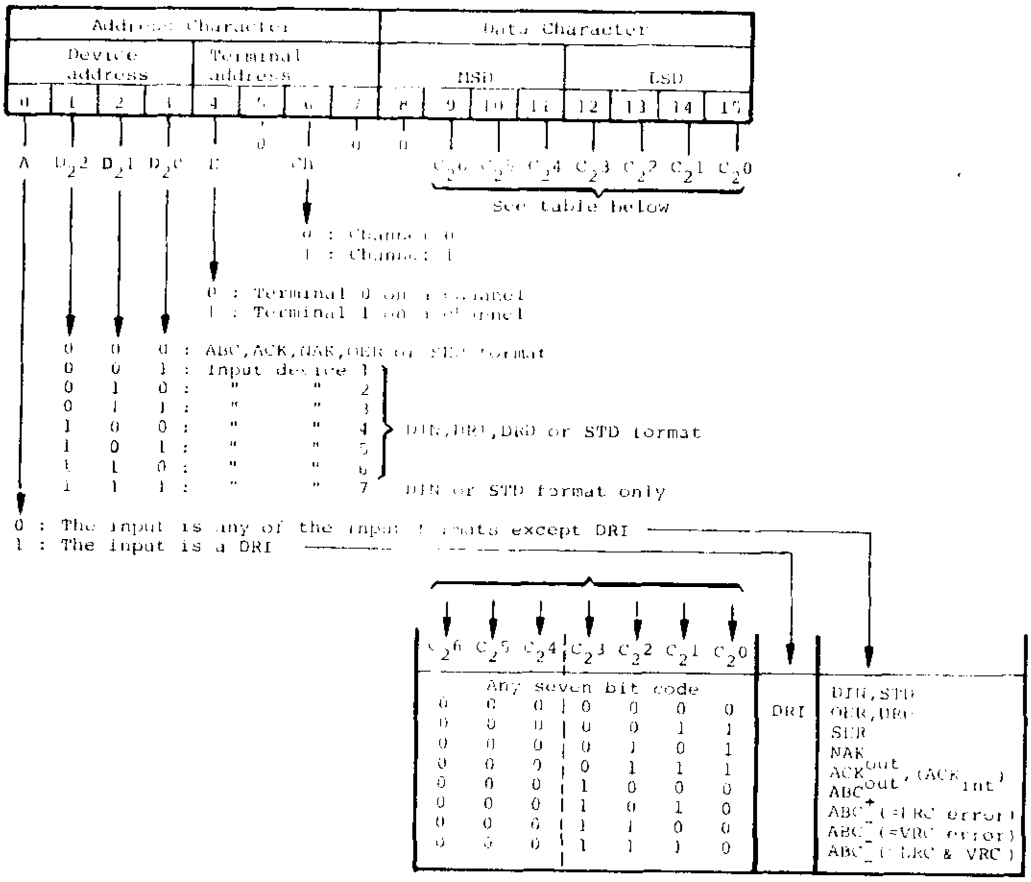

Figure 13.5. Layout of Input Messages from CHRT to CPU (copy of 2.9).

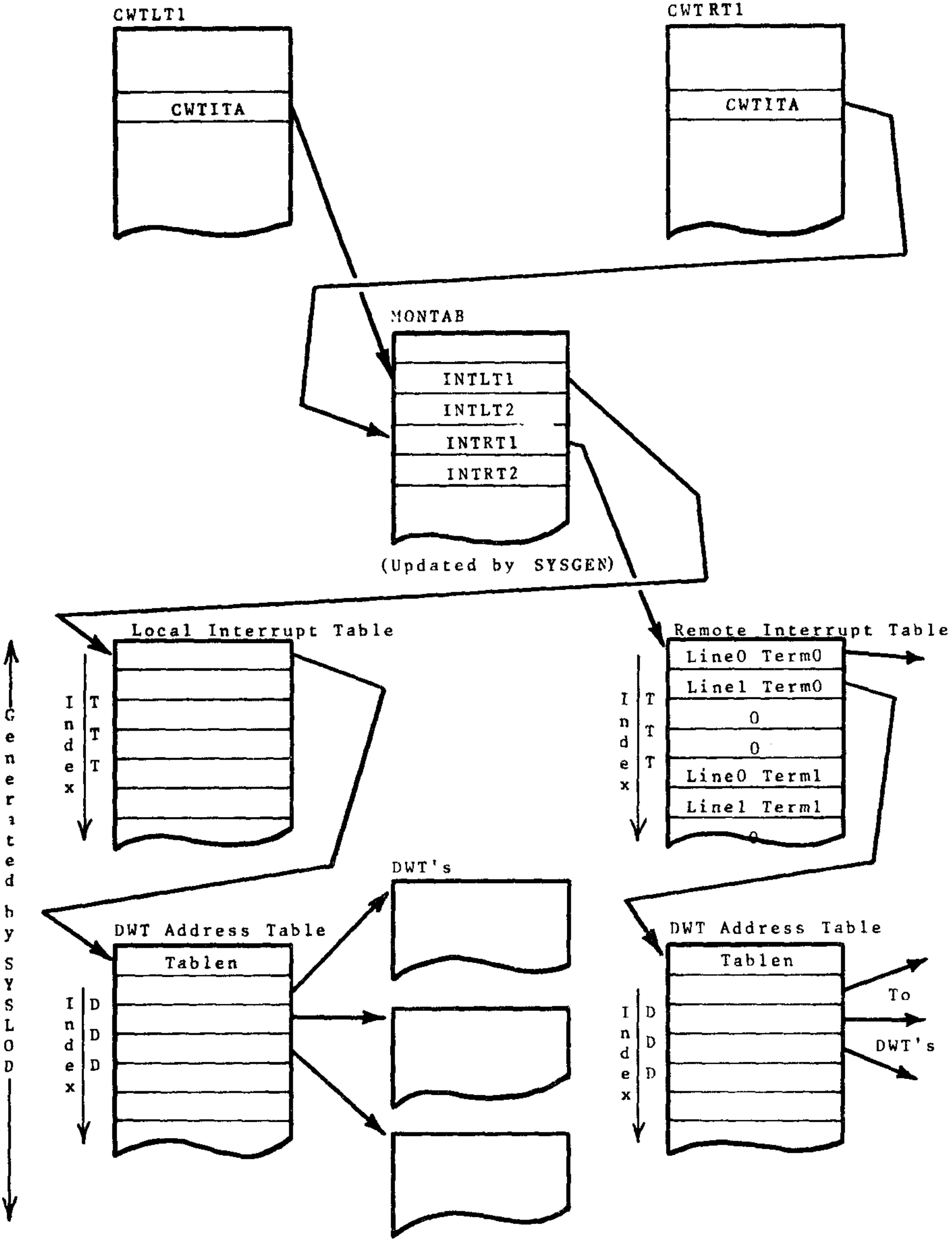

Figure 13.6. Channel Work Table and Associated Control Blocks.

13.3.2 Modules in DRRT01 used by Device Drivers

Below are described five routine calls used by device drivers. The routines are parts of DRRT01:-

1. CF A5,OUTPUT

All outputs to devices connected to a CHLT or CHRT are administered by DRRT01. This call is the normal way to send a character from device driver to device.

Register contents before calling:-

A2 = Character to send (bits 8-15).	A6 = DWT address.
A3 = Work Register.	A7 = Order.
A4 = Work Register.	A8 = ECB address.
A5 = DWT Stack pointer.	

Procedure:-

* Contents of registers A3 thru A5 are saved in the DWT.

* 'Interrupt Allowed' is indicated in the DWT (bit 7 in DWTST).

* When the channel unit is free, a character is sent to the device via the channel unit (word zero in the CWT is tested).

* If the channel unit is busy the output request is queued for transmission later.

* Control is given to the dispatcher while the channel unit is waiting for a return character.

* Possible entry points in the device driver after transmission are the interrupt handler or a recovery routine.

2. CF A15,OUTLIN

OUTLIN is a subsection of the routine OUTPUT described above. The difference between the calls is that OUTLIN does not save A3-A5 in DWT before transmission, and 'interrupts allowed' is not indicated. Simple drivers which do not use many registers use this call.

Register contents before calling:-

A2 = Character (bits 8-15).	A7 = Order.
A6 = DWT address.	A8 = ECB address.

Remotely connected devices select transmission in block mode if only one terminal is connected to a TFU. DRRTO1 performs this test. For devices with the data request function, e.g. printers, this implies that the block is sent character-by-character with LRC control of the whole block. Devices without the data request function use the CHRT block transmission feature with which LRC control of the block is performed.

The beginnings and ends of messages under LRC control are marked by the device driver sending the characters STX (/02) and ETB (/17). These characters must be complemented for DRRTO1 by the setting of the leftmost bit in the byte. There is no need for separate routines in the device driver for locally and remotely connected devices, as these two special characters are ignored in DRRTO1 for local devices.

3. CF A5,STREG

Input is possible only in single character transmission mode. A circular input buffer is used to store characters from an input device. At the time of a request, this buffer is collected first, and then the device driver has to prepare itself to get the next input character from the channel unit driver. This preparation is performed using the above call.

Register contents before calling:-

A3 = Work Register.	A6 = DWT address.
A4 = Work Register.	A7 = Order.
A5 = DWT Stack base.	A8 = ECB address.

Procedure:-

* Registers A3 to A5 are saved in the DWT.

* ´Interrupt allowed´ is indicated (bit 7 in DWTST).

* Control is given to the dispatcher while waiting for a channel unit interrupt.

* Possible entry points in device drivers after input transmission are the interrupt handler or a recovery routine.

4. CF A15,GETCHR

This subroutine is frequently used by drivers as a general printer, video display, and numeric display driver. It is used to fetch the next character from the ECB buffer and, when the end of the buffer is reached, to store the effective length in the ECB. Register A3 is used as an index and contains a displacement from the start of the buffer to the character to be fetched.

Register contents before calling:-

A3 = Buffer Index.	A8 = ECB address.

Register contents after return:-

A1 = Positive if not last character. Else zero or negative.	A3 = Updated Buffer Index.
A2 = Character (bits 8-15).	A8 = ECB address.

5. ABL LDREG

This call is used by almost all device drivers, especially in the interrupt handling part. After some tests the interrupt handler gives control back to the running part of the device driver by executing return from the call CF A5,OUTPUT or CF A5,STREG.

Procedure:-

* A test for 'interrupt allowed' is carried out, and if so the contents of A3, A4 and A5 saved in the DWT are restored. Bit 7 in the status word, DWTST, is reset, indicating interrupts not allowed. The Order is stored in A7 and the ECB address is stored in A8. A return is executed from the subroutine call CF A5,OUTPUT or CF A5,STREG.

13.3.3 Activation of Device Drivers from DRRT01

Every output from DRRT01 will cause an interrupt from the channel unit. Such interrupts, together with data inputs, are controlled by DRRT01, and the device driver is activated. Two entry points are possible, depending on the result of the transmission:-

* Interrupt Handler.
* Recovery Routine.

Examples of situations where the interrupt handler is activated:-

* An output has been executed and found to be correctly transmitted.

* An output has been executed and found to be incorrectly transmitted; a special return code is set.

* Data input, for example from a keyboard.

Register contents before the interrupt handler in the device driver is entered:-

A1 = Return Code (0 = operable, 1 = not operable).
A2 = Input word (for example, containing voucher and journal
     status or data character).
A6 = DWT address.
A7 = Order.
A8 = ECB-address.

Examples of situations where the recovery routine is activated:-

* Selector unit powered after a power break.

* Hardware timeout for a transmitted character.

* Bad LRC on transmitted block.

* Seven NAK's from the selector unit for the same character.

* Software timeout on data request supervision for printers.

* Software timeout on block check character supervision for a selector unit.

* Voucher status change before the whole message has been transmitted to a teller printer.

Register contents before the recovery routine in the device driver is entered:-

A4 = 0 if selector unit power-on,
     <u>or</u> hardware timeout for transmitted character,
     <u>or</u> seven NAK's from a selector unit.
A4 ≠ 0 if LRC error (or any other type of error).
A5 = DWT stack base.
A6 = DWT address.

13.3.4 Interface Program for CHLT and CHRT

Channel units are controlled by instructions from the following set:-

- * CIO Start ..... Start I/O.
- * OTR .... Output.
- * INR .... Input.
- * CIO Halt ...... Stop I/O.
- * SST .... Send Status.
- * TST .... Test Status.

DRRT01 uses only CIO Start, OTR, and INR in order to control the channel units for local and remote terminals.

1. CIO START.

Format:-

```
 ____________________________________________
| 0 | 1000 |  R3   | 0 | F |      DA       |
|___|__//__|__//___|___|___|______//_______|
  0  1    4 5     7  8   9  10           15

          R3 = Operand Register
                (not used)
```

DA = Device Address
/03 for CHLT1
/13 for CHLT2
/23 for CHLT3
/33 for CHLT4
/01 for CHRT1
/11 for CHRT2
/21 for CHRT3
/31 for CHRT4

Function and Condition Register contents:-

CIO Start is used to prepare the channel unit for transmission of characters; used at system start and after mains power failure.

Address not recognized:- CR = 3

Address recognized:-
command rejected - CR = 1
command accepted - CR = 0

2. OTR - Output one character.

Format:-

```
 _____________________________________________
| 0 | 1000 |  R3  | 0 | F |     DA     |
|___|__//__|__//__|___|___|____//______|
  0  1    4 5    7  8   9  10        15
```

DA = Device Address
R3 = Operand Register
F for remote channel:-
    0 = channel 0
    1 = channel 1
F not used for
    local channel.

Function and Condition Register contents:-

The OTR instruction is used to transfer data from a CPU register to a channel unit. It is accepted only if the channel unit is in exchange state.

Format of R3:-

```
 ________________________________________________
| A |  D   |  L   |  I  |       C        |
|___|__//__|__//__|_____|_____//_________|
  0  1    3 4    6  7   8  9            15
```

A = Procedure control bit (remote only).
    1 = Block transmission mode.
    0 = Single character transmission mode.

D = Device Address.
    0 = OBC message, clears LRC logic in SUMR (remote only).
    1 thru 6 = Output devices.
    7 = SYN format (remote).
    or Output device (local).

L = Line number.
    Local ..... Line numbers 0 thru 7.
    Remote .... /0 = Terminal 0, Channel 0.
                /2 = Terminal 0, Channel 1.
                /8 = Terminal 1, Channel 0.
                /A = Terminal 1, Channel 1.

I = Irrelevant.

C = Character (any seven-bit code).

Address not recognized:- CR = 3

Address recognized:-
    command rejected - CR = 1
    command accepted - CR = 0

3. INR - Input one character.

Format:-

```
 _______________________________________
| 0 | 1001 |  R3  | 0 | 0 |    DA    |
|___|__//__|__//__|___|___|___//_____|
  0  1    4 5    7  8   9  10      15
```

DA = CU Device Address.

R3 = Operand Register.

Function and Condition Register contents:-

The INR instruction is used to transfer data from a channel unit to a CPU register.

Format of R3:-

```
 ____________________________________________
| A |  D    |   L   |   I   |       C       |
|___|__//___|__//___|___|___|_____//________|
  0  1     3 4     6  7   8  9             15
```

A = Procedure control bit (remote only).
1 = Data Request Immediate (DRI) in bits 9 thru 15.
0 = Any other input format in bits 9 to 15. This depends also on the value in D (device address).

D = Device Address.
0 .... Following formats possible in bits 9 thru 15;
ABC - Block control character.
SER - Power-on.
OER - Hardware timeout control character.
ACK - Output acknowledged.
NAK - Output not acknowledged.

1 thru 6 ... Following formats possible in bits 9 thru 15;
DIN - Data input.
DRD - Data request delayed.
STD - Status input.
DRI - Data request immediate.

7 .... SYN input (remote).
or Device input (local).

L = Line number.
Local ..... Line numbers 0 thru 7.
Remote .... /0 = terminal 0, channel 0.
/2 = terminal 0, channel 1.
/8 = terminal 1, channel 0.
/A = terminal 1, channel 1.

I = Irrelevant.

C = Character code and meaning; depends on A and D fields.

/0 =	OER	Hardware timeout.
/2 =	DRI	Data request immediate (remote).
/3 =	SER	Power on.
/5 =	NAK	Output not acknowledged.
/7 =	ACK	Output acknowledged.
/8 =	ABC+	Acknowledgement of block transmission, no error.
/A =	ABC-	Acknowledgement of block transmission, bad LRC.
/C =	ABC-	Acknowledgement of block transmission, bad VRC.
/E =	ABC-	Ack. of block transmission, bad LRC and bad VRC.

Any seven-bit code.
DIN = Data input.
STD = Status input (remote).
DRD = Data request delayed.

Address not recognized:- CR = 3

Address recognized:-
command rejected - CR = 1
command accepted - CR = 0

Note One important rule is always adhered to in program/channel unit communication:-

* Local: Every output results in a reply from the channel unit to the program after transmission. This includes the case where a power-on character appears from a selector unit just after an output. The three replies possible are ACK, NAK, and OER.

* Remote: There are four possible replies in this case. They are ACK, NAK, OER, and DRI.

13.3.5 Functions of DRRT01

The most fundamental functions performed by the channel unit driver are listed below.

* Activation of channel units.
* Output and input administration.
* LRC control for output messages.
* Transmission of synchronisation characters.
* Time supervision of data requests from printers and block control characters after block messages.
* Remote line Test.
* Hardware status control by use of error accumulators.
* Logging of inputs and outputs for checking of both hardware and software actions.

The channel unit driver can handle up to four local channel units and four remote channel units at the same time. These channel units are placed on eight different interrupt levels.

For programming purposes, a Channel Work Table (CWT) is needed for each channel in the system. The CWT holds information concerning the status of transmission on the channel.

Channel Work Table for Local Terminals, CWTLTy (y = 1 thru 4):-

CWTLDW	Last output DWT
CWTLOW	Last output word
CWTITA	Address of interrupt table
CWTINR	INR instruction
CWTOTR	OTR instruction
CWTCIS	CIO Start
CWTRTC	Retransmission counter
CWTEQ	Queue first terminal on channel
CWTADD	NAK accumulator
......	Retransmission fault accumulator
Reserved	
ACKTIM	Timeout accumulator (3 seconds)

Channel Work Table for Remote Terminals, CWTRTy (y = 1 thru 4):-

CWTLDW	Last output DWT
CWTLOW	Last output word
CWTITA	Address of interrupt table
CWTINR	INR instruction
CWTOTR	OTR instruction
CWTCIS	CIO Start
CWTRTC	Retransmission counter
CWTEQ	Queue first terminal on channel
	Queue second terminal on channel
CWTADD	NAK accumulator
......	Block error accumulator
Reserved	
ACKTIM	Timeout accumulator (3 seconds)
CWTTP	Timer pointer
CWTSYN	SYN character
LRCDWT	DWT of running LRC process, terminal 1
	DWT of running LRC process, terminal 2
LRCACK	LRC accumulator terminal 1
	LRC accumulator terminal 2
CWTRST	Channel Status Word (loop test)
CWTBLK	Block sending indicator

CWT field descriptions:-

CWTLDW Last output DWT.
When an output is done to the transmit buffer in a CHLT or CHRT the channel unit is not then able to receive any more characters. The software indicates this by storing the DWT address in CWTLDW. The DWT concerned is for the device occupying the channel unit for output transmission. When the channel unit receives a reply for the transmitted character, its state switches from Execute to Exchange, and further outputs can then be done, indicated by clearing CWTLDW.

CWTLOW Last output word.
The last output word used when a selector unit replies NAK for output transmission. The last output word may be retransmitted up to seven times.

CWTITA Address of interrupt table pointer.
Channel units have one interrupt table each. This word points to a word in MONTAB (INTLTx for local, INTRTx for remote) which points to the interrupt table. MONTAB is created at SYSGEN. The interrupt table reflects the physical configuration of the devices connected to the channel. A word in this table points to the DWT address table, which consists of pointers to the DWT's of devices present at the workstation. The device causing the interrupt can be determined from the device address received. For terminal recovery, the table is used to find which devices are connected to the terminal. See figure 13.6.

CWTINR
CWTOTR } These words contain the INR, OTR, and CIO instructions used by the driver.
CWTCIS

CWTRTC Retransmission counter.
If the channel unit has problems when receiving a character, a NAK is sent, the character is retransmitted, and this counter is incremented by one.

CWTEQ Terminal queue.
Every local channel and every remote terminal has its own output queue. CWTEQ is a pointer to the first device in the queue. The second device pointer is found in the DWT of the first device in the queue, and so on (DWTTQ). The output queue is built when an output request is made and the word CWTLDW is not equal to zero. See figure 13.7.

CWTADD NAK accumulator.
This accumulator is incremented by one whenever a NAK is received from the SUMR.

...... Retransmission fault accumulator (local).
When a character is received from a keyboard and the channel cannot accept it, up to four retries are made. This word contains the retry count.

...... Block error accumulator (remote).
In block mode, whenever a negative ABC is received, one is added into this accumulator.

ACKTIM Timeout accumulator (3 seconds).
When, for example, a data request (DRI) is missing, i.e. not received within 3 seconds, this accumulator is incremented by one.

CWTTP Timer pointer (500 ms).
Contains a pointer to a timeout block containing a value of 500ms. Used for SYN timing.

CWTSYN SYN character (remote).
Code X'55' stored in this word is sent every 500 ms when there is no other transmission on the channel. The terminals connected to the line are addressed alternately. This is needed in order to obtain the line status, for example when loop-connecting the line.

LRCDWT Every remote terminal has a pointer to the DWT for sending a message under LRC control. Other devices on the terminal needing to send such messages must wait until this indicator is cleared, i.e. the current message is sent, and the reply from the hardware about the result of the block transmission has come back.

LRCACK This accumulator is used for the control of even parity failures in the transmission of blocks. It is cleared at the beginning of every block to be sent, and when the last character in the block is sent, the value of the accumulator is sent down the line so that the selector unit can compare it with a hardware accumulator. Each remote terminal has its own software accumulator.

CWTRST Channel Status Word.
Bits 9, 14, and 15 when set have the following meanings:-
Bit 9 - CHRT not active or hardware failure.
Bit 14 - No reply received on output from channel (no ACK).
Bit 15 - No SYN received. TFU switch is probably not correctly set for looptest.
If bits 14 and 15 are both set, the line is probably broken.

CWTBLK Block sending indicator.
Bits 1 and 15 have the following meanings when set:-
Bit 1 - Block sending is active.
Bit 15 - Block sending is allowed (one terminal on TFU).

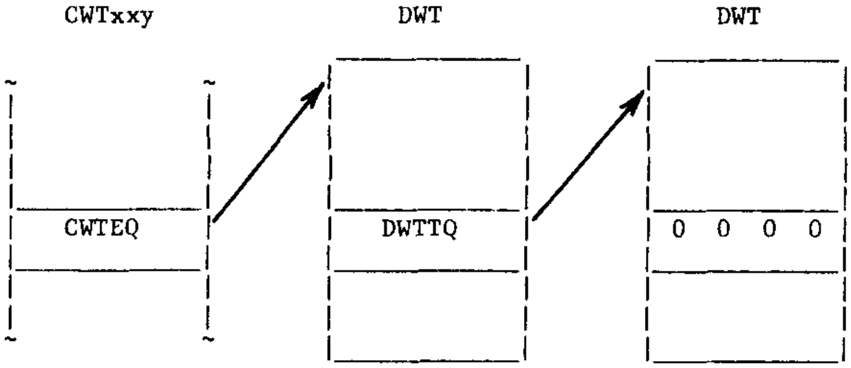

Figure 13.7 Terminal Queueing.

13.3.6 Starting of Channels

At system start local and remote channels are made active by CIO Start commands. The channels remain active all the time that the program is running. Mains power failure sets channels inactive, but the program activates them immediately on power-on.

Activation of channel units causes a power-on interrupt from each active selector unit in the system, which in turn causes device recovery for active terminals. For remote terminals, a timer is started to send SYN characters every 500 ms. If a channel is busy at power-on time, for instance if a character was being sent from the program just before the power break, a dummy is sent to the device to ensure that the device driver gets control of the situation.

13.3.7 Output Administration

All outputs are handled by the driver DRRT01. Every device driver orders output by preparing the device work table and calling a subroutine in the channel unit driver. If the channel unit is free, i.e. the result of the last character transmission has reached the program, the character is sent to the hardware.

However, if the channel unit is busy, the character must be queued. In the local situation, each channel has its own queue, and the characters are queued by the First-In-First-Out (FIFO) method. Every time the channel unit becomes free, i.e. a transmission control character is received from the channel unit as a reply to an output character, the next character in the queue is sent.

Each remote terminal has its own output queue, and each channel may have two terminals. When a channel becomes free, the queue belonging to the terminal not involved in the last transfer is checked. If the queue is not empty a character is transmitted, otherwise the other queue is checked.

For local printers every output character causes an ACK, NAK, or OER. If it is ACK, then Data Request Delayed (DRD) is sent when the selector unit is ready for the next character.

There is a 40-character receipt buffer in the selector unit for use by remote terminals. If this buffer is not full, Data Request Immediate (DRI) is sent by the selector unit as the reply to printer output. When the buffer is full, ACK is sent as reply, and when it is no longer full, DRD is sent to the channel unit. The program can check when the whole message has been printed by sending /03 and waiting for DRD after ACK, when the selector unit buffer is empty.

A special character, /01, can be sent to remotely connected printers when the program requires the printer to stop sending characters from the selector unit buffer.

13.3.8 Input Control

The following types of input interrupts are possible:-

* ACK - Output acknowledgement.
* NAK - Output not acknowledged.
* DRI - Data request immediate.
* OER - Hardware timeout on output character.
* DRD - Data request delayed.
* DIN - Data input.
* STD - Status input.
* ABC - Block control character.

ACK, NAK, or OER are possible replies from a local channel unit after an output. Remote units additionally use DRI.

DRRT01 operations caused by various interrupts are outlined below:-

* ACK The next character from the output queue is sent and a return is executed to the device driver.
  Note: If the ACK is from a printer, control is not passed to the device driver until DRD has been received.

* NAK The character is retransmitted up to seven times, then the recovery routine in the device driver gets control.

* DRI The next character from the output queue is sent and control is returned to the printer driver. (Remotely connected printers only).

* OER The first timeout causes device recovery. If two timeouts occur in succession, control is passed to the interrupt handler in the device driver with a 'not operable' return code.

* DRD The printer driver's interrupt handler is activated. For locally connected printers this data request character gives information about voucher and paper status. For a remotely connected printer the data request character is supplemented with voucher and paper status information in DRRT01.

* DIN The data input character is transmitted to the interrupt handler in the device driver.

* STD A status message is sent from printers when a change in voucher or paper status occurs. The new status is indicated in the DWT. Sometimes a change in voucher status requires termination of the current operation, for example if a voucher is taken away while printing. In this case the recovery routine in the printer driver is activated, and it sends a code /01 to clear the print buffer in the selector unit, stopping the printer immediately. (Remotely connected printers only).

* ABC There are four types of block control characters:-

  ABC+ ... Block transmission correct.
  ABC- ... Bad LRC.
  ABC- ... Bad VRC.
  ABC- ... Bad LRC and VRC.

  A check is made that block control characters are currently permitted. Then, if ABC+, control is given to the interrupt handler in the device driver. In the case of bad LRC or VRC, the recovery routine in the driver is activated.

13.3.9 Longitudinal Redundancy Check Control (LRC)

This feature is only implemented with remotely connected selector units. Each remote selector unit has an LRC accumulator for LRC control of messages sent to its devices.

Beginnings and ends of messages are determined by sending a special character, OBC, to device address zero. The first OBC clears the selector unit accumulator and the last one contains an accumulator set up by software to compare with the hardware accumulator.

Since only one LRC accumulator per selector unit is available, only one of the unit's devices at a time can send a message under LRC control. If one device is sending an LRC-controlled message and another device on the same selector unit also needs to send one, the second device must wait in a queue until the first device is ready with its message.

However, devices not needing LRC control of messages can mix their characters with characters from the device sending an LRC-controlled message. Two devices belonging to different selector units connected to the same TFU can send messages under LRC control at the same time.

The LRC feature may be used with both modes of transmission. Block mode is used particularly for video displays and character-by-character transmission under LRC control is used with printers.

Bad LRC in a printer message immediately causes activation of the recovery routine in the device driver, and its first action is to send a character /01 to clear the print buffer in the selector unit. Although the whole message has been sent to the selector unit when bad LRC is detected, the printer may not have printed all of it when it is stopped.

Block transmission is selected by setting a bit in the output register. The channel unit generates an internal ACK immediately for each character, and does not wait for a reply from the selector unit. This means that the device driver gets no information about the success of the transmission while it is in progress. It therefore has to send a character in single transmission mode after the block, in order to get TFU, selector unit, device, and line status.

13.3.10 SYN Transmission

A synchronisation (or SYN) character, code /55, is transmitted by DRRTO1 every 500 ms when no other input or output messages are in progress. Device address 7 on the selector unit is reserved for SYN transmission. The two selector units connected to a line are addressed alternately, so that ACK is received at least every 2nd time each selector unit is busy. This is important when the line is loop connected.

The SYN timer is started when the channel unit is activated at system start or after a power break. Each input or output restarts the timer, and if there is no activity on the channel for 500 ms, a SYN character is sent to the selector unit.

13.3.11 Time Supervision of Data Request and Block Control Characters

For output to devices with data request (printers), and devices sending messages in block transmission mode, there is a timeout facility to supervise data request characters and block check characters.

Printers:-

The first output in a printer message to a selector unit (OBC or some other character) starts the timer. For every ACK or DRI during output of the message, the timer is restarted. DRD after ETX, which always ends a printer message, resets the timer.

The last block output character (OBC) is sent before ETX to test for even parity. If timeout occurs before ETX is sent, it means that either a data request or a block check character is missing. Either case causes message recovery.

Time before timeout is set to 3 seconds. If a timeout occurs when a teller printer is waiting for grasp or release, ETX characters are sent every third second until the grasp or release is executed by the operator.

Block mode devices:-

ACK of the first block output character (OBC) starts the timer, and the character after the last OBC resets it. If timeout occurs, one of the two block characters (ABC) is missing, and recovery of the message is carried out.

13.3.12 TFU Loop Switch and Test of Remote Line

For line testing purposes, the loop switch on the TFU is set to Test mode. The characters sent down the line turn round in the TFU and return to the channel unit. The channel unit normally responds with ACK, which again turns round in the TFU and returns to the channel unit. If the line is faulty, the hardware times out, and an OER interrupt is sent to the program.

A special request, test remote line, is available. SYN transmission is used for this test. The request ends when two SYN characters are sent to different selector units connected to the same TFU.

Examples of remote line testing are shown on the following page.

```
1.   SYN       ACK       SYN       ACK
        out                 out
     ------>   <------   ------>   <-----

2.   SYN       ACK       SYN       OER
        out                 out
     ------>   <------   ------>   <------

3.   SYN       OER       SYN       ACK
        out                 out
     ------>   <------   ------>   <------
```

Conclusions:- The loop switch is probably not in Test mode.
2 and 3 indicate that only one selector unit is active.

```
4.   SYN       OER       SYN       OER
        out                 out
     ------>   <------   ------>   <------
```

Conclusions:- If the TFU is in test mode the line is probably broken. If the TFU is in normal mode it is possible that both selector units are inactive.

```
5.   SYN       SYN       ACK       SYN       SYN       ACK
        out       in                  out       in
     ------>   <------   <------   ------>   <------   <------

6.   SYN       SYN       ACK       SYN       ACK
        out       in                  out
     ------>   <------   <------   ------>   <------

7.   SYN       OER       SYN       SYN       ACK
        out                 out
     ------>   <------   ------>   <------   <------
```

Conclusion:- Line OK.

```
8.   SYN       SYN       OER       SYN       OER
        out       in                  out
     ------>   <------   <------   ------>   <------
```

Conclusion:- Line not OK.

The examples above show that the test remote line request must be repeated several times before any valid conclusions can be drawn.

When DRRT01 detects SYN-in as a reply to SYN-out, it switches status to test mode, i.e. all data inputs are blocked and all outputs to devices are queued. When the loop switch is set to the normal position from the test position, a power-on character is sent to the channel unit and the driver changes status from test to normal. Queues are resolved and inputs are allowed.

Only the traffic on the faulty channel is affected. One remote channel may be in test status while normal work is in progress on the other.

13.3.13 Error Accumulators

For test purposes, there are error accumulators in the control work table of each channel, updated while the channels are active.

These accumulators are:-

* Local - NAK.
  - Retransmission faults.
  - Undefined interrupts.
  - Software timeouts.

* Remote - NAK.
  - Block transmission errors.
  - Undefined interrupts.
  - Software timeouts.

Trouble in hardware configurations can often be diagnosed by analyzing the contents of these accumulators.

13.3.14 The Log Function

A log function is built into the channel unit driver for the testing of both device drivers and channel units. Every input and output is logged in a special circular log buffer.

13.4 PTS 8000 LOCAL & REMOTE TERMINAL DRIVERS

PTS 8000 systems use the V-24 interface for the connection of workstations. The cards used to implement this interface are called ASCU4Z and SALCUZ for local and remote connection respectively; the associated drivers are DRAS01 and DRSL01.

13.4.1 ASCU4Z & SALCUZ Configuration

A maximum of 6 ASCU4Z's and 4 SALCUZ's may be configured in an 8000 system; devices which can be connected are PTS 8046 (6346) displays, PTS 8071/72 (6271/72) keyboards, and PTS 8081 printers.

Each ASCU4Z supports 4 half-duplex lines; a SALCUZ supports 1 full-duplex line. This gives a maximum of 24 half-duplex lines for local connections, and 4 full-duplex for remotes.

Possible device combinations connecting to an ASCU4Z are as follows:-

* 2 displays with keyboards.
* 1 display with keyboard + 1 printer.
* 2 printers.
* 4 displays.
* 2 displays + 1 display with keyboard.
* 2 displays + 1 printer.

A SALCUZ can connect 1 display with keyboard, 1 display, or 1 printer.

13.4.2 Driver Switching in PTS 6800

By means of a new module (DRSW01) added to DRRT01 or DRLT01, the ASCU4Z and SALCUZ cards may be used in a PTS 6800 system. This module performs the necessary routing of I/O for the appropriate drivers; DRAS01, DRSL01, or DRRT01/LT01 (see figure 12.1, boxes 34, 35, 36, & 37).

If DRSW01 is used, then all calls from device drivers are made to it instead of to DRRT01/LT01. Five calls are possible:-

```
a.  CF  A5,OUTPUT
b.  CF  A15,OUTLIN
c.  CF  A4,STREG      A5
d.  CF  A15,GETCHR
e.  ABL LDREG
```

Execution of calls c, d, and e is done within DRSW01 in the same way as in DRRT01/LT01. The other two, a and b, involve the checking of channel parameters and are described below.

a. CF A5,OUTPUT

This call is used for normal output from the device driver. DRSW01 checks the character in bits 8-15 of register A2; if it is not /82 or /93 and ASCU4Z's or SALCUZ's are configured, then either DRAS01 or DRSL01 is accessed.

If the character is /82 or /93, then an immediate return is made via A5 and the interrupt routine of the calling device. These characters indicate block start and end for the old remote procedure, and have no meaning for ASCU4Z and SALCUZ.

b. CF A15,OUTLIN

This is a request for output of a character with no special handling, and without returning through the A5 stack. DRAS01 or DRSL01 is accessed immediately in appropriate configurations.

13.4.3 Device Connections to ASCU4Z & SALCUZ

1. Display with Keyboard.

   ASCU4Z - Since this device requires one line for input and one for output, only two may be connected to an ASCU4Z, each occupying a pair of lines. Physically there is only one connection for each device, and this must be to the higher number line in the pair, ie. 2 or 4.
   The logical configuration numbers do not have to be the same as the physical connections; for example, two displays with keyboards could be configured as 2 and 3, leaving 1 and 4 unused.

   SALCUZ - Only one input and one output line are available, so that only a single display with keyboard may be connected.

2. Printer.

   ASCU4Z & SALCUZ - This is connected in the same way as a Display with Keyboard.

3. Display.

   ASCU4Z - Displays can be connected to any of the lines, making a maximum of four devices possible.

   SALCUZ - Only one display can be connected; this is because only a single output line is available.

The first word in the Device Work Table, DWTCHP, contains the channel parameters, as follows:-

Bit 0 - Device without data request when set.

Bits 1 to 3 - Device Address.
1 = Keyboard.
2 = Display Indicator.
3 = Display / Printer.

Bits 4 to 6 - Line Number.
0 = line 1, ASCU4Z 1, 3, or 5 / SALCUZ 1.
1 = line 2, ASCU4Z 1, 3, or 5 / SALCUZ 2.
2 = line 3, ASCU4Z 1, 3, or 5 / SALCUZ 3.
3 = line 4, ASCU4Z 1, 3, or 5 / SALCUZ 4.
4 = line 1, ASCU4Z 2, 4, or 6.
5 = line 2, ASCU4Z 2, 4, or 6.
6 = line 3, ASCU4Z 2, 4, or 6.
7 = line 4, ASCU4Z 2, 4, or 6.

Bit 7 - Input Device if set.

Bits 8 to 11 - Not Used.

Bits 12 to 15 - Channel Index.
0 - /B Reserved for CHLT/CHRT.
/C ASCU4Z 1 or 2.
/D ASCU4Z 3 or 4.
/E ASCU4Z 5 or 6.
/F SALCUZ 1 to 4.

13.4.4 DRAS01 & DRSL01 Channel Work Tables

Each line has a Channel Work Table (CWT) associated with it, laid out as follows:-

CWT for DRAS01

CWTLDW	Device using the line
CWTLON	Last / next output characters
CWTINR	INR instruction
CWTOTR	OTR instruction
CWTCIS	CIO Start instruction
CWTEQ	Queuing DWT
CWTCIH	CIO Halt instruction
CWTSST	SST instruction
CWTTST	Channel status

CWT for DRSL01

CWTLDW	Device using the line
CWTINR	INR instruction
CWTOTR	OTR instruction
CWTCIS	CIO Start output instruction
CWTTP	Timer pointer
CWTEQ	Queuing DWT
CWTCIH	CIO Halt output instruction
CWTSST	SST output instruction
CWTTST	Channel status
CWTCSI	CIO Start input instruction
CWTSSI	SST input instruction
CWTTP2	Timer pointer 2

CWTLDW Pointer to the DWT of the device using the line.
Set to zero if the line is free.

CWTLON Left byte is the last output character (CWTLOC); right byte is the next (CWTNOC). CWTNOC is zero if no character is waiting.

CWTEQ Pointer to the DWT of a device which has requested use of the line when it is already occupied.

CWTST Status bits used by the driver.

13.4.5 Output to ASCU4Z

When output is required CWTLDW is checked to see if the line is free. If the line is free it is opened with CIO Start and the character is stored in CWTNOC. Control is then given to the dispatcher, an interrupt is generated, and the character is transferred to the device.

Control passes to the device driver interrupt handler with A1 set to zero, and the handler returns via ABL LDREG and sends another character which is saved in CWTNOC. The dispatcher regains control and when another interrupt occurs CWTNOC is checked to see if another character is waiting for transfer.

If this is the case CWTLOC and CWTNOC are updated and control again passes to the interrupt handler of the device driver. If no character is waiting, or if it is /03, the transfer is terminated by CIO Halt. Only a part of the request will be transferred at a time if all the characters do not reach the communication driver before an interrupt occurs.

After the transfer is finished another interrupt is generated and this must be followed by an SST instruction. Before control is given back to the dispatcher CWTNOC and CWTEQ are checked to see if a new character is waiting for output to the same device, or if another device is queued waiting for the line. This can only happen if the display driver is waiting during indicator output, or vice-versa.
In this case output must be started again with CIO Start. CWTLDW, CWTLOC, CWTNOC, and CWTEQ are updated.

13.4.6 Output to SALCUZ

CWTLDW is checked when output is required to see if the line is free. If it is, a character is transferred to the device using CIO Start, OTR, and CIO Halt, and control passes to the dispatcher. When the transfer is terminated an interrupt is generated and SST is executed, giving line status information.

If bad status is detected (not operable), control is given to the device driver interrupt handler with A1 set to 1. DRSL01 then continuously scans the line and performs recovery when possible.

For good status, control is passed to the interrupt handler with A1 set to zero. Before control is passed a check is made to see if another device is queued for the line, in which case the request is serviced.

13.4.7 Display Indicator Output

Special actions are taken by ASCU4Z and SALCUZ when output to a display indicator is requested. The keyboard where the indicator is located is directly connected to the screen; A character /1C is sent before the requested output and this causes the output to be routed to the keyboard instead of the screen. This is accomplished internally and makes no difference in programming terms.

13.4.8 Input from ASCU4Z & SALCUZ

Interrupts are generated for input characters which are looked after by the interrupt handlers of the individual device drivers. Neither parity nor any other kind of controls are implemented.

13.4.9 Recovery

After program loading each line configured for output is opened and the recovery routines for the devices are executed. Lines configured for input are opened for input.

For SALCUZ recovery is also performed when a line has become operable after being non-operable. DRSL01 transmits /00 twice per second as long as the line is not occupied with normal output.

If a carrier failure is detected on the input line, or it is not operable, an attempt is made to reopen the line every second until it becomes operable again. The device recovery routines are then executed.

Note: Since there is no signal indicating power-up from the terminals, the states of the terminals are lost during a power-off sequence at the terminal end; the keyboard lamps and key-lock positions are not updated at power-on.