

10. DATA COMMUNICATION

10.1 Introduction

The subject of data communication is a complex one, with different protocol systems for each kind of mainframe. For a programmer to arrange all the correct sequences of control instructions to go down the line at the right time will obviously be a complex task. This is handled in CREDIT by having a number of different drivers to handle the differing protocols.

There are two instructions used with data communication :-

READ	read data from the line
WRITE	send data down the line

The DC action is treated in exactly the same manner as any I/O operation. A dataset must be defined for the line and given a dataset name for use with the instructions. The action on the line is different between point-to-point and multipoint configurations, as shown below.

10.2 Time out

Because of the way in which DC applications work, it is necessary to incorporate a 'timeout' function. This means that a time limit is set for completion of the read or write request. If no message has been received on a read after the timeout expires, the request is completed, and an indication is given to the task of this occurrence.

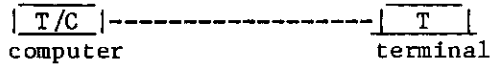
It is important that the terminal task and mainframe task should have different timeout values, to prevent a BID collision. This occurs when both the terminal and the mainframe are trying to use the line at the same time. If both requests time out and then try the request again, the same collision will occur. This can be avoided if the mainframe task has a shorter timeout value than the terminal task, thereby getting priority on the line.

To set the time out, which may be altered before each read or write, a data set control instruction is used:-

```
DSC1      DSDC,X'0B',TIME
```

10.3 Point-to-point

This is the simplest form of data communication with the computer connected to the terminal by means of a cable, see below:-



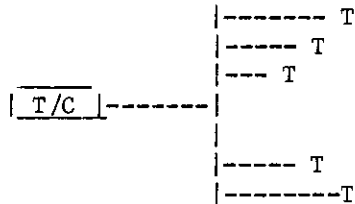
The two instructions described before are adequate for this situation. The terminal should be polled regularly in case it wants to send a message. This can be performed by issuing a read instruction at regular intervals. If an answer is required, the write instruction is performed to send data back to the terminal.

If the task is performing on a strict question and answer basis, it must be remembered that the task may still be switched between reading and writing, since this involves an LKM request.

If the terminal is connected to the computer via a telephone or telegraph line, then it may be necessary for an operator to dial the number and switch the modem or accoustic coupler on, when the carrier tone is heard.

10.4 Multipoint

This situation is more complex than point-to-point as a number of terminals could be connected to the line. this is shown in the diagram below:-



The line can be connected either permanently, or switched if all terminals are in the same area, so the connection is the same as described above. All messages must be addressed to the required terminal by transferring the terminal address to the DC driver. The same must be performed for a read instruction from a particular terminal, or by the terminal task when addressing a message to the computer.

CREDIT PROGRAMMERS GUIDE

10.5 DC task

With some applications it is possible for a terminal computer to receive data from the mainframe without it being requested. It is advisable in this case to write a task dedicated to the receiving of unsolicited messages. This task would have a READ outstanding on the line with no time out set. It would be entirely up to the application how the message was handed over to the processing task but this could be done using intertask communication.

