7500
(7501/02)

Restricted publication

Independent Data Entry

Restricted Publication 3023

Edition One August 1980

International Computers Limited
Head Office: ICL House, Putney, London SW15 1SW

This publication describes the ICL Independent Data Entry System (IDE), for use with the ICL 7501 and 7502 terminal systems.

The information covers the facilities of IDE, its installation and operation. The publication is intended for managers and analysts designing their data entry systems, for terminal operators and supervisors using IDE and for mainframe operators involved with IDE.

Further information on the 7501 and 7502 which may be required by the terminal supervisors can be obtained from the following ICL 7500 Range publications:

7500 System	(Edition 1, TP4802)
7502 Operating	(Edition 1, TP4804)
Operating 7500 Range Video Terminals	(Edition 1, TP4803)

Note: The information relating to the 7501 and 7502 terminal systems has been collated into two new publications:

7501/7502 Hardware Operating	(Edition 1, TP4818)
7501/7502 Software Operating	(Edition 1, TP4817)

Information of interest to analysts may be obtained from:

TPL Application Programming	(Edition 1, TP4809)

Additional information that may be required by the mainframe opertors can be found in the following ICL 2903 Range publications:

Remote Job Entry to the 2903 Range	(Edition 1, TP4610)
Stand-alone Facilities on the 2903 Range	(Edition 1, TP4611)

Chapters 1 and 2 of this publication introduce IDE and its facilities and will be of general interest. Chapter 3 describes how to design data entry screens and how to associate the IDE validation with these screens. This chapter is intended for designers and analysts using the system. Chapter 4 describes how to use IDE for data entry and should be read by the IDE operators. Chapter 5 describes the supervisory functions provided. Chapter 6 describes the transmission of data to the mainframe and the operation of the mainframe program that writes this data to the specified files. The Appendices cover general 7501 and 7502 operating, the installation of IDE and IDE error messages. Chapters 5 and 6 and the Appendices will be mainly of interest to the IDE supervisors.

The text of this publication is divided into chapters in the normal way, and each chapter is subdivided into sections. A section's level in the hierarchy is indicated by its number. Therefore, within Chapter *n*, first level section headings are numbered *n*.1, *n*.2 and so on; second level headings are numbered *n*.1.1, *n*.1.2 ... *n*.2.1 and so on; third level headings are numbered *n*.1.1.1, *n*.1.1.2 ... *n*.1.2.1 and so on.

The contents list and index, and cross-references in the text, all refer to section numbers.

Pages are numbered within chapters, in the form *c*-*p*, where *c* is the chapter number and *p* the page number within that chapter. Figures and tables, where they appear, are also numbered within chapters, so that Figure *n*.2 is the second figure in Chapter *n*, and Table *n*.2 is the second table in that chapter.

Section numbers, page numbers and figure and table numbers in appendices are preceded by the letter A.

Independent Data Entry (IDE) is data entry software for use with ICL 7501 and 7502 terminal systems linked to the ICL 2903 range of computers and the 2900 range with DME/2 software. IDE is software designed to input, validate, amend and verify batches of data off-line.

The data is validated as it is keyed in through a video and is stored automatically on diskettes at the terminal from which it can be retrieved for verification and editing. Only valid data, verified to user defined criteria, is transmitted to the mainframe for processing by user programs.

The system supports a range of data entry methods form the emulation of simple card punching methods to sophisticated data validation and verification. It is of particular use where it is required to input large quantities of data at locations remote from or local to the mainframe without involving the mainframe in the data entry task. Data is collected and errors detected and corrected by the terminal operating off-line from the mainframe. The terminal is only connected to the mainframe to transmit the verified data. Not only does the installation suffer fewer delays in processing because the data is correct but also data capture can be situated at the most convenient location for the user and the resources of the mainframe are not used for the data capture task.

IDE uses the standard ICL *terminal executives* (TEs) and the *7501/7502 Application Programming Development System*. The TEs and the Development System are described in the relevant chapters of this publication.

The minimum configuration required to operate IDE is an ICL 7501 or 7502 terminal system with 40 Kbytes of store (K being 1024 bytes), a dual floppy disc drive (half megabyte) and one video fitted with a personal identification device reader. Up to two videos can be attached to each 7501 and up to four videos, to each 7502. The use of two dual floppy disc drives is recommended where a more flexible data capture system is required.

This chapter summarises the facilities provided by the IDE package and some of the concepts involved in its use.

2.1 The data

In general the data to be input will be written on coding sheets or specially designed forms. A convenient quantity of data, termed a batch, will be input at one time and the batch will be handled as a unit while it remains in the system.

2.2 IDE facilities

The following list outlines the main facilities provided by the IDE package:

1 It provides data capture through videos of user records up to 120 characters long. Each record is held in card image form, so records over 80 characters are transmitted as two card images

2 It provides a simple method of designing screen layouts to user's specifications

3 It provides a comprehensive range of validation on input fields

4 It allows the user to control the selection of data from the input screen for transmission

5 It holds input data in batches and provides automatic batch totalling, two totals per batch

6 It provides facilities to chain together a number of input screens

7 It allows amendment and verification of batches of input data

8 It automatically collects some statistics on operator performance

9 It provides several supervisor functions to control the data input workload

10 It allows batches of data to be copied either to specified user files or to the mainframe spoolfile

2.3 Supervisor functions

The team of data entry operators is usually headed by a supervisor who identifies herself to the system by calling the IDE supervisor functions through a video and keying in a password. This can be done at any video. She can then use that video to initiate the supervisor functions. These functions include starting a data entry session, specifying the mainframe files to receive transmitted batches, displaying lists of the batches in the system and their current status, erasing unwanted batches and listing statistics of operator performance.

2.4 Security

Security features are a part of IDE design. Batches cannot be released for transmission until they are fully validated and verified and the transmission can only be initiated by the supervisor. Data held on diskette is unlikely to be erased by mistake since the command to erase batches can only be issued by the supervisor.

The facility exists to retransmit batches if data is lost or corrupted at the mainframe.

As the terminal operates off-line for data entry, that is, not connected to the mainframe, if the communication link to the mainframe is out of action, data entry can continue and verified batches can be kept until a link can be established.

Note: On 7501 or 7502 systems with a single dual floppy disc drive, data entry can continue only until the file containing verified batches is full. This file must then be emptied by transmitting the contents to the mainframe. On terminal

systems with two dual floppy disc drives, where one complete diskette is used for this transmit file, data entry can continue after the initial transmit file is full by replacing this with another empty transmit file. Appendix 2 describes how to set up transmit files of the required size and Chapter 6 describes transmission and emptying of transmit files.

The terminals also provide a security device on the video keyboards. This is a reader designed to accept a *personal identification device* (PID) in the form of a pen coded with an 8-bit binary pattern. In order to input screen formats and validation criteria to IDE a special PID numbered 254 is required.

In addition the supervisor uses the 7501/7502 utilities T2U*x*I to take security copies of the data on the diskettes. Security copying is described in section 6.7.

2.5 The IDE software

The IDE system is based on the 7501/7502 Application Programming system, a description of which can be found in *TPL Application Programming*.

The IDE system uses *screens* displayed on a video. A screen will typically consist of *headings* which will be constant and which the operator will not change and *data-fields* into which the operator will input data.

The system provides a number of routines coded in the *Terminal Programming Language* (TPL) which are used to validate and process the information which is input into the data-fields. These routines are known as *global action routines* and they are combined as appropriate into *user action routines* which provide the particular validation required for individual data-fields.

A group of screen formats, not necessarily related, with their associated data-field definitions and global action routines is termed a *path*.

The IDE system consists of four paths:

1 THE BASIC DATA ENTRY PATH This contains a set of screens and action routines which provide the basic data entry facilities such as batch totalling and operator statistics. The data entry screen formats with their associated validation built up from the IDE global action routines are added to this path

2 THE GLOBAL PATH This path contains the global action routines which perform the processing required for each of the data entry screen formats. This path also contains the IDE system software routines

3 THE SUPERVISOR PATH This contains a set of screens and routines accessible only by the supervisor and provides various control facilities not available to the other operators

4 THE SET-UP PATH This path contains the single screen required to define the files being used. It is normally used only when IDE is installed initially

These four IDE paths are held in a diskette file termed the *path library file* from which they can be accessed by the data preparation operators.

In addition to the IDE software described above the installation requires the Application Programming Development System program, T2AD01. This is used to add the data entry screen formats and validation to the basic IDE system.

The following terminal executives are also required:

1 T2AP11 This is the terminal executive used to control the program T2AD01

2 T2AP31 This is the terminal executive used when running the IDE system off-line for data capture and verification

3 T2RB31 This is the terminal executive used when transmitting data to the mainframe

4 T2U*x*I This is a series of utlities used to initialise diskettes, take security copies and allocate files

Only one terminal executive can be loaded at any one time. Their use is explained in Chapters 3, 5 and 6 and in the Appendices.

When receiving data the mainframe is under the control of program #XJXV. The mainframe utility #XIDE is provided to place the batches of data into the mainframe spoolfile or users' files as dictated before the batches are transmitted, (see section 5.2.8). The mainframe programs are described in Chapter 6.

2.6 The IDE operational sequence

When the IDE system is being installed the sequence consists of:

1 Establishing the Basic Data Entry path

2 Creating the files required by the IDE system

3 Using the IDE Set Up path

During the data entry session the operational sequence is:

1 Start of session control functions

2 Data input and validation

3 Modification

4 Verification

5 Transmission

2.6.1 Establishing the Basic Data Entry path

Before the IDE system can be used the Basic Data Entry path must be defined. This is done by modifying the Basic Data Entry path supplied as part of the IDE software. To modify this path details of the data entry screen formats are supplied to the system. The appropriate global action routines are also combined to form the user action routines which perform the processing required for each of the screen formats.

To modify the Basic Data Entry path the Application Programming Development System is used. For details of this see Chapter 3.

2.6.2 Creating the files required by IDE

IDE uses a number of files on the terminal and on the mainframe. They are:

1 ON THE TERMINAL:

(a) ENTRY FILES (ENT*n*) One entry file is created for each video on the system; *n* indicates the video number with which the entry file is associated. The entry files contain the batches that have been entered by the video

(b) TRANSMIT FILE(S) (TRANS*nn*) These files hold the verified batches to be transmitted to the mainframe. They also hold the operator statistics records when these are ready for transmission. Normally only one transmit file is created but up to nine transmit files can be used for the data from one data entry session. The numbers *nn* indicate the number of the transmit file, from 01 to 09

(c) COPY CONTROL FILE (IDECNTRL) This contains the copy information which tells the program #XIDE to which mainframe file each batch must be written

(d) SUPERVISOR CONTROL FILE (SPACNTRL) This contains the current supervisor password

(e) OPERATOR STATISTICS FILE (OPERATOR) This file contains a record for every operator identified to the system and every video in the system. When data is entered, modified or verified the system accumulates the number of correct keystrokes for each operator for each mode of entry. Should the operator not be identified the count is accumulated for the appropriate video

(f) PATH LIBRARY FILE (IDEPATHS) This is the file which holds the IDE software including the Basic Data Entry path as modified to include the data entry formats

(g) BATCH DETAILS FILE (BATCHDET) This contains one record for each batch known to the system. It holds details including the status of the batch. Batches that have been entered have a status D; batches that have been verified have a status V; batches that are in the process of being verified have a status I; batches that have been transmitted or are ready for transmission, having had the details of the files to which they are to be written supplied, have a status C. Batches that are in the process of being entered do not appear in this file

2 ON THE MAINFRAME:

(a) BATCH STATISTICS FILE (IDEBATCHSTAT) This file collects statistics for all the batches successfully received and processed at the mainframe. For each batch named the file holds the data entry session date and the number of correct keystrokes typed at each of the entry, modify and verify stages of input

(b) OPERATOR STATISTICS FILE (IDEOPERSTATS) Every time the operator statistics collected by the terminal are transmitted to the mainframe a new record is appended to this file for each operator and each video. The records contain the date the statistics were transmitted and the correct keystroke counts collected in the terminal file OPERATOR

(c) WORK FILE (IDEDWORKFILE) This is used by the mainframe program $XIDE

These terminal and mainframe files must all be created when the IDE software is installed. Appendix 2 describes the method of creating the files and their recommended sizes.

2.6.3 Using the Set Up path

Having created the terminal files described in section 2.6.2 the IDE system must be informed of the sizes of these files. This is done by using the Set Up path. In addition this path is used to specify the first supervisor password.

2.6.4 Start of session control functions

These are carried out by the supervisor and include checking that the transmit file is empty and ready to receive data. For details see Chapter 5.

2.6.5 Data input and validation

Data is collected and validated by the IDE system without the terminal being connected to the mainframe; that is, off-line.

Each batch of data to be input is given a name by the operator at the beginning of the input sequence. A password can also be associated with the batch at this stage.

When starting a batch the operator must input the number of documents (screens) in the batch and the number of records generated by each document. This allows the system to calculate the amount of space to reserve for this batch. The operator also has the option of specifying up to two batch control totals. Both the batch controls and the number of documents are used for consistency checking when input of the batch is complete.

The operator selects the screen format required for the data by inputting a two digit number associated with that format. This screen format will then remain in force until changed by the operator unless the format is linked to others in which case the new formats will be displayed automatically.

Validation checks are carried out at the end of each field after the operator has pressed the TAB key.

When a validation error occurs the operator should retype the field in error. If this is not possible she has the choice of either moving on to the next field to continue data entry or aborting the whole screen. The former action results in a record being input with that field in error. However the opertor can amend the field at a later stage or can change it on verification. Section 4.3.2 explains error correction in more detail and sections 4.4 and 4.5 deal with amending and verifying records.

When the last document has been entered, the operator closes the batch. The control totals specified when the batch was opened, if any, are now compared with similar totals accumulated by the system. If there is any discrepancey between the control totals and the actual totals, a message is displayed on the screen. The operator has then the option of forcing the system to accept the batch.

The system will warn the operator when the space allocated to this batch is nearly full. If there remain more record than will fit into this allocated space the operator can start a new batch. When this occurs it is likely that batch totals will be incorrect but the batch total error warnings can be ignored by the operator (see section 4.3.3). The batches can be merged when they are transmitted to the mainframe.

As batches of data are entered they are written to entry files held on diskette. Each video has its own entry file.

2.6.6 Modification

In modify mode the system allows the operator to open any batch that has not yet been verified. Data can then be amended or deleted from the batch.

The data in the batch is presented to the operator serially in the order in which it was placed on the file. The operator can skip over unwanted data until the required data is reached.

2.6.7 Verification

The process of validation checks that the data has been put in the right place and is of the right description (numeric or character) for the field. However it does not ensure that the correct values have been keyed in. The process of checking the precise contents of data-fields is termed verification.

When verifying a batch the operator keys in all of the data, or selected parts of it, for a second time and the terminal compares this new version with the original. Should a discrepancy occur, the operator has the option of retaining the original version or replacing it. The new version is validated when the TAB key is pressed.

After verification the batch is automatically transferred from the entry file to the transmit file from which it is transmitted. Batches on the transmit file cannot be accessed by the IDE system for further modification.

2.6.8 Transmission

Periodically the supervisor initiates the transmission of all the verified batches to the mainframe. Here they are processed by the utility #XIDE. This program places the data into either 1 block bucket user files or the mainframe spoolfile. The file into which the batch is to be copied must be specified for each batch before transmission. The order in which the batches are written to these files is determined by the order in which this COPY information is provided at the terminal.

While the terminal is transmitting data to the mainframe it cannot be used for data entry.

2.7 Analysing the statistics produced

Batch statistics and operator statistics will be written to the two mainframe files described in section 2.6.2. Data will be appended to these files by the program #XIDE.

Standard software is not provided to analyse and clear these files. It is up to each installation to develop its own programs and to run these as and when required.

This chapter describes how to modify the Basic Data Entry path to include formats for data entry and validation. It gives detailed instructions on how to set up screen formats and how to define the data-fields in those formats. It also describes how to specify the action routines required to use the IDE facilities.

One screen format is set up to represent each screen of input data. An application may require a number of different screens and therefore a number of screen formats must be defined to the system. A format consists of a screen layout, definitions of the data-fields, user action routines to specify the validation required and standard action routines to initiate IDE system functions.

The first task to be undertaken is defining the screen formats. This entails designing the layout of headings and data-fields on the screen. The next task is to write the user action routines to provide the data validation and to include the system routines required. The final task is to modify the Basic Data Entry path supplied as part of the IDE software to include these screen formats and associated validation.

3.1 Defining screen formats

3.1.1 The format

The format of a screen of input data is specified on a *screen layout form*. Screen formats will differ between applications and can be specified to meet the particular requirements of each application.

In general screen formats should be designed to follow the layout of the input documents as closely as possible. An example fo this is shown in Figure 3.1.

All headings and narrative required on the screen should be specified on the layout form. During data entry these will be *protected* fields on the screen; that is, the operator will not be able to type over these headings and they will not appear on the record output to the diskette.

Every *input field* should be positioned in the appropriate place on the layout form and enclosed by field delimiters ⌊ ⌋. These indicate that the field is *unprotected* and therefore the operator will be able to key data into this field. This data will appear on the record output to diskette, unless specifically excluded.

Within a screen of input data there may be a requirement for the contents of certain data-fields to be predefined. This facility is provided by the use of *constants*. Constants and input fields together form the *data-fields*. If constants are to be used the contents should be specified explicitly on the layout form.

Constants are not enclosed by field delimiters but they are unprotected.

Note: If the same constant appears twice or more on the screen it need only be explicitly declared once. The data-field definition for each occurrence of the constant then refers to the position on the screen where the constant is declared (see Figures 3.2 and 3.4 for examples of repeated constants). This facility can be used to save space or provide a more easily readable screen format.

Any unprotected field can be blanked so that data input to this field is not displayed on the screen. It is usual to blank constants so that they are not displayed. Section 3.2.2.1 describes this facility.

When designing screen formats for use with IDE the following restrictions apply:

1 Only 79 unprotected input fields can be specified on any one screen layout. This number is reduced by one for every constant in the layout

2 The last line of the screen should be left blank. This will be used by the system to display error messages

3 The last 12 characters in the next to last line of the screen should be left blank for use by the IDE system

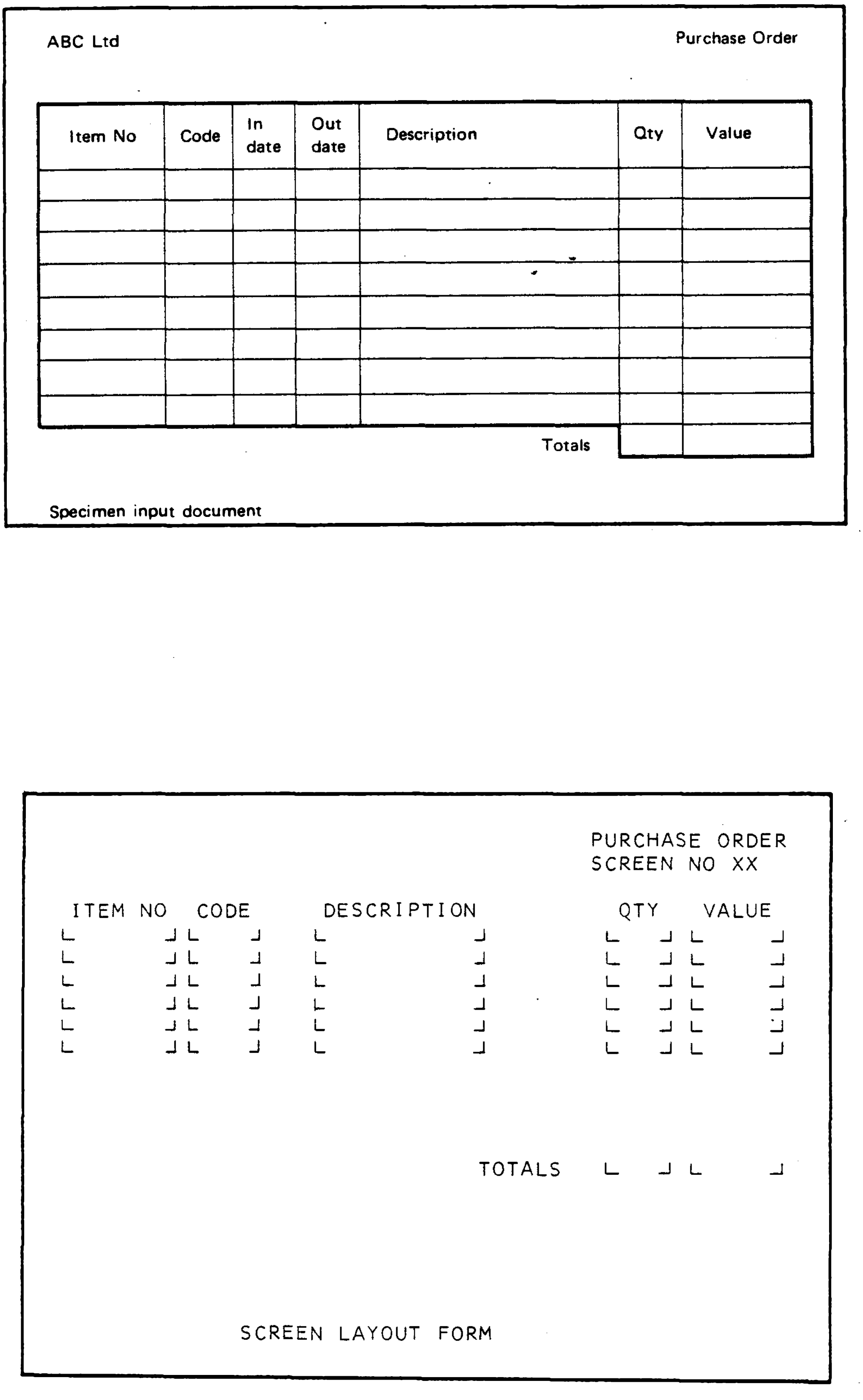


Figure 3.1 Screen format designed to follow the layout of the input document

3.1.2 Data-field numbers

Every unprotected field on the screen, including constants, must be given a *data-field number* (DF*n*) where *n* is in the range 01 to 79.

The data-field number specifies the sequence in which the data-fields are written to the diskette to form the data entry record. The data-fields need not appear on the screen in data-field number sequence although they will be written to the data entry record in sequence. Regardless of the order in which the data fields have been numbered, they will appear in the position specified by the screen format when they are displayed on the screen. An example of the allocation of data-field numbers is shown in Figure 3.2.

Even if a repeated constant has only been defined once a data-field number must be allocated for each occurrence of the constant in the position in which it is to appear on the record. Figure 3.2 shows an example of this.

The last 12 characters in the next to last line of the screen format, that have been left blank for use by the IDE system., must be given the data-field number 80.

3.1.3 Action routine numbers

Every input field must be allocated a *user action routine number* (AR*n*) where *n* is in the range 2 to 30. Constants do not have a user action routine number.

Each format must include the standard action routine AR1 as the *initial action routine*. This is not associated with any input field but is used by the IDE system to set up the screen format and blank any data-field specified. Section 3.2.2.1 describes this initial action routine.

Each format must also inlcude the standard acltion routine AR31, termed the *send action routine*. This is used by the system to write the data from the screen to diskette records when the SEND key is pressed. Section 3.2.2.2 describes this action routine.

Figure 3.3 shows user action routine numbers allocated to a layout.

Notes:

1 A user action routine number simply nominates the user action routine containing the validation associated with that data-field. If exactly the same validation is required for more than one data-field then these data-fields should be allocated the same user action routine number. In Figure 3.3 the data forms a repeated sequence, so the action routine numbers associated with the first line also apply to subsequent lines. However, if the same validation is required for input fields of different lengths they will need different user action routines, as each user action routine must specify the length of the field to which it applies

2 If each input data-field on the screen layout requires a different validation action then only 29 data-fields can be used since IDE restricts the number of action routines to 31 per format

3.1.4 Completing the data-field definition

A sheet containing data-field definitions should be completed for each screen format designed. Figure 3.4 shows an example of a data-field definition sheet. Data-field definitions should be completed as follows:

1 FORMAT NO This should be a two digit number unique for each screen layout and greater than 15 since screens 01 to 15 are reserved for use by the IDE system

2 NAME A meaningful name for the layout

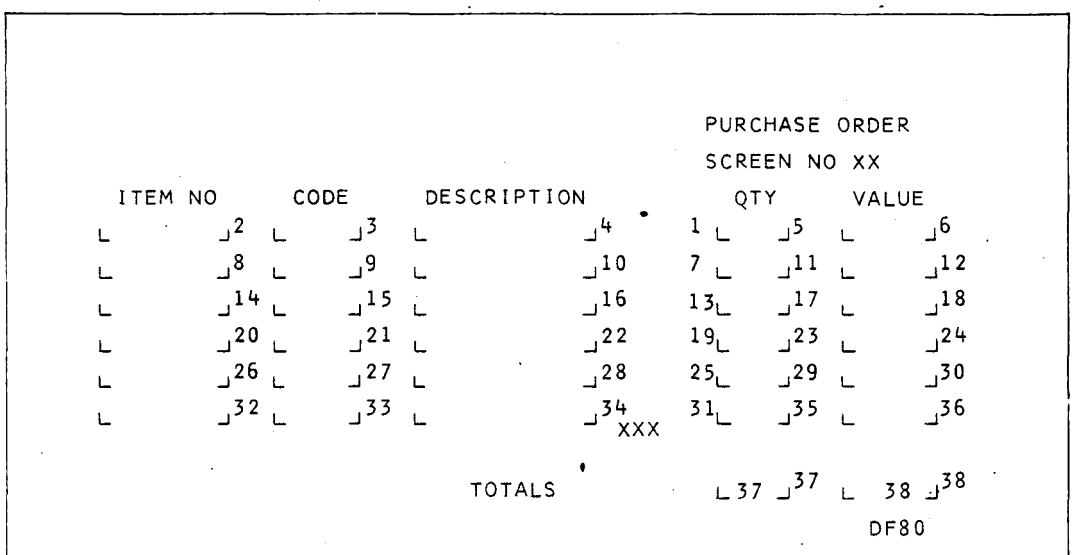
3 FIELD This contains the data-field numbers defined for this screen

4 START This is the start position of the data-field. For input data-fields it is the first character position after the start of field delimiter. The character position is defined as l.c where:

l is the line number (01 to 25)

c is the character number (01 to 80)

Note: The Screen Format Definition screen may be used to calculate or check these data-field positions (see section 3.3.2.4).



XXX is the value of the repeated constant, DF1, 7, 13, 19, 25 and 31 are the data-field numbers of the constants which will be written to diskette before their respective ITEM NOs

Figure 3.2 The allocation of data-field numbers

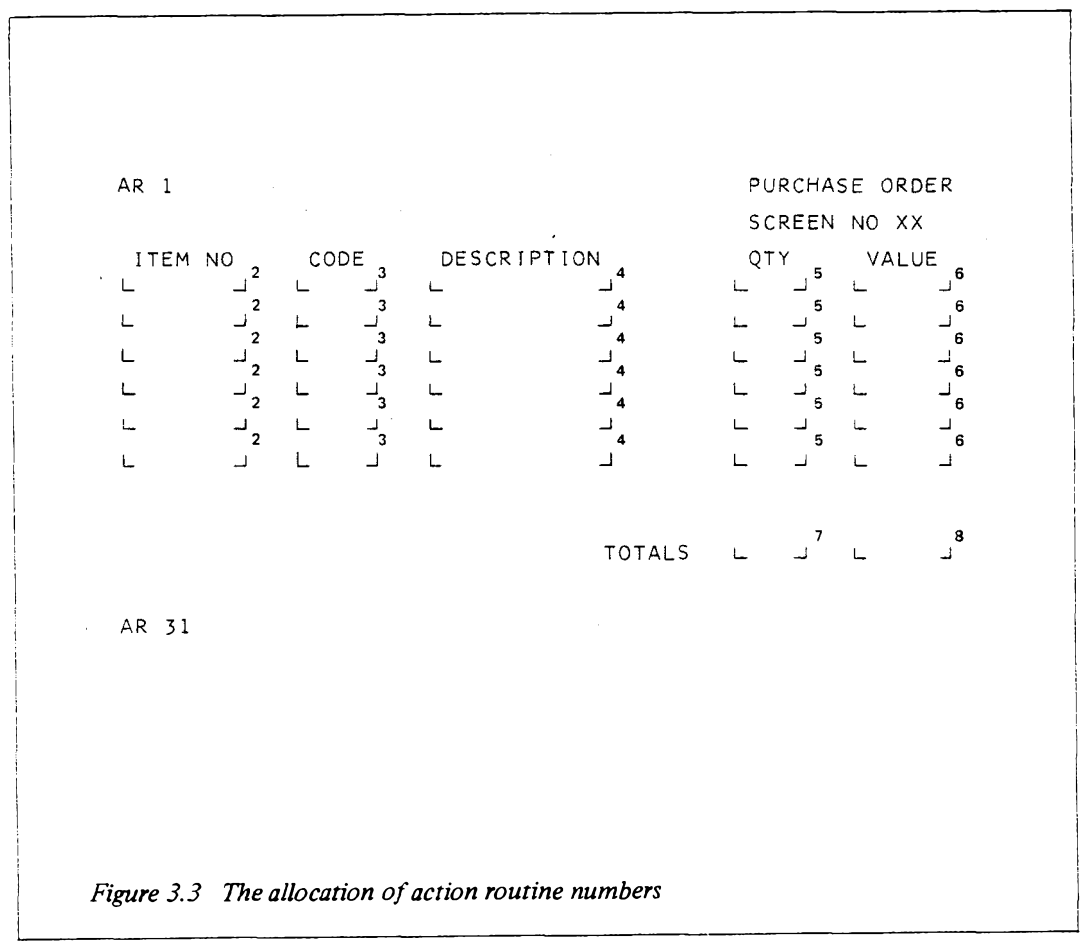


Figure 3.3 The allocation of action routine numbers

Field Definition for: Format no Name Purchase Order

Field	Start	Length	Input?	AR no.	Transmit?	Dec Plcs	Insert/Verify
DF1	15.43	3			Y		Y
DF2	05.03	5	Y	2	Y		Y
DF3	05.10	3	Y	3	Y		Y
DF4	05.15	20	Y	4	Y		
DF5	05.50	2	Y	5	Y		Y
DF6	05.55	5	Y	6	Y	2	Y
.							
.							
.							
DF80	24.69	12				3	

This shows the data-field definitions for the first data line (line 5) of the screen described in Figures 3.1 to 3.3. DF1 is the repeated constant explicitly defined in line 15

Figure 3.4 Data-field definitions

5 LENGTH This is the number of characters in the field

6 INPUT? Y is inserted if data will be keyed into the field being described. If the data-field contains a constant, this field should be left blank

7 AR No. This is the user action routine number allocated to the field when the screen was designed

8 TRANSMIT? Y is inserted for all data-fields that are to be copied to the diskette records. Usually all data-fields will be transmit fields

9 DEC PLACES The number of decimal places in the field, if any, is specified here. If none this field should be left blank

10 INSERT/VERIFY Y is used to indicated all the data-fields that require verification

Data-field 80 should be defined for every format as shown in Figure 3.4.

3.2 Writing action routines

The user action routines define the validation to be carried out on each data-field. Having decided on the validation required for the screen contents the approriate user action routines can be built up from the IDE global action routines. Several global action routines can be combined in a single user action routine referring to one data-field. It is the global action routines which perform the particular validation required.

The global action routines are entered using TPL instructions. Each instruction consists of an optional *label*, a *verb* and up to three *operands*.

A user action routine will consist of code to perform the following steps (section 3.2.1 specifies the detailed coding):

1 Load the terminal working storage registers with any constants or field lengths required by the global routine that is to be called. The registers are either numberic (NR*n*) or character (CR*n*) where *n* is the number of the register. A TPL verb is used to load the registers; MOVEL for characters and MOVEN for numbers. Section 3.2.1 defines the global routines' requirements

2 If required by the global routine, set the boolean indicators ON or OFF before entering the routine

3 Enter the global action routine using the ENTERG verb

4 Repeat from item 1 for all the global action routines required in this user action routine

5 Load the register NR1 with the field length of the current field and enter global routine 4. This routine collects the operator keying statistics and must therefore be entered for every input field

6 Use the EXIT verb to pass control back to the video so that the operator can type into the next data-field. This is only required if constants are specified in the action routine

7 Specify any constants required by the global action routines entered. Constants are specified using the CONST verb and the label CC*n* where *n* defines the label number

8 Use the END verb to indicate the end of the user action routine. This verb is mandatory. It will pass control back to the video if this has not already been done by an EXIT instruction

If a data-field requires no validation its action routine will consist simply of the entry to global routine 4 and the END instruction.

Every screen defined must have an initial action routine. The coding for this is shown in section 3.2.2.1.

Every screen defined must also have a send action routine. The coding for this is detailed in section 3.2.2.2.

Coding sheets should be completed for each user action routine required by the screen including the initial and send routines. One sheet should be completed per action routine.

Figure 3.5 shows an example of action routine coding linking more than one global action routine together.

If it is necessary to construct more complex processing of the screens than provided by the IDE global routines this can be written in TPL and included in the user action routines. For a full description of TPL see *TPL Application Programming*.

The IDE global routines for validation are listed in section 3.2.1 together with the TPL coding required for the user to enter each routine. The standard user action routines to call the system functions are listed in section 3.2.2.

3.2.1 Global action routines

3.2.1.1 *WRITE SCREEN TO DISKETTE Global routine 1*

This is one of the system functions provided by IDE. It is incorporated into the send action routine for each data entry screen as specified in section 3.2.2.2.

3.2.1.2 *DISPLAY NEW SCREEN FORMAT Global routine 2*

This system function is incorporated into the initial action routine for each data entry screen as specified in section 3.2.2.1.

3.2.1.3 *LOAD NEXT FORMAT Global routine 3*

This system function provided by IDE is incorporated into the send action routine for each data entry screen as specified in section 3.2.2.2.

Action Routine coding for screen format no. XX

AR no. 3 Batch totalling

Label	*Verb*	*Operands* a	b	c	*Comment*
	ENTERG	11			Field present check
	ENTERG	13			Numeric check
	MOVEN	2	NRI		Batch total
	ENTERG	26			
	MOVEN	4	NRI		Field length 4 characters
	ENTERG	4			
	END				

Figure 3.5 Action routine coding

3.2.1.4 *COLLECT OPERATOR STATISTICS Global routine 4*

This function counts the number of correct keystrokes made by the operator each time a data-field is entered. The count is accumulated and written to a record. Individual counts are recorded for each operator named in the system. The counts are held separately for the entry, modify and verify modes. If the operator using the video has not been named, or her name is not recognised by the system the count is written to a record for that video.

This function should be incorporated into the user action routine for each data-field after any validation has been performed. The coding required to call this function is:

```
label   verb      operands

        MOVEN     x     NR1
        ENTERG    4
```

where *x* is the field-length of the data-field being entered.

3.2.1.5 *FIELD PRESENT Global routine 11*

This routine checks that the field contains at least one non space character. This routine must be the first global routine entered for any field which would be invaliid if left blank. This is because all other global routines accept a competely blank field as valid. This routine is entered as follows:

```
label   verb      operands

        ENTERG    11
```

3.2.1.6 *ALPHABETIC Global routine 12*

This routine checks that all characters in a field are in the range A to Z or space. This is entered as follows:

```
label   verb      operands

        ENTERG    12
```

3.2.1.7 *NUMERIC Global routine 13*

This routine checks that all characters in a fields are in the range 0 to 9. this is entered as follows:

```
label   verb      operands

        ENTERG    13
```

3.2.1.8 *ALPHANUMERIC Global routine 14*

This routine checks that all the characters in a field are in the range A to Z, 0 to 9 or space. This is entered as follows:

```
label   verb      operands

        ENTERG    14
```

3.2.1.9 *FULL NUMERIC Global routine 15*

This routine checks that the field contents obey the following rules:

1 Optional leading spaces followed by

2 Optional positive or negative sign followed by

3 Characters 0 to 9 including an optional decimal point followed by

4 Optional trailing spaces

This routine is entered as follows:

```
label   verb      operands

        ENTERG    15
```

3.2.1.10 *INTEGER Global routine 16*

This routine checks that the current field contains an integer. Leading and trailing spaces are allowed. The routine is entered as follows:

```
label   verb      operand

        ENTERG    16
```

3.2.1.11 *OVERPUNCH CHECK Global routine 17*

This routine checks that the current data-field contains only the characters 0 to 9 except the first character which may be an overpunch sign. The field length must be moved to NR1 before entering the routine.

```
label   verb      operands

        MOVEN     x     NR1
        ENTERG    17
```

where *x* is the field length.

3.1.1.12 *FLOATING POINT CHECK Global routine 18*

This routine checks that the field contains a valid floating point number; that is an optionally signed number followed by an exponent. This field will be rejected if any spaces are present. The field length must be entered in field NR1 before calling the routine.

```
label   verb      operands

        MOVEN     x     NR1
        ENTERG    18
```

where *x* is the field length.

3.2.1.13 *FULL FIELD CHECK Global routine 19*

This routine checks that a field does not contain any space characters. The field length must be moved to field NR1 before entering the routine.

```
label   verb      operands

        MOVEN     x     NR1
        ENTERG    19
```

where *x* is the field length.

3.2.1.14 *RANGE CHECK Global routine 20*

This routine checks that the contents of the field are within a given range of values. The range check may be numeric or character. For a numeric check the maximum field length is 12 characters. For a character check the maximum field length is eight characters. The limits for the range check are set up as constants. These are moved into the fields CR1 and CR2 for character checks and into fields NR1 and NR2 for numeric checks. If a character check is required, boolean 1 is set ON before entering the routine.

When defining the limits they must be set up as eight or twelve character constants, whatever the length of the input field. These limits are defined in groups of four characters as shown in the examples. Character range checks of less than eight characters must have the remaining character positions up to eight specified as spaces. For numeric range checks the required limits should be aligned at the right hand end of the constant preceded by spaces or zeros to fill unused character positions. A sign and/or a decimal point can be included in a numeric constant. The following examples show the coding required to check a character range and a numeric range:

Example 1 This checks that input to a five charracter field is in the range ABCDE to AZZZZ.

```
label   verb      operands      comments

        MOVEL     CC1   CR1     moves in lower limit
        MOVEL     CC2   CR2     moves in upper limit
        ON        BL01          indicates a character comparison
        ENTERG    20
        MOVEN     5     NR1     field length 5  mandatory for all action
        ENTERG    4
        EXIT                                    routines
CC1     CONST     8             maximum length defined
        CONST     ABCD          }lower limit 8 characters
        CONST     E             }defined ABCDE
CC2     CONST     8
        CONST     AZZZ          }upper limit AZZZZ
        CONST     Z             }
        END                     mandatory for all action routines
```

Example 2 This checks that a five character input field contains a number in the range 10,000 to 99,999.

```
label   verb      operands    comments

        MOVEN     CC1   NR1   moves in lower limit
        MOVEN     CC2   NR2   moves in upper limit
        ENTERG    20
        MOVEN     5     NR1   }collects operator
        ENTERG    4           }statistics
        EXIT
CC1     CONST     12    )     defines the length of constant
        CONST     ▼▼▼▼  {
        CONST     ▼▼▼1  {       lower limit 10,000
        CONST     0000  )
CC2     CONST     12    )
        CONST     ▼▼▼▼  {
        CONST     ▼▼▼9  {       upper limit 99,999
        CONST     9999  )
        END
```

3.2.1.15 *DATE CHECK Global routine 21*

This routine checks that the date in the field is a valid date. Dates can be in the form DDMMYY or DD/MM/YY. This check is called as follows:

```
label   verb      operands

        ENTERG    21
```

3.2.1.16 *LEFT ZERO Global routine 22*

This routine is entered to right justify the contents of a field. All non-significant character positions are set to zero. A sign, if present in the first significant position in the input field, is positioned in the leftmost character position. The user must move the field length into NR1 before entering this routine. The following coding will left zeroise a 10 character field:

```
label   verb      operands

        MOVEN     10   NR1
        ENTERG    22
```

The left zero action will not be effective until the field is verified. However it is possible to left zeroise a field that is not to be verified by setting Boolean 1 ON before entering the routine. In this case the coding required will be:

```
label   verb      operands

        ON        BL01
        MOVEN     x    NR1
        ENTERG    22
```

where *x* is the length of the field.

3.2.1.17 *SPACE FILL Global routine 23*

This routine is called to right justify the contents of a field. It fills the non-significant character positions with spaces. The coding required is as follows:

```
label   verb      operands

        ENTERG    23
```

The space fill action taks place when the field is verified. If the field is not defined for verification, then in order to space fill, Boolean 1 must be set ON before entering the routine. In this case the required coding will be:

```
label   verb      operands

        ON        BL01
        ENTERG    23
```

3.2.1.18 *DUPLICATE Global routine 24*

This routine is incorporated into the user action routine for any field which contains or is to receive duplicated data. Before entry to this routine a number in the range 1 to 16 is moved into field NR1 and this number is then used to identify the duplication required as described below. A maximum of 16 different fields may be duplicated in a batch.

The routine examines the current data-field. If the field contains data, a copy of this data together with it's identifier (the number in NR1) is stored in the terminal memory. If the field is empty the value in NR1 becomes the key used to search the stored data. When a match of identifiers is found that stored data is duplicated into the current data-field.

Duplication of data can take place within a screen and between the screens in a batch. The data to be duplicated must appear before the blank fields into which it is to be written.

Fields for duplication have a maximum length of 12 characters.

The following coding is required to enter this routine:

```
label   verb      operands

        MOVEN     x    NR1
        ENTERG    24
```

where *x* is the identifier in the range 1 to 16.

3.2.1.19 *GET DATE Global routine 25*

This routine is used to insert today's date into the field. This field can be six or eight characters depending on the installation standard. The six character format is DDMMYY, the eight character format is DD/MM/YY. If the operator leaves the field blank, when the TAB key is pressed today's date is

moved into the field and displayed. The operator can override this action by entering the required date in the field. No parameters are needed and the routine is entered by the following instruction:

```
label   verb      operands

        ENTERG    25
```

3.2.1.20 *BATCH TOTALLING Global routine 26*

This routine must be called for any field that is to be added into a batch total. A validation check must ensure that the field is numeric before entering this routine. Two batch totals can be accumulated per batch and to indicate which batch total the field is to be added into,, the appropriate batch total number must be moved into field NR1. This number is either one or two. The example below will add the value in the field into batch total two:

```
label   verb      operands

        MOVEN     2   NR1
        ENTERG    26
```

3.2.2 Standard user action routines

These user action routines do not refer to specific data-fields and therefore do not require entry to global routine 4.

3.2.2.1 *INITIAL ACTION ROUTINE User action routine 1*

Each format must have an initial action routine. In its simplest form this routine will consist of a call to global routine 2.

```
label   verb      operands

        ENTERG    2
        END
```

This will display the screen format. However if any fields in the format are to be blanked additional coding will be required. It is normal to blank any constants that have been included in the format. Blanking is done by using the following statement:

```
label   verb      operand

        VARBL     DFx   DFy   DFz
```

where *x*, *y* and *z* are the data-field numbers of the fields to be blanked (normally constants).

A single VARBL statement can blank one, two or three data-fields and the statement can be repeated as many times as necessary to blank all the required fields.

For example: These statements blank out the values defined on the screen format as DF4, DF10, DF16 and DF22.

```
label   verb      operands

        VARBL     DF4   DF10   DF16
        VARBL     DF22
        ENTERG    2
        END
```

Where are repeated constant has been defined only once explicitly on the screen, the blanking statement need only refer to the data-field number accorded to this explicitly defined constant.

3.2.2.2 *SEND ACTION ROUTINE User action routine 31*

This action routine must be specified for every screen format in the system. It must always be specified as user action routine 31.

The data input on a screen will be written to diskette as one or more records. Each output record can contain a maximum of 120 characters.

The send action routine gives details of the screen data-fields and constants to be placed in each output record and specifies the output records' length (where this is fixed). The routine writes these records to diskette and calls the next screen format required for data entry.

If the screen data is to be split into 10 records or less the field number of the last field to go into each record is moved into register NR*n* (where *n* is in the range 1 to 10). The registers must be used in ascending sequence of *n* with no gaps. The first unused register NR*n* must be set to zero unless all 10 are used.

If all the records generated by the screen are of the same length this record length is moved into register NR11. If variable length records are required or the record length is not known, NR11 is set to zero. Global action routine 1 is then entered to write the specified records to diskette.

If more than 10 records are required the procedure above is repeated for each set of 10 records until the whole screen has been written to diskette. For these subsequent entries to global action routine 1, Booean 1 must be set ON.

Having written the records to diskette the send action routine must load the next screen format. This requires entry to global routine 3.

If the current screen format is to be loaded again entry to global routine 3 will suffice. If, however, a repeating series of different screen formats is required for one data entry task, the send action routine can be coded to call the next format in the series. This is achieved by loading the format number of the next screen into register CR12 and moving CR12 into an IDE system table immediately before calling global routine 3. The coding for this is detailed in the example.

Note: The send routine for the last format in the series should load the first format to restart the series.

The send action routine example below shows the coding required to split a screen into five variable length records with:

record 1 containing data-fields 1 to 7
record 2 containing data-fields 8 to 14
record 3 containing data-fields 15 to 21
record 4 containing data-fields 22 to 29
record 5 containing data-fields 30 to 40

Only fields defined as transmit fields in the field definition are inlcuded in the records.

This coding also loads the new screen format numbered 20.

```
label   verb      operands

        MOVEN     7          NR1
        MOVEN     14         NR2
        MOVEN     21         NR3
        MOVEN     29         NR4
        MOVEN     40         NR5
        MOVEN     0          NR6
        MOVEN     0          NR11
        ENTERG    1
        MOVEN     20    )    CR12
        ITEM      2     }    1       This coding sets up the next format number
        MOVEL     CR12  )    TF30
        ENTERG    3
        END
```

Note: The ITEM verb specifies the system table referred to by TF30. It should always be coded as specified in the example above.

3.3 Modifying the Basic Data Entry path

To create a path to inlcude the designed data entry screen formats the IDE Basic Data Entry path is modified. This is done using the Application Programming Development System program T2AD01. This Development System allows the Data Entry path description to be built up on the video screen; it then writes this description to diskette for subsequent use.

Before the Development System can be used to modify the Basic Data Entry path the IDE system must be installed as described in Appendix 2.

Six packages are provided as part of the Development System. Each performs a specific function in the task of creating or amending the Basic Data Entry path description.

The six packages are:

1 THE DEVELOPMENT PACKAGE This is used to amend the IDE Basic Data Entry path to include the data entry screen formats and processing

2 THE CREATE LIBRARY PACKAGE This allows an initial path library file to be created. This package is required only when setting up IDE

3 THE TIDY LIBRARY PACKAGE This provides the facility to copy to another file and tidy a path library file that has become fragmented through frequent deletions and additions

4 THE SIZING PACKAGE This should be used to determine the size of the individual screen formats in terms of the store space occupied

5 THE CLEAR FILE PACKAGE This package can be used to clear library files

6 THE PART COPY PACKAGE This allows elements of one path to be copied to another. It is used to generate additional copies of the IDE Basic Data Entry path software when more than one data entry path is required. This will be necessary if more than 84 different data entry screen formats are required and when a new version of the IDE software is issued

The remainder of tis chapter describes input to the development package. The other packages in the Development System are described in Appendix 2.

The Development System provides a hierarchy of screens into which the data entry path details are typed. Figure 3.6 shows the Development System screens provided.

The first screen in the hierarchy is the Development System path catalogue. This provides access to the different packages described above.

Selecting the development package causes the IDE path catalogue to be displayed. This provides access to the four IDE paths described in section 2.5.

Each IDE path has a header screen listing all the screen formats in that path.

Other screens are provided by the development package to input the data entry screen designs and their associated user action routines. These development package screens and how to use them are described in sections 3.3.2.3 to 3.3.2.6.

3.3.1 Loading the development system

Ensure that the following items are available on site:

1 A terminal executive library diskette containing T2AP11, the controlling executive for the development program T2AD01

2 A development system diskette containing T2AD01; this is issued in file DEVLIB

3 The diskette containing the IDE system normally in file IDEDEVEL (see Appendix 2)

4 An application development PID, numbered 254

5 The screen layouts and associated documentation previously designed

Switch on the terminal. Then to laod the Development System to input the screen formats proceed as follows:

1 Switch on the lowest-numbered video

2 Insert the terminal executive library diskette containing T2AP11 into drive 1 of the floppy disc drive or drive 2 if drive 1 is inoperable

3 Ensure that the rotary switch is in position 3 then lift and press down the genres/activate switch. The diskette menu select display will then appear

4 Insert the diskette containing the development system file T2AD01

5 Press the blue numeric key(s) corresponding to the display number of T2AP11, and then press the SEND key. The message T2AP11 APPLICATION PROGRAMMING EXECUTIVE will appear on the screen

6 Insert the application development PID into the PID reader

7 Remove the terminal executive library diskette from drive 1, and then insert in its place the diskette containing the path library development file, normally IDEDEVEL

8 Press the LOCAL INPUT key and type CE D to configure the terminal executive for the Development System

9 Press the SEND key

10 (a) Press the LOCAL INPUT key and type OV once to open all volumes (diskettes)

(b) Press the SEND key

11 Press the LOCAL INPUT key

12 Type GA *filename*$_1$,D,R

where *filename*$_1$ is the name given to the Development System file, normally DEVLIB

13 Press the SEND key to assign the Development System file

14 Press the LOCAL INPUT key

15 Type GA *filename*$_2$,U,W

where *filename*$_2$ is the name given to the path library development file, normally IDEDEVEL

16 Press the SEND key to assign the path library development file

17 Press the LOCAL INPUT key

18 Type AD

19 Press the SEND key. The Development System path catalogue display will now appear on the screen (see Figure 3.7)

Note: When inputting commands a space must be left between the command and the parameter following it, for example:

CE▽D
GA▽ *filename*,D,R

3.3.2 Running the development package to create the Basic Data Entry path description

3.3.2.1 *Obtaining the IDE path catalogue*

Having loaded the Development System the Development System path catalogue display shown in Figure 3.7 will appear on the screen. When this display appears proceed as follows:

1 Type A into the field marked *Path*

2 Press the TAB key

The IDE path catalogue display shown in Figure 3.8 will appear.

3.3.2.2 *Selecting the Basic Data Entry path*

When the IDE path catalogue appears proceed as follows:

1 Type D into the field marked *Select path for amendment*

2 Press the TAB key

The Basic Data Entry path header shown in Figure 3.9 will be displayed.

Note: It is not possible nor is it necessary to amend the IDE system paths other than the Basic Data Entry path.

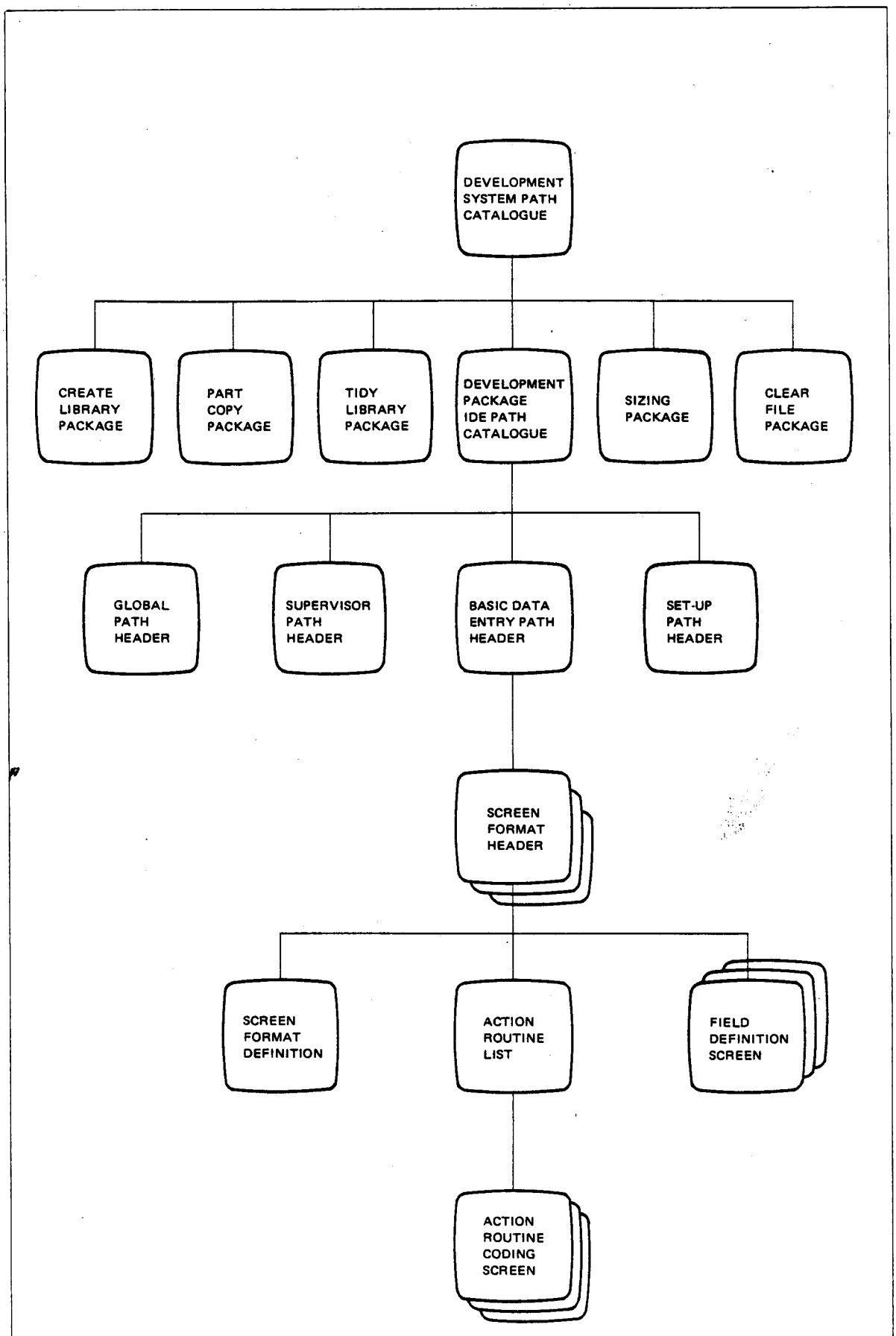


Figure 3.6 Hierachy of Development System display screens

3.3.2.3 *Creating the data entry screen format header*

When the Basic Data Entry path header display appears proceed as follows:

1 Type F_x into the field *Item/Type No.* where x is the format number of the new screen as specified during the design phase.

Format numbers F1 to F15 are reserved by the IDE system

2 Press TAB. A blank screen format header will then appear (see Figure 3.10)

When the screen format header is displayed proceed as follows:

1 Type into the field marked *Name* the name of the screen format being defined and press TAB

2 Enter into the field marked *Next Format* the number of the format being defined and press TAB

3 Move the cursor to the field marked *Mandatory*. Enter Y in this field and press TAB

4 Type 1 into the field marked *Initial AR No* and press TAB

5 Type 31 into the field marked *Send AR No* and press TAB

6 No other fields are required by IDE so press CTRL together with 9 on the numeric block to write the completed format header to the path library file. When the message:

STORED SUCCESSFULLY, PLEASE CONTINUE

appears on the bottom line of the screen, continue

7 Enter into the field marked *Select Item* the number corresponding to *Screen Layout* for VT2000 as indicated by the list under the Select Item field. Press TAB to call the format definition display (initially a blank screen)

3.3.2.4 *Inputting the screen format*

Initially a blank screen is displayed into which the screen design should be typed as follows:

1 Type all protected data and constants onto the screen in the positions in which they are to appear, leaving a space immediately before and after the unprotected fields which are to receive input data at run time. The system will put field delimeters in these spaces before the screen is displayed for use

Note: Refer to section 3.1.1 for the positioning of repeated constants.

2 If the start position of each data-field (input fields and constants) was calculated when designing the screen as described in section 3.1.4 continue from step 6. (Steps 3 to 5 can be used to check the calculated start positions)

3 Move the cursor to the first character position of the field whose start position is to be defined

4 Press CTRL together with 0 on the blue numeric block. The message ADDRESS= *start position* will be displayed on the last line of the screen. Make a note of the value of *start position*

5 Repeat steps 3 and 4 to ascertain the start position of each field

6 Press CTRL together with 9 on the numeric block to write the completed screen format to the path library diskette. The field definition display will then appear (see Figure 3.11)

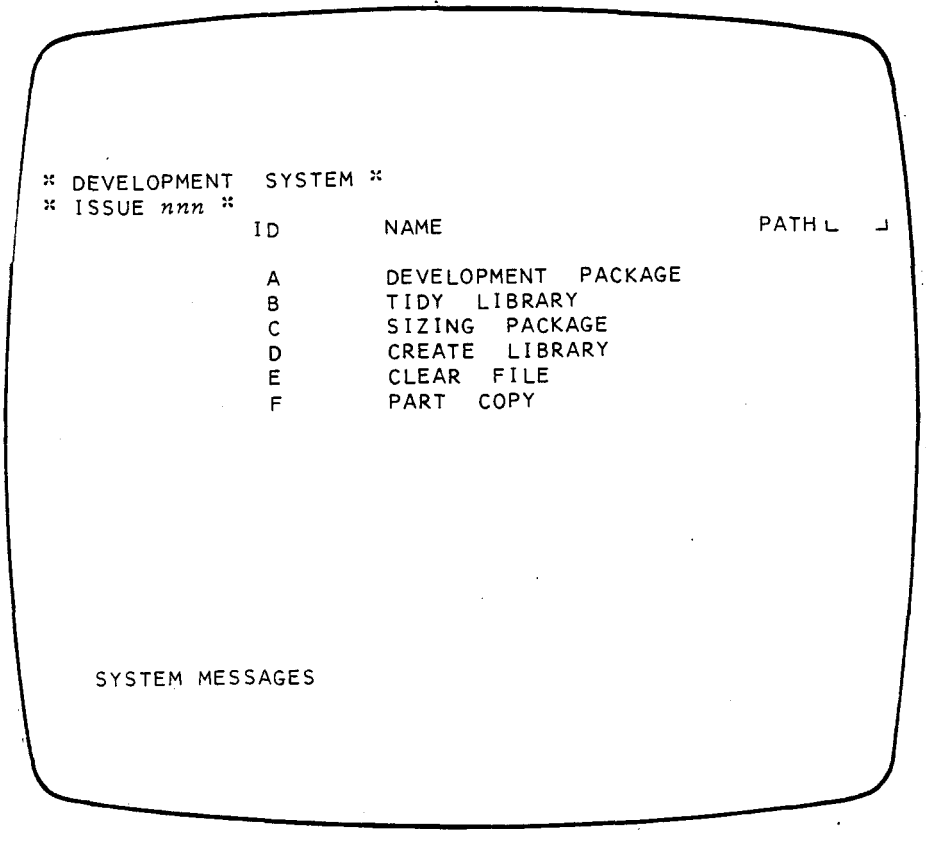


Figure 3.7 Development System path catalogue

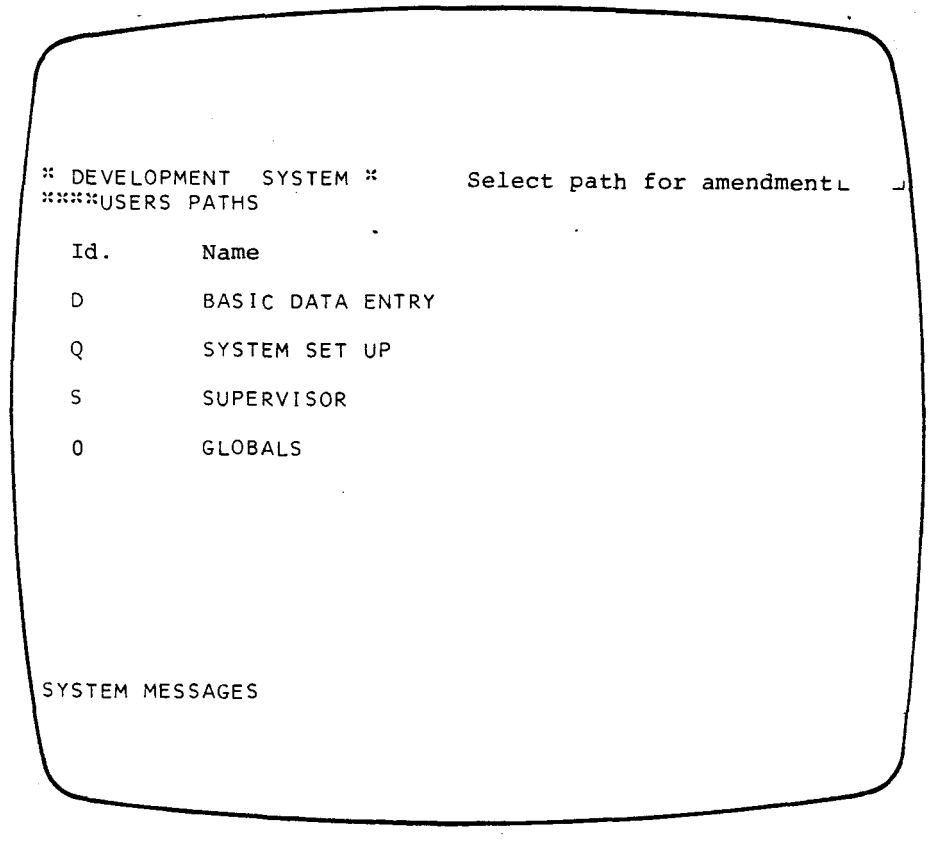


Figure 3.8 IDE path catalogue

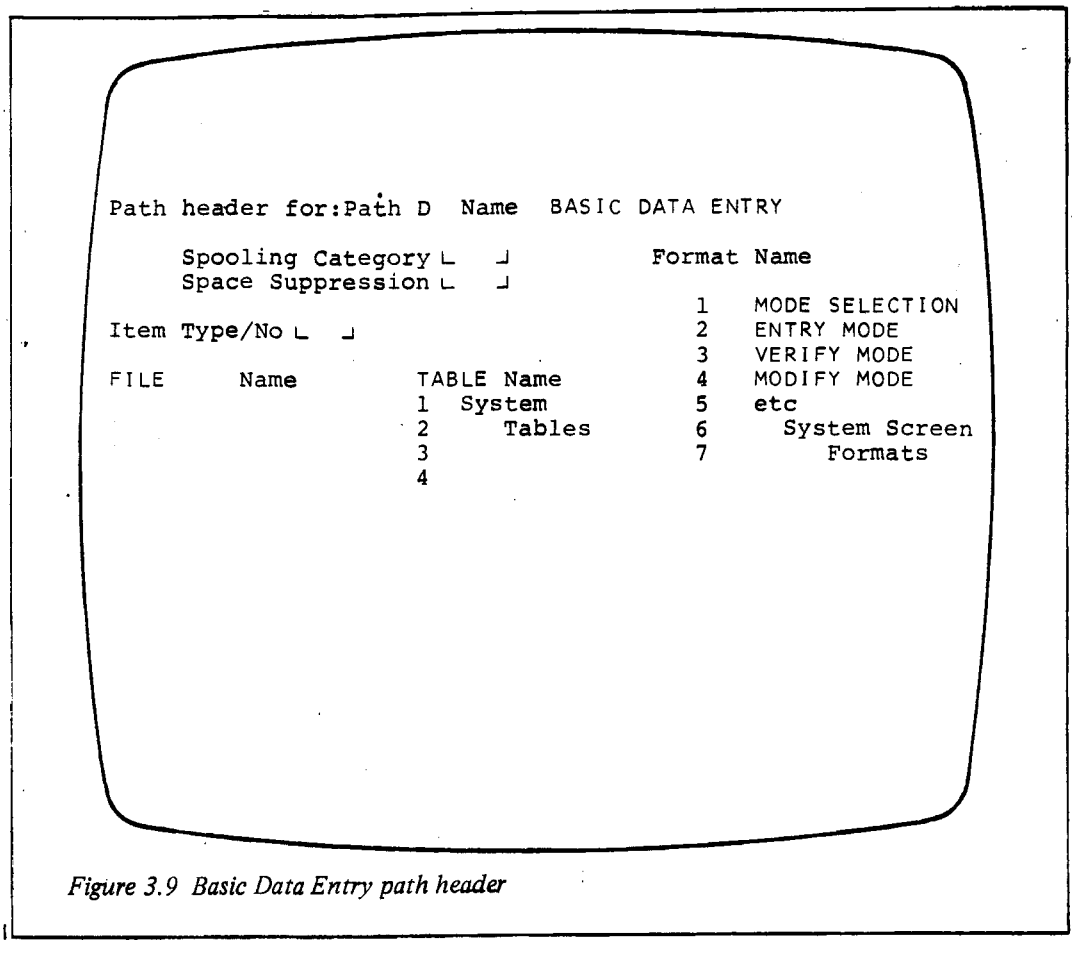


Figure 3.9 Basic Data Entry path header

3.3.2.5 *Inputting the data-field definitions*

When the field definition display appears, proceed as follows:

1 Type into the field marked *Start* the position of the field being defined (ascertained by step 4 of the previous section) and press TAB

2 Type into the field marked *Length* the length of the field being defined, and press TAB

3 Move the cursor to the field marked *Input?*

4 If the field being defined is to have data keyed into it at run-time continue from step 5, otherwise continue from step 8

5 Type Y into the field marked *Input?* and press TAB

6 Type into the field marked *AR No.* the number of the Action Routine entered from the field, and press TAB

7 Move the cursor to the field marked *Dec. Plcs*

8 If decimal places are to be specified for the field being defined, type into the field marked *Dec. Plcs* the number of decimal places to be specified, and press TAB

9 If the contents of the field being defined are to be written to the data entry record on diskette, type Y into the field marked *Transmit?* and press TAB

10 If the field being defined is required to be verified at run-time, type Y into the field marked *insert/verify* and press TAB

11 Repeat steps 1 to 10 for each field to be defined

12 If a further page of the field definition display is required, continue from step 13, otherwise continue from step 20

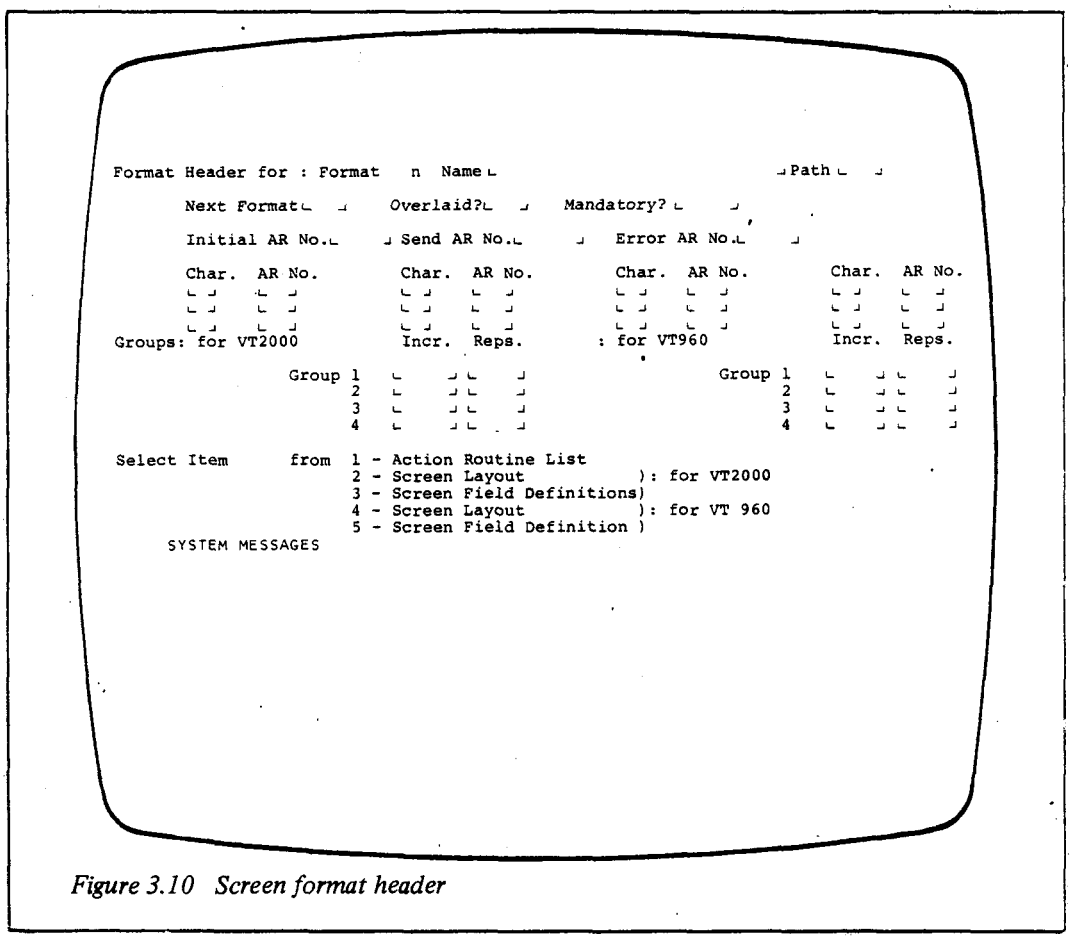


Figure 3.10 Screen format header

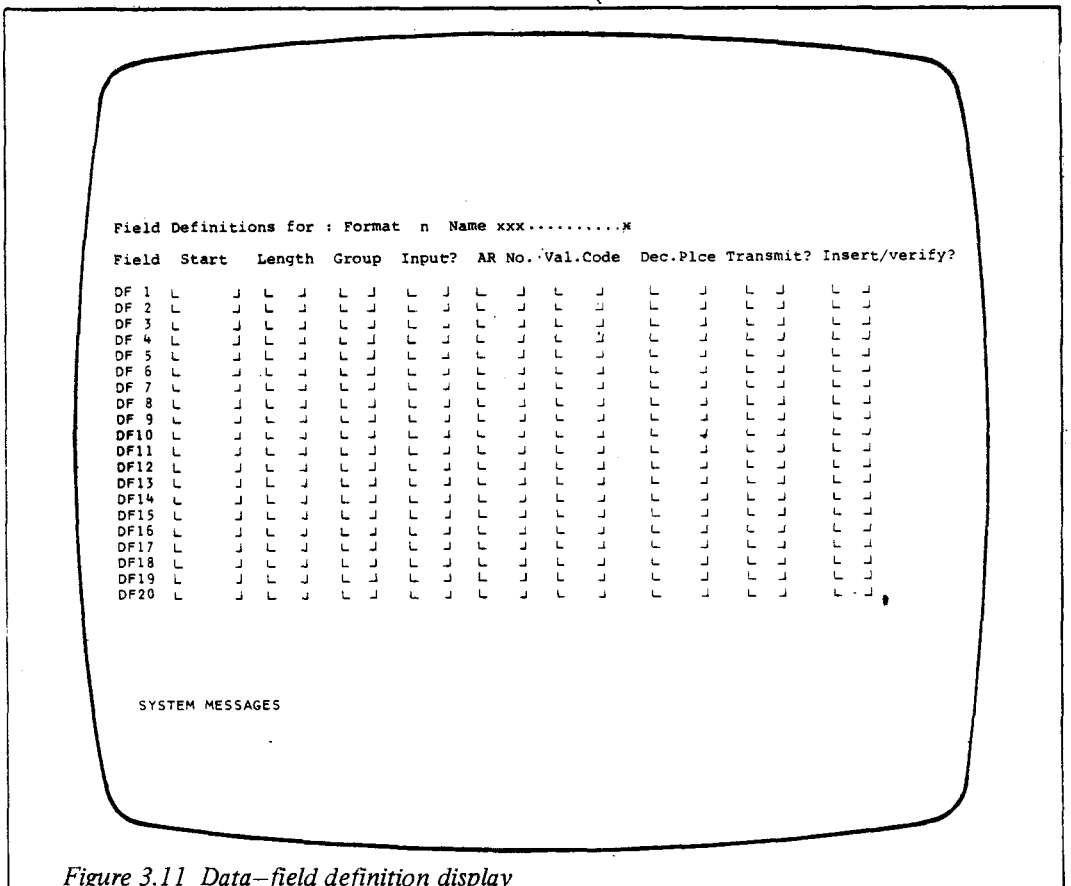


Figure 3.11 Data–field definition display

13 Press CTRL together with 6 on the numeric block to store the current page temporarily

14 Press CTRL together with . on the numeric block to call a continuation page of the display

15 Repeat steps 1 to 12 for the current page of the display

16 If it is desired to return to a previous page of the display, continue from step 17, otherwise continue from step 20

17 Press CTRL together with 6 on the numeric block to store the current page temporarily

18 Press CTRL together with - on the numeric block to display the previous page. Repeat this action if necessary

19 Repeat steps 1 to 13 as desired

20 Press CTRL together with 9 on the numeric block to write the complete field definition display(s) to the path library file. The associated screen format will then be displayed with delimiters

21 Press CTRL together with 1 on the numeric block to return to the format header

3.3.2.6 *Inputting the user action routines*

When the screen format header shown in Figure 3.10 is redisplayed proceed as follows:

1 Type 1 into the field marked *Select Item*

2 Press TAB

The action routine list display, an example of which is shown in Figure 3.12, will appear. This display will be blank if no action routines have been input for this screen. Continue as follows:

3 Enter into the field marked *AR No.* the number of the user action routine that is about to be defined

4 Press the TAB key. The action routine coding sheet display will then appear (see Figure 3.13)

When the action routine coding sheet display appears, proceed as follows:

5 Enter the action routine name into the field marked *Name*, and press TAB

6 If a label is specified for the TPL instruction to be keyed in, enter the label into the field marked *Label*, otherwise leave this field blank. Press the TAB key

7 Enter the verb part of the TPL instruction into the field marked *Verb*, and press TAB

8 If the TPL instruction being defined has operands, enter them into the fields marked *a*, *b* and *c*, pressing TAB after each entry

9 If a comment describing the TPL instruction is required on the coding sheet, enter it into the field marked *Comment*, otherwise leave this field blank. Press the TAB key

10 Repeat steps 6 to 9 for each TPL instruction to be defined. Where a continuation page of the coding sheet is required, continue from step 11, otherwise continue from step 14

11 Press CTRL together with 6 on the numeric block to store the current page temporarily

12 Press CTRL together with . on the numeric block to call a continuation page of the display

13 Repeat steps 6 to 11 for the current page

14 Press CTRL together with 9 on the numeric block to write the complete coding sheet to the path library file. The action routine list will then be redisplayed

15 If further action routines are to be defined for the current screen format, continue from step 3

16 If the current format is the last in the current path, press CTRL together with 2 on the numeric block to redisplay the Development System path catalogue, otherwise continue from step 18

17 If a further screen is to be defined, continue from section 3.3.2.3

18 Press CTRL together with 1 on the numeric block to redisplay the format header, and repeat this process to redisplay the path header (alternatively, press CTRL and 2 to redisplay the Development System path catalogue, and perform sections 3.3.2.1 and 3.3.2.2. The path header will then appear)

19 Continue from section 3.3.2.3 for the definition of subsequent formats on the current path

3.3.3 Ending a development session

All files opened during the development session must be closed before the end of the session. This is achieved by the use of the CD command as follows:

1 Clear the screen by pressing the key marked CLEAR SCREEN and the INTERLOCK key

2 Press the LOCAL INPUT key

3 Type CD

4 Press SEND

The system replies with DONE. New files and terminal executive can now be loaded as required.

3.3.4 Running the development package to amend the Basic Data Entry path description

Having created the data entry path description any of the screens can be subsequently deleted or amended. To do this the Development System is loaded as described in section 3.3.1 and the *control functions* of the numeric block of keys on the video keyboard are used to amend or delete any of the screens accessible to the Development System.

The control functions are divided into two groups:

1 General control functions

2 Private control functions

General control functions are actions that operate on any screen display.

Private control functions operate only on certain individual screen displays of a path description, such as the screen definition display.

All control functions operate on whichever screen display is current when the function is called. Control functions are initiated by the following procedure:

1 Press and hold down the CTRL key on the control block of the keyboard

2 Press the appropriate key of the numeric block

The two types of function are described below.

3.3.4.1 *General control functions*

The general control functions are as follows:

key pressed with CTRL	*Control function*
1	Select next highest display in the Development System hierarchy shown in Figure 3.6
2	Select Development System path catalogue
3	Delete current display
9	Store current display

Details of each of these functions are given below. In the case of the select previous display and select Development System path catalogue functions, any amendments made to the current display since the last store function will be lost. (See below for a description of the store function).

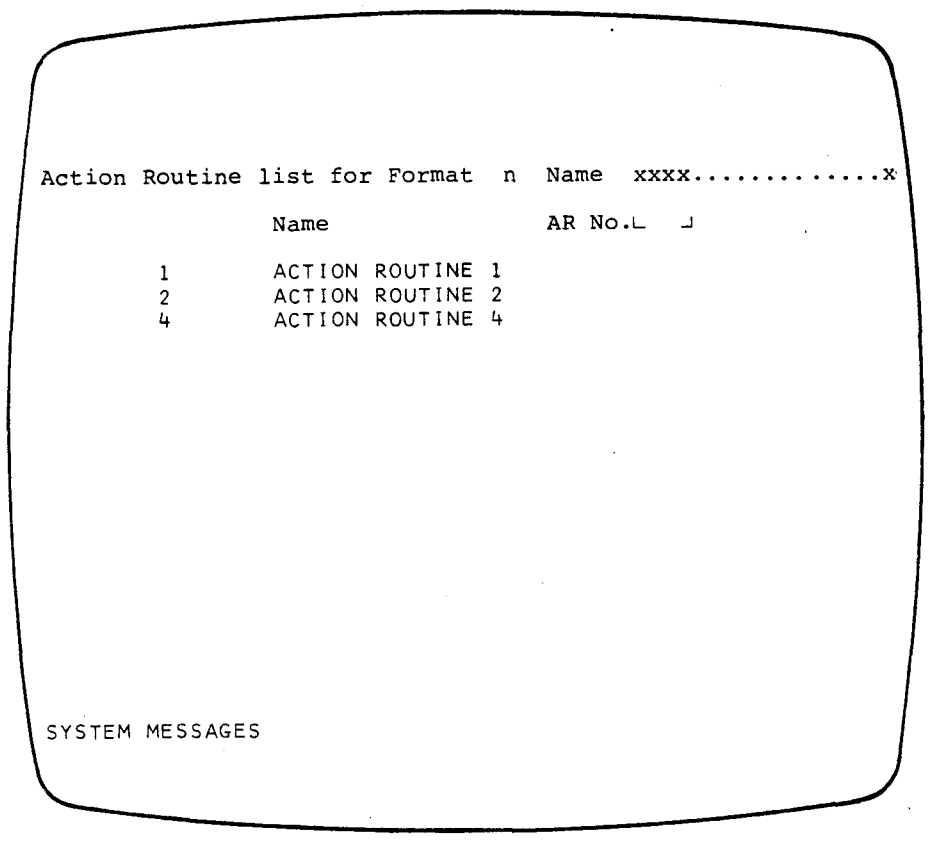


Figure 3.12 Action routine list display

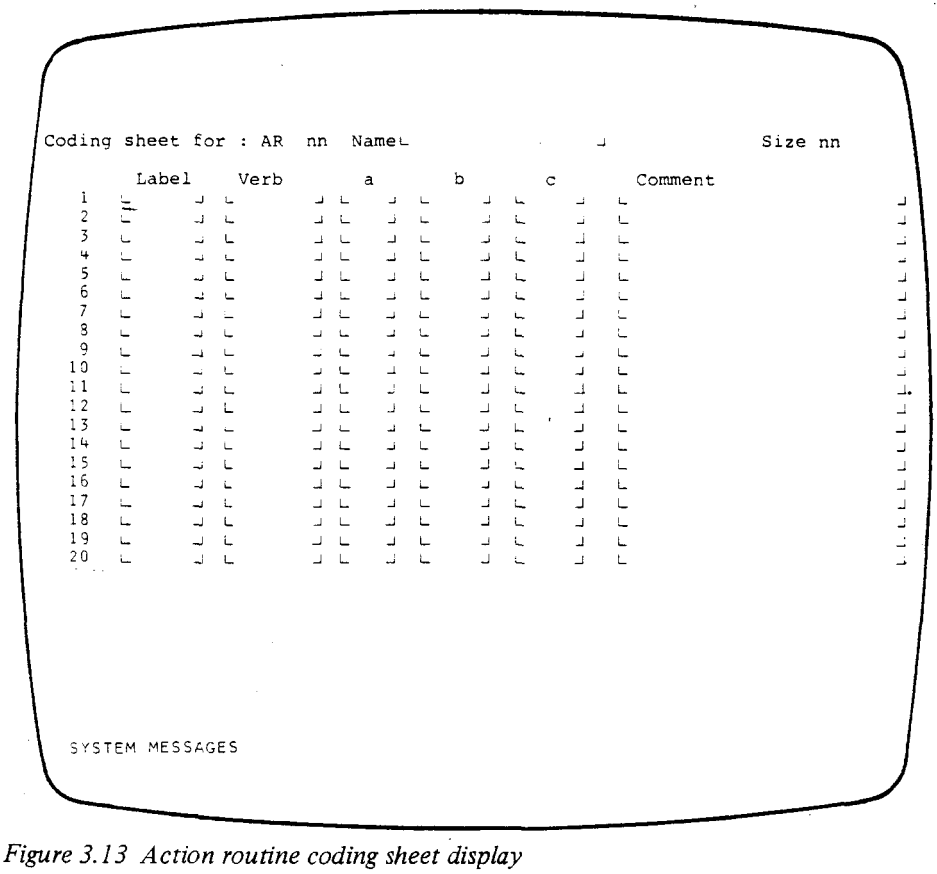


Figure 3.13 Action routine coding sheet display

Select next highest display

Pressing CTRL and 1 allows the user to select the Development System display that forms the next highest stage in the hierarchy (see Figure 3.6). Thus if the current display was the field definition table, pressing CTRL and 1 would cause the format header to be displayed. Sequencing can only proceed upwards along the branches of the tree structure indicated by Figure 3.6.

Select Development System path catalogue

Pressing CTRL and 2 causes the Development System path catalogue to be displayed; from here, the user can select any of the six packages of the Development System.

Delete current display

Pressing CTRL and 3 causes the current display to be deleted from the path library, and selects the display forming the next highest stage in the hierarchy (see Figure 3.6). Where the current display is the path header or format header, the delete function forms one stage in the process of display deletion, which is performed as follows:

1 Press CTRL and 6

2 Press CTRL and 3 to delete the display

This function also causes the deletion of all displays dependent on the current display (that is, those displays that can be reached from the current display via a downward branch on the tree structure of Figure 3.6).

Thus if the current display is a format header, the delete function will delete that format header, the associated screen format and field definition table and all action routine coding sheets associated with the screen format. The path header of the current path will then be displayed.

If the current display is an action routine coding sheet, CTRL and 3 will merely delete that coding sheet and display the action routine list, amended as appropriate.

If the current display is a path header, CTRL and 3 will delete the entire current path description from path header to action routine coding sheets.

Store current display

Pressing CTRL and 9 causes the current display to be written to the current path description on the path library file. Where the current display is the path header or format header, the store function forms one stage in the display storage process, which is performed as follows:

1 Press CTRL and 6

2 Press CTRL and 9 to store the display

3.3.4.2 *Private control functions*

The private control functions are as follows:

Key pressed with CTRL	*Control function*	*Display affected*
6	Update current display	Path header Format header Field definition Action routine coding sheet
7	Display field delimiters Insert TPL coding	Screen format Action routine coding sheet
8	Remove field delimiters Delete TPL coding	Screen format Action routine coding sheet
0	Display field start posiltion	Screen format
.	Page forwards	Screen format Field definition Action routine coding sheet
-	Page backwards	Field definition Action routine coding sheet

Details of each of these functions are given below.

Update current display

Path header and format header displays

These displays require an update operation before they can be stored. Proceed as follows:

1 Press CTRL and 6

2 Amend the display as required

3 Press CTRL and 9 to store the header, or CTRL and 3 to delete the header as required

Steps 1 and 2 may be performed in either order.

Field definition and action routine coding sheet displays

Proceed as follows:

1 Amend the display as required

2 Press CTRL and 6 to store the current display temporarily

These actions must be performed in this order and followed by page forwards or page backwards as appropriate.

Display/remove field delimiters

Pressing CTRL and 8 when the current display is a user-created screen format causes the field delimiters to be removed from the screen. This then permits the use of the block insert/delete text facility on the video; this facility is initiated by pressing INTERLOCK together with either the INSERT or DELETE keys on the control block.

Pressing CTRL and 7 when the current display is a screen format causes the field delimiters to be redisplayed.

Insert/delete TPL coding

Pressing CTRL and 7 when the current display is the action routine coding sheet causes the text NULL to appear in the *Verb* field at the current cursor position, and any lines of TPL coding from the current cursor position onwards to be moved down one line. The NULL line can then be overwritten by any TPL instruction that the programmer may wish to insert at that position.

Pressing CTRL and 8 when the current display is the coding sheet causes the TPL instruction at the current cursor position to be deleted and any subsequent instructions to be moved up one line. Before using these functions CTRL and 6 should be used to preserve any previously made amendments.

Display field start position

Pressing CTRL and 0 when the current display is a user-created screen format causes a numeric indication of the current cursor position to appear on the screen. This function is used when determining the start position of each unprotected field for inclusion on the field definition display. The use of CTRL and 0 causes the following message to appear in the bottom right-hand corner of the screen:

ADDRESS=*l.c*

where

l is the line number of the current cursor position (a number in the range 1 to 24)

c is the screen character position of the cursor (a number in the range 1 to 80).

Page forwards/backwards

The field definition display and the action routine coding sheet are both *paged* displays, that is, each field definition and each set of action routine coding can take up more than one display if necessary.

When paging forwards or backwards, any amendments made to the action routine coding sheet or field definition display contents since the previous store function will be lost unless the following procedure is performed:

1 Perform the amendment as required

2 Press CTRL and 6

3 Page forwards or backwards as required

When the coding sheet or field definition is complete, press CTRL and 9 to store its contents.

Field definition display

Pressing CTRL and . when the current display is a user-created screen format causes the first page of the field definition display to appear. Repeating this action causes further pages of the display to appear. Subsequent pressing of CTRL and - returns the display to the first page of the field definition display, and a further CTRL and - causes the screen format to re-appear. All amendments to the displays during paging will be lost unless the above procedure is carried out. An attempt to page back beyond the screen format will be ignored, as will an attempt to page forwards beyond the last page of the field definition display.

Action routine coding sheet

Pressing CTRL and . when the current display is an action routine coding sheet causes further pages to appear. Subsequent pressing of CTRL and - causes the previous page to re-appear. An attempt to page back beyond instruction 1 is ignored, as in an attempt to page forwards beyond instruction 99.

This chapter describes the features of operating the 7501/7502 videos for data entry under IDE. For a full description of video operating see *Operating 7500 Range Video Terminals* or *7501/7502 Hardware Operating*.

The 7501 and 7502 systems that support the use of IDE consist of a processor with 40K bytes of store and up to two videos on the 7501 and up to four videos on the 7502.

The video is a free standing unit which may be sited up to one kilometre cable length from the processor except on the 7501 where the first video is integral with the processor. The screen has a diagonal measurement of 15 inches and a capacity of 2000 characters (in 25 lines of 80 characters each).

The keyboard on the video is used for data entry. It is physically separate from the display. (See Figure 4.1 for the keyboard layout.)

In addition to the main QWERTY keyboard and shift free numeric block there is a block of control keys. The keying of CONTROL (CTRL) together with any other key is encoded as an *action key*. Action keys always perform a specific function such as clearing the screen. This chapter describes the action keys that are used with the IDE system.

The cursor (_), which appears on the screen, underlines and overlines the position at which the next character typed will be entered. The cursor can be moved around the screen using the cursor control keys which are the arrowed keys on the furthermost right hand block.

Data can be inserted or altered only in unprotected fields on the screen. (For a description of unprotected fields see section 3.1.1.) The cursor can be positioned in protected fields but if an attempt is made to type a character, the keyboard is locked and the audible alarm sounded. The keyboard remains in this state until the INTERLOCK key is pressed; that is the unlabelled white key on the extreme right of the top row of the main keyboard.

After data has been entered into each unprotected field the TAB key is pressed. This moves the cursor automatically to the beginning of the next unprotected field in the sequence specified on the screen format. When the screen is full the SEND key is pressed to write the record to diskette and display the next screen.

When the operator presses the TAB key the IDE system initiates the validation checks on that field. No validation of a field takes place until the TAB key is pressed. It is therefore recommended that whether inputting data, modifying, or verifying it, TAB should always be pressed after typing data into each input field. When inputting or verifying data all the input fields on the screen should be tabbed through before pressing SEND. This makes sure that no fields have been missed. If SEND is pressed before entering data into any field that field will be blank on the diskette record, whatever the validation criteria.

During the data entry operation various control functions are available such as abandoning a screen, closing a batch, etc. To perform these functions the action keys are used. A list of these action keys is given in Appendix 3.

When an error is detected during data entry the system takes the following actions:

1 The audible alarm is sounded and all key depressions are ignored until the INTERLOCK key is pressed

2 The field in error is set flashing and remains in that state until the field is revalidated

3 An error message is displayed on the bottom line of the screen

4 The cursor is positioned at the start of the field in error

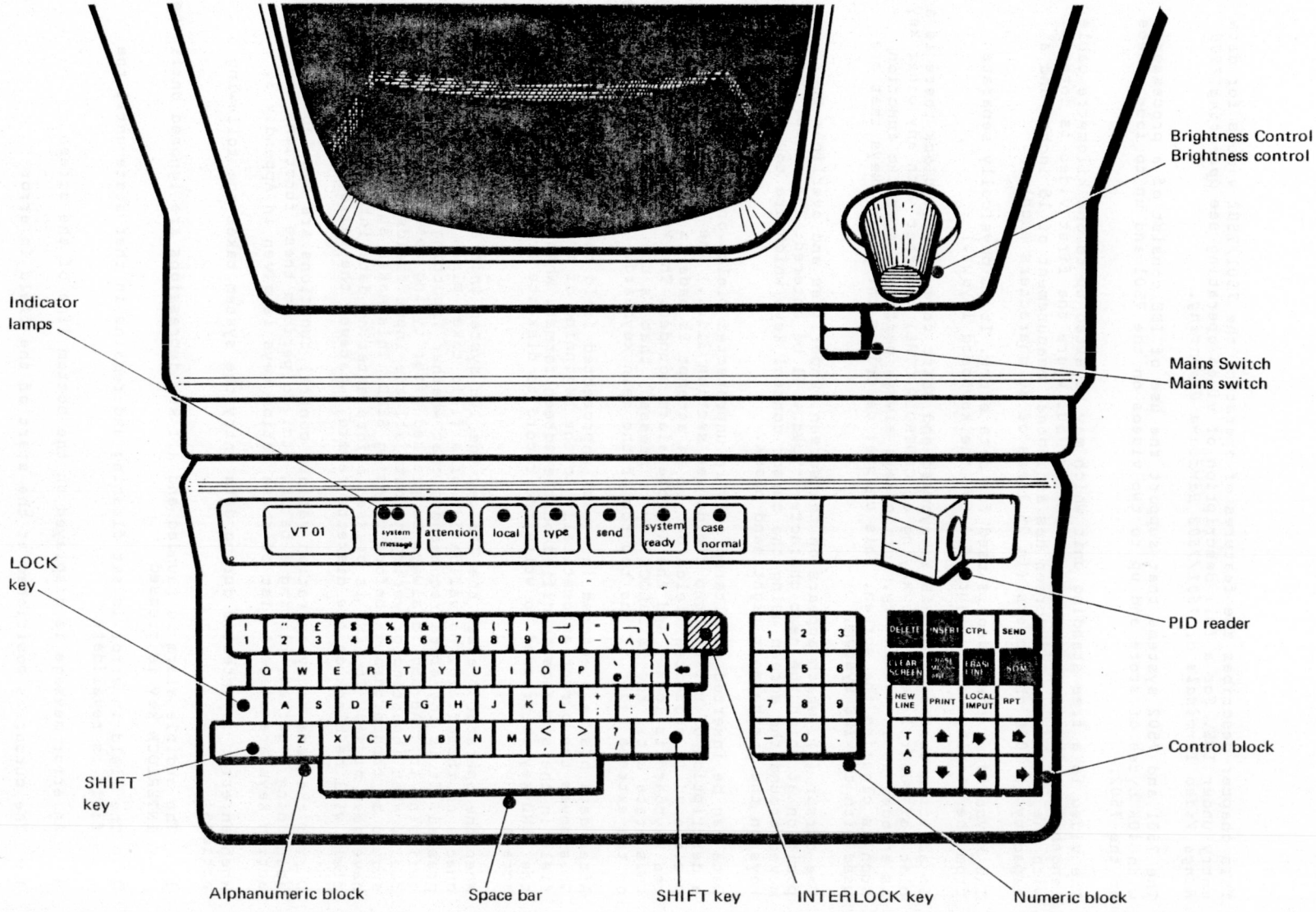


Figure 4.1 Keyboard layout

4.1 Starting a data entry session

4.1.1 Supervisor functions

Before the IDE system can be used for data entry, the diskettes containing the files required by the system must be loaded. Also the appropriate session start up routine must be performed. Details of these supervisor procedures are given in Chapter 5.

4.1.2 Operator functions

Once the supervisor has performed the session start up routine each operator should proceed as described below:

1 If necessary switch on the video. The message SYSTEM READY should be displayed

2 (a) Press the LOCAL INPUT key

(b) Type PC

(c) Press the SEND key. The IDE path catalogue will be displayed on the screen. An example of the path catalogue display is shown in Figure 4.2

3 (a) Type D into the *Path* field of the display. This is the identifier of the Basic Data Entry path

(b) Press the SEND key or the TAB key

This will cause the mode selection screen shown in Figure 4.3 to be displayed unless the session start up routine has not been completed. In that case an error screen will be displayed with the message:

Message	*Meaning/opertor action*
SESSION NOT STARTED	The session has not been started by the supervisor. Wait until the supervisor completes the session start from another video then press SEND to continue. Alternatively action key Z may be used to exit from the Data Entry path

4.2 Input mode selection

The three input modes available are *entry*, *modify* and *verify*. Entry mode is used to input a fresh batch of data. This can be corrected after it has been entered and before it has been verified by using the system in modify mode. When all the necessary corrections and alterations have been made the batch is processed again in verify mode. When successfully verified it can be transmitted to the mainframe.

The appropriate input mode should be selected by typing 1, 2 or 3 into the mode selection screen, indicating data entry, verification and modification respectively. The mode selection screen layout is shown in Figure 4.3.

On selecting a mode the header screen for that mode will be displayed.

The following error message applies to input mode selection:

Message	*Meaning/operator action*
INVALID MODE	The mode entered is not in the range 1 to 3. Reinput the mode and press TAB

Action key Z can be used with the mode selection screen. It causes the IDE path catalogue to be redisplayed.

4.3 Using data entry mode

4.3.1 Completing batch header details

Having typed 1 into the mode selection screen the entry mode header screen will be displayed. Figure 4.4 shows a blank entry mode header.

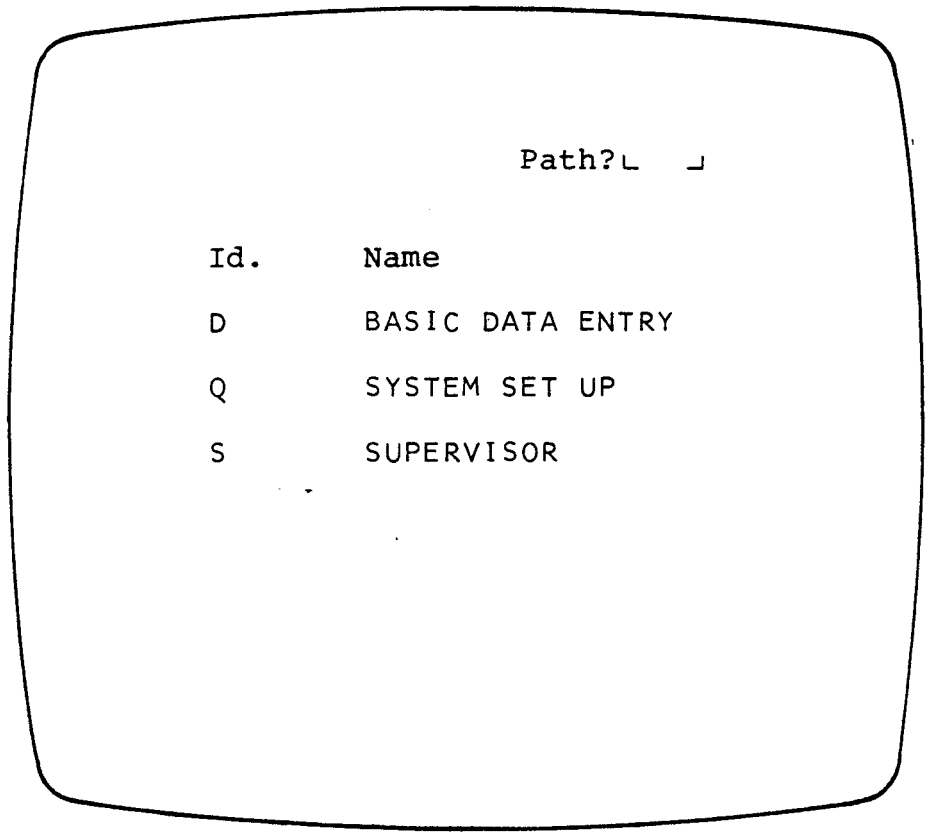


Figure 4.2 IDE path catalogue

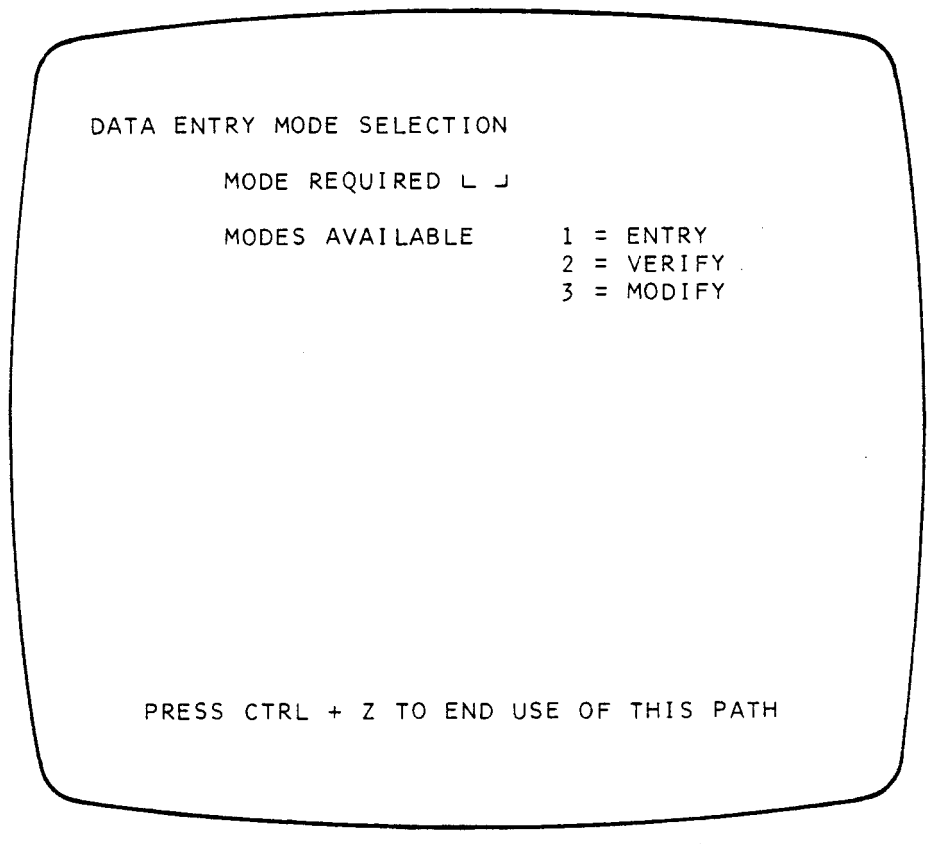


Figure 4.3 Mode selection screen

Complete the fields on the format as follows:

Field	*Content*
BATCHNAME	Any 12 alphanumeric character name including spaces but with at least one non-space character. This field is mandatory
DATE	This is not input. It is displayed on the screen by the system
PASSWORD	Any eight alphanumeric character password. If a password is input at this stage is will be required when any other attempt is made to access the batch. If this field is left blank no security checking is possible when the batch is accessed again
OPERATOR	Any eight alphanumeric character name. It should be the name of the operator entering the batch
NO OF DOCUMENTS	The number of documents in the batch. This field is mandatory, see section 2.6.5 and 4.3.2
LINE FACTOR	The number entered in this field reflects the number of records generated by each document. If a document always produces a fixed number of records this is the number input. If a document genreates a variable number of records, the number input should approximate to the number of records generated by the average document with the proviso that the number input should be at least half the maximum number of records generated, (see sections 2.6.5 and 4.3.2). If this field is left blank the system assumes that no records are generated for this batch so no records can be input and an empty batch is created
BATCH TOTALS 1 and 2	These can be blank or any number in the range 0 to 99999999.999. The batch totals can contain up to three decimal places. The decimal point should be input explicitly

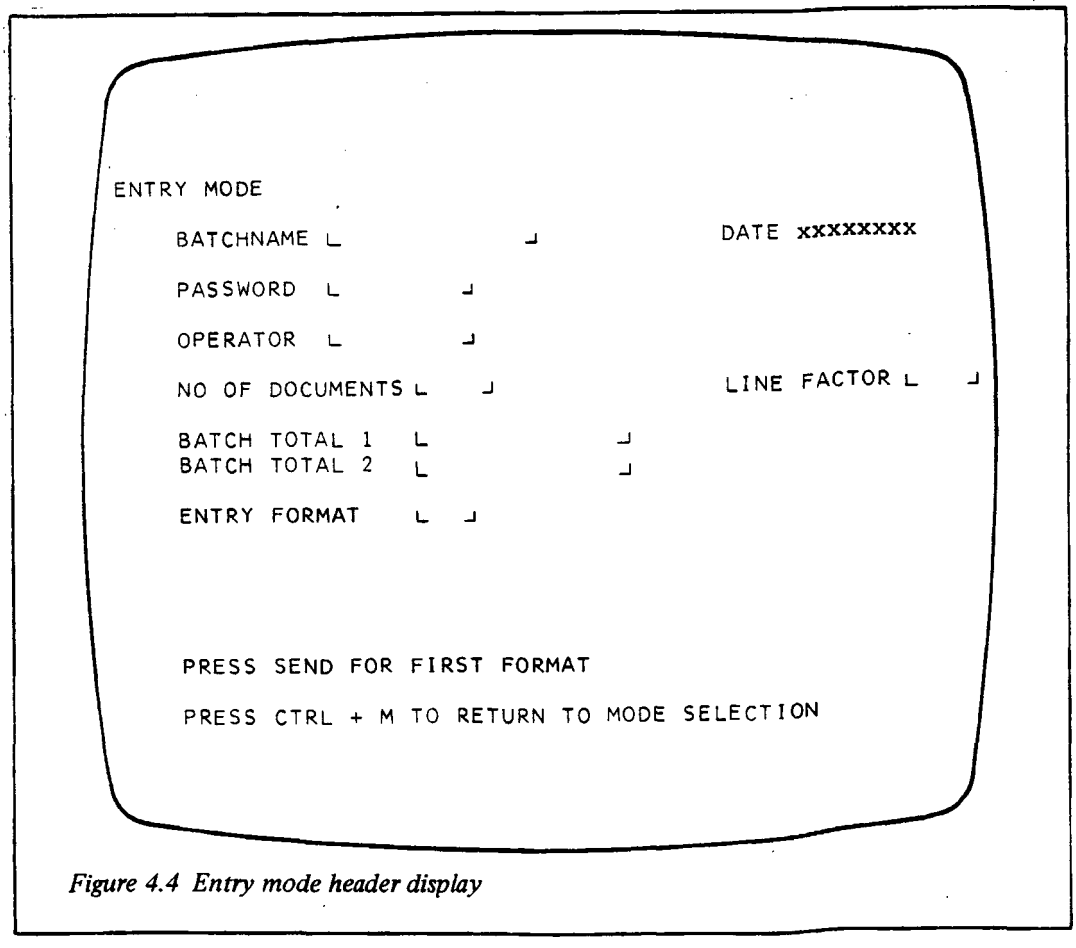


Figure 4.4 Entry mode header display

FORMAT NO	This field is mandatory as it calls the screen format into which the data will be entered. It should be a number greater than 15 since formats 1 to 15 are used by IDE

The following errors may occur on input to the entry mode header:

Message	*Meaning/operator action*
BATCHNAME IN USE	A batch currently in the system has the same name. The operator should either use another name for this batch or the supervisor should erase the existing batch
ENTER NO OF DOCUMENTS	The number of documents has been input as spaces or zero. Correct, press TAB and continue
ENTER BATCHNAME	The batchname field has been left blank. The operator should input a batchname and press TAB
INVALID FORMAT	The format number input is less than 16. This must be corrected and TAB pressed before continuing
NO SPACE ON ENT*n* (where *n* is in the range 1 to 4 and indicates the entry file number associated with that video)	There is insufficient space on the entry file for this batch to be written to diskette. There are three options: 1 Abandon entry at this video and retry at another video on the system 2 Abandon entry at this video and verify some batches at this video to free space on the entry file. Entry of this batch can then be attempted again 3 Reduce the number input to LINE FACTOR and press TAB
LOAD FILE ENT*n* (where *n* is in range 1 to 4)	The entry file for this video is not loaded. The supervisor should load the file and the operator press TAB
ERROR D	A screen format does not exist for this number. Correct the format number, press TAB and then SEND
ERROR G 08 XX	Consult supervisor
ERROR D 02 09 XX	Consult supervisor
ERROR D 02 10 XX	Consult supervisor
ERROR D 02 20 XX	Consult supervisor

When all the fields on the screen have been filled in successfully the operator should press SEND. The user screen format specified in FORMAT NO will be displayed for data input.

The action key M will abandon entry of this batch and display the mode selection screen if it is used before SEND is pressed.

4.3.2 Entering data

The data is entered into a user screen format designed to reflect the layout of the input.

Data is entered into the unprotected fields on the screen as follows:

1 Type the data into the first unprotected field

2 Press the TAB key to invoke field validation and to move the cursor on to the next input field

3 Repeat steps 1 and 2 until all the fields on the screen have been completed. Then press SEND to write the record to diskette and display the next empty screen format

Note: Always TAB through all the fields on a screen format before pressing SEND, even if no data is to be entered.

At the bottom of the screen the system will display the record number of the first record entered on this screen. Thus if the screen format contains only one record, the record number displayed will be incremented by one every time the empty screen is displayed; if the screen contains two records the count will be increased by two each time, etc.

A field in error will cause a message to be displayed on the bottom line of the screen immediately after the TAB key has been pressed for that field. To correct the error press the INTERLOCK key. This turns off the alarm and returns the video to type mode so that the error can be corrected. After correcting the error press the TAB key again.

Notes:

1 When an error is detected, if the error message disappears from the screen and/or the alarm continues after pressing INTERLOCK the following action should be taken:

   (a) Depress the INTERLOCK key repeatedly until the alarm ceases

   (b) Retype the field that starts at the current cursor position

   (c) Press TAB to revalidate this field and wait until the cursor moves to the next field or an error message appears before continuing

2 Always press TAB after typing in the last field on the screen before pressing SEND

This input procedure is continued until all the data for the batch has been entered and written to diskette.

The commands available in this mode are as follows:

1 CLOSE BATCH Press CTRL and C. This will close the batch and perform the batch total checking. The close batch screen shown in Figure 4.5 will be displayed

2 ABANDON The action key A (press CTRL and A) will cause the current batch to be abandoned. All references to this batch will be deleted from the system. The entry mode header screen will then be displayed

3 LOAD NEW FORMAT CTRL and F is used if the current empty format on the screen is not the one required to continue entering the data. In response to this action key a screen is displayed into which the number of the required format should be typed

The error messages that appear in this mode fall into three categories:

1 VALIDATION ERROR MESSAGES These will appear if an input field fails its validation checks. The messages are self-explanatory, for example:

   VALUE OUT OF RANGE

   NOT NUMERIC

   For a full list of the errors see Appendix 5.

   When a validation error occurs the field in error should be corrected and TAB pressed before continuing with the data input. Should it be impossible to correct the error the cursor can be moved from the field in error to the next input field using the arrowed control keys. Data input can then continue.

   Note: In the second case the field will be written to diskette in its incorrect form. If this facility is used the input document should be clearly marked so that the error can be corrected in modify mode. The system will not automatically display records that have fields in error; however, if the field is not changed on verification, the error message will be displayed again when TAB is pressed

2 BATCH ERROR/WARNING MESSAGES

Message	*Meaning/operator action*
BATCH NEARLY FULL	The number of records already input in this batch is approaching the total number allocated when the batch header was input. This message appears with the empty input format when the number of empty record slots remaining is less than twice the number input as LINE FACTOR in the batch header. The alarm will sound. Press the INTERLOCK key to switch it off and to continue input. If only a few records remain to be input it will probably be possible to fit them into this batch. However if there remain too many records to fit into this batch it may be closed and the remaining records input to a new batch.

If this latter action is taken the supervisor must be informed so that she can use the COPY facility to recombine the batches after transmission.

The COPY facility is described in section 5.2.8

NO ROOM FOR THIS DOCUMENT - CLOSE THIS BATCH	This occurs when the batch is so full that there is not enough space to write the current screen to the diskette. The batch must be closed and a new batch started with the data from the input screen which generated this error message. The supervisor must be informed so that the two batches can have the appropriate COPY information provided to recombine them after transmission

Note: If a document consists of several screens and this forced closure occurs, the contents of a single document may be split between batches. The data will be merged after transmission provided the correct COPY information is supplied (see section 5.2.8).

3 OTHER ERRORS

Message	*Meaning/operator action*
INVALID FORMAT	When using action key F to select a new format, a format number less than 16 has been input. The format number should be corrected before continuing
ERROR G 01 XX	Consult the supervisor
ERROR G 02 XX	Consult the supervisor
ERROR G 03 XX	Consult the supervisor
ERROR G 07 XX	Consult the supervisor

Note: If errors are displayed, at any time, in the format ERROR X XX XX XX the supervisor should be consulted

4.3.3 Closing a batch

When a batch in entry mode is closed using action key C the close batch screen is displayed as shown in Figure 4.5. This screen provides comparisons of the actual and expected totals for the number of documents in the batch and the batch totals. If there is a mismatch between actual and expected the fields will be displayed flashing.

If the totals agree SEND should be pressed to close the batch and return to the entry mode header screen.

If the totals do not agree the following actions are available:

1 Use action key E (CTRL and E) to obtain an empty screen format to add further data to the batch

2 Use action key C to close the batch in spite of the mismatch

4.4 **Using modify mode**

4.4.1 Completing the modify mode header

Having selected modify mode the header shown in Figure 4.6 will be displayed. Complete the fields as follows:

Field	*Content*
BATCHNAME	The 12 alphanumeric character name of the batch to be modified
PASSWORD	The eight character password used with this batch
OPERATOR	The eight character name of the operator modifying this batch

The errors that may occur when inputting to this screen are:

Message	*Meaning/operator action*
BATCH IN USE	Another video is using this batch. Check that the batch name has been input correctly. If not change the name and press TAB to continue. Otherwise this video cannot access the batch named until it has been released by the other video using it

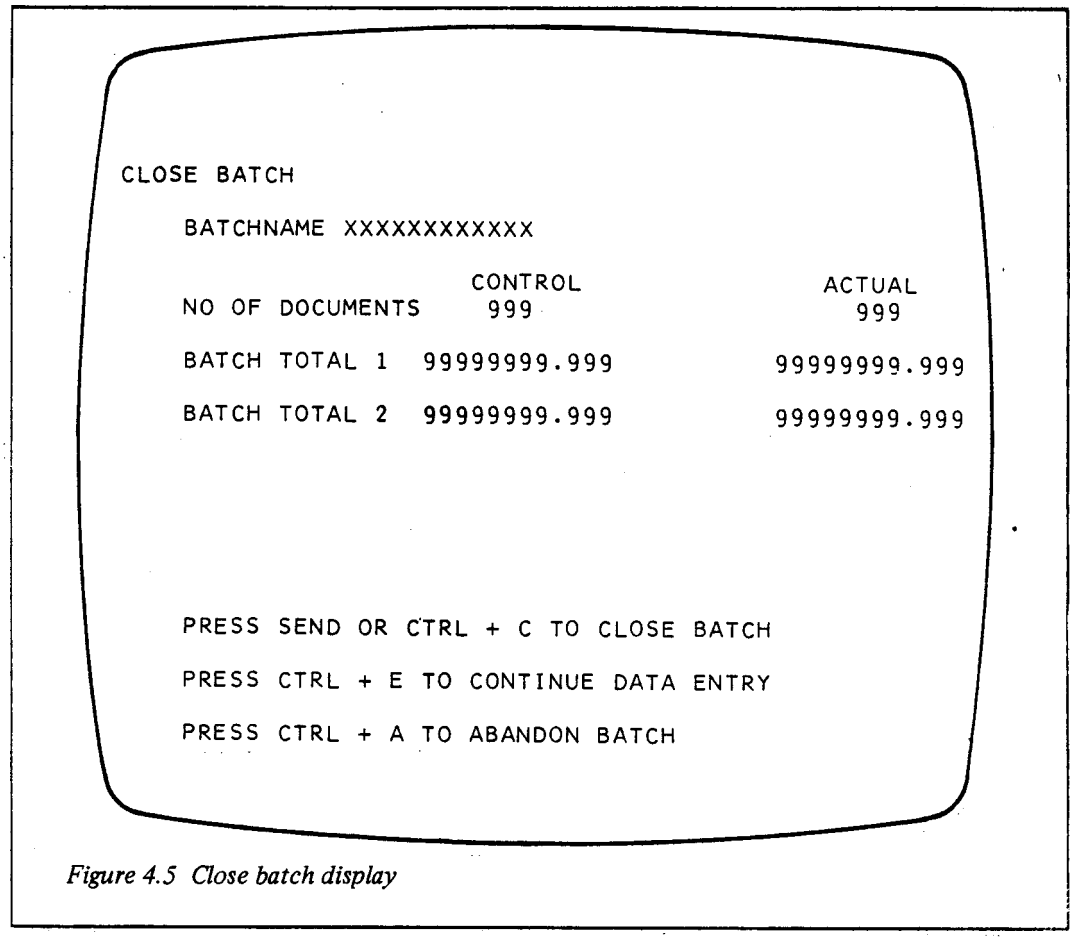


Figure 4.5 Close batch display

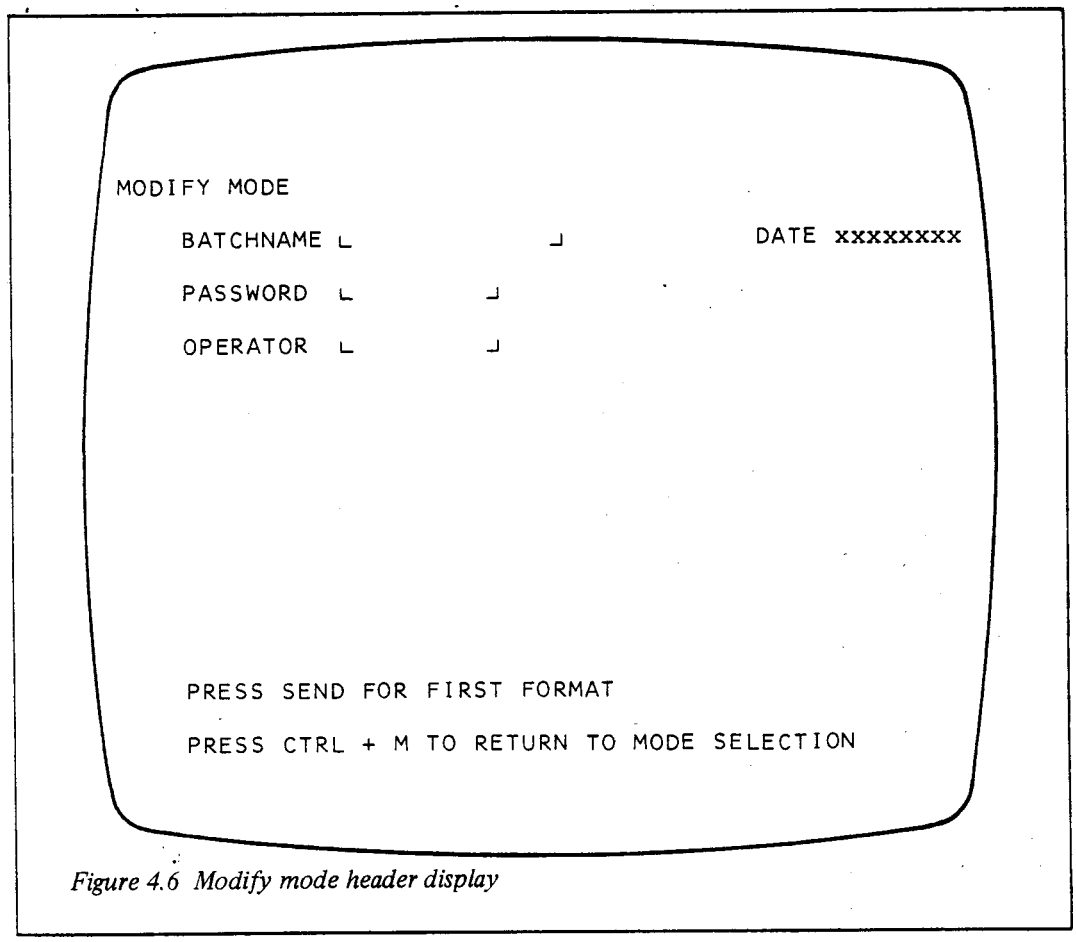


Figure 4.6 Modify mode header display

NO SUCH BATCH	No batch of this name has been entered. Check that the batch name is correct. If not change the name and press TAB to continue
INCORRECT BATCH STATUS	The batch is not available for modification as it has already been verified or transferred to the transmit file
INCORRECT PASSWORD	Correct the password and press TAB
ERROR D 04 04 XX	Consult the supervisor
ERROR D 04 05 XX	Consult the supervisor
ERROR G 08 XX	Consult the supervisor
LOAD FILE ENT*n* where *n* is in the range 1 to 4	The entry file containing this batch is not loaded. The supervisor should load the file. The operator should then press SEND to continue
ENTER BATCHNAME	The batchname has been omitted. Enter the batchname and press TAB

Having completed the screen correctly press SEND to obtain the first record in the batch. To abandon the modify mode header before pressing SEND the action key M should be used. This will redisplay the mode selection screen.

4.4.2 Modifying data

When a batch is opened in modify mode the records are presented screen by screen in the same order and format as they were input. All the records are displayed.

To select and modify the appropriate records proceed as follows:

1 Press CTRL and S to skip over records that are not to be modified

2 When the screen requiring changes is displayed use the cursor control keys (the arrowed keys on the furthermost right hand block) to position the cursor over the characters for alteration

3 Make the alterations

4 Press TAB after every altered field. This performs the valdiation checks on the field. Errors will be reported and should be corrected as described in section 4.3.2

5 Use action key D (press CTRL and D) to delete the information displayed on the screen.

Note: If the screen contains more than one record it is not possible to delete individual records

6 When all the necessary modifications have been made to a screen and the validation carried out, press SEND to write the record back to diskette. SEND should not be used after action key D

7 Press CTRL and C to close the batch when all modifications have been made. This will display the END OF FILE screen. This screen will also be displayed when the end of the batch is encountered during modification

8 Press SEND. This will cause the modify mode header screen to be displayed

4.5 Using verify mode

4.5.1 Completing the verify mode header

Having selected verify mode the header screen shown in Figure 4.7 will be displayed.

Complete the header as follows:

Field	*Content*
BATCHNAME	The 12 alphanumeric character name of the batch to be verified
PASSWORD	The eight character password of the batch to be verified
OPERATOR	The name of the operator verifying this batch

The following errors may occur on input to this screen:

Message	*Meaning/operator action*
BATCH IN USE	See section 4.4.1
NO SUCH BATCH	See section 4.4.1
INCORRECT BATCH STATUS	The batch is not available for verification. Consult the supervisor
INCORRECT PASSWORD	The opertor should correct the password and press TAB
TRANSMIT FILE FULL	There is no room for this batch in the transmit file. The batch cannot be verified. Consult the supervisor
LOAD FILE ENT*n*	See section 4.4.1
ENTER BATCHNAME	See section 4.4.1
ERROR D 03 04 XX	Consult the supervisor
ERROR D 03 05 XX	Consult the supervisor
ERROR G 08 XX	Consult the supervisor

When this screen has been correctly completed press SEND to obtain the first screen for verification.

At any time prior to pressing SEND action key M can be used to abandon this verification. This will return the video to the mode selection screen.

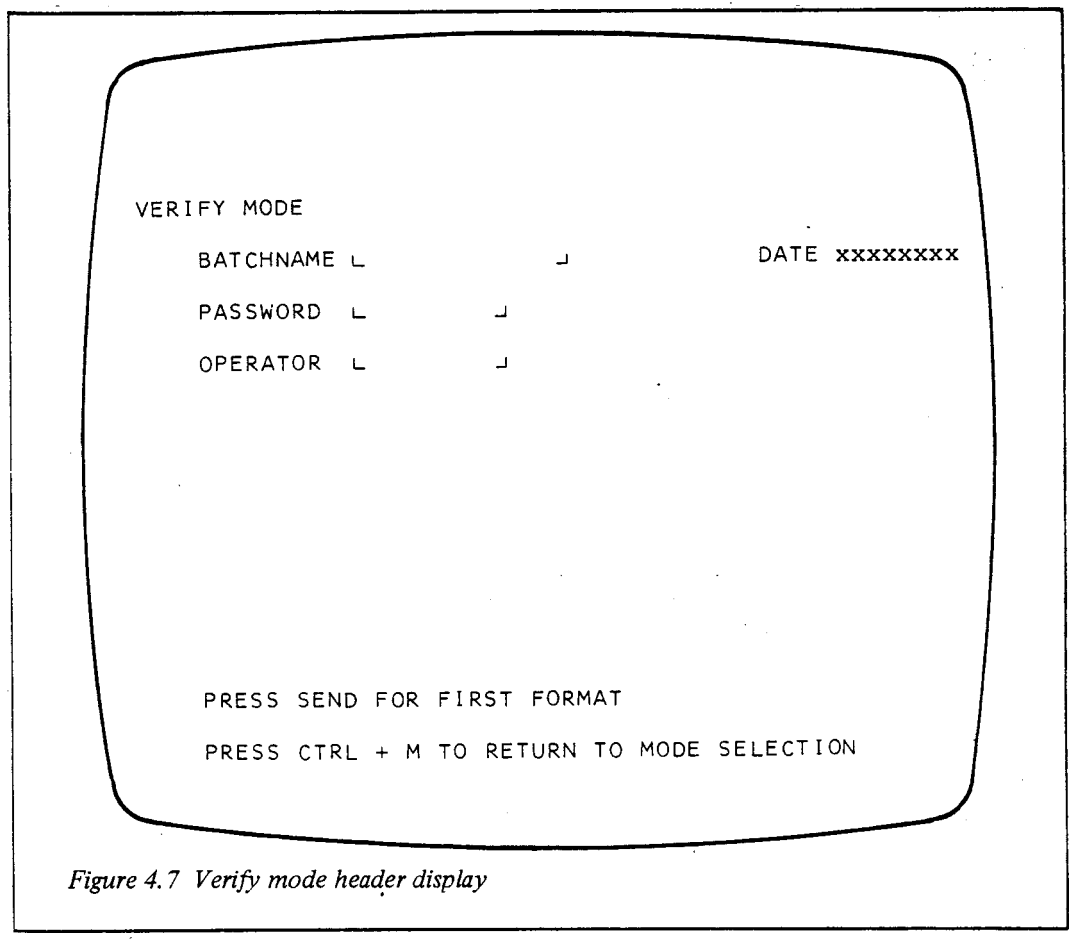


Figure 4.7 Verify mode header display

4.5.2 Verifying data

Records for verifitation are displayed in the same order and format as they were entered.

Each screen is presented with the fields that are to be verified blank. Proceed as follows:

1 Type the data into the blank unprotected fields on the screen. The data is compared on a character by character basis. As each character is typed, if it matches the character that was originally entered the character is displayed and the cursor moves on to the next character position

2 If the character input differs from the original, the original character is displayed, the keyboard locked and the alarm sounded. Press the INTERLOCK key to unlock the keyboard and switch off the alarm

3 Retype the original or verifying character whichever is correct. If neither is correct type the correct character. Whatever character is typed will be accepted by the system and the cursor moved on to the next character position

4 When all the characters for a particular field have been entered press the TAB key. The system will not accept a field unless all the non-space characters have been verified. If TAB is pressed before all the characters have been verified the system will repeat the error sequence from item 2

5 Pressing the TAB key initiates the field validation and moves the cursor on to the next field to be verified. Any validation errors reported should be corrected as described in section 4.3.2

6 When all the data on the screen has been verified press SEND. This will write the verified data to the transmit file and display the next record for verification

7 When the last record in the batch has been verified the END OF FILE screen will be displayed. Press SEND. This will cause the verify mode header screen to be displayed

The following errors may occur during verification:

Message	*Meaning/operation action*
ERROR G 01 XX	Consult the supervisor
ERROR G 05 XX	Consult the supervisor

Use of action key A will cause the current batch to be abandoned. All the verification done on the batch will be ignored and verification will have to start again from the beginning of the batch. Pressing CTRL and A will display the verify mode header screen.

Note: No attempt should be made to close the batch while in verify mode. Action key C, normally used to close a batch will cause the following message to appear:

BATCH CANNOT BE CLOSED IN VERIFY MODE

Verification should then continue on the screen displayed or action key A can be used to abandon the batch.

In general the supervisor responsibilities are:

1 Operating the 7501 or 7502 system

2 Maintaining the appropriate files on diskette using the terminal utilities

3 Setting up the IDE system

4 Controlling the data entry operation

5 Establishing the communication link with the mainframe to transmit verified data

This chapter deals in detail with controlling the data entry operation. Chapter 6 deals with the transmission of data and Appendices 1 and 2 cover the first three items.

In order to perform some of the supervisor functions one video, normally the lowest numbered, is made into a console. This is done by inserting a console PID, encoded 255, into the PID reader. When a video is being used as a console the other videos on the system cannot be used. When using the supervisor functions in the IDE supervisor path any video can be used and no PID is required.

When controlling the data entry operation supervisor commands can only be issued from a video that has entered the IDE supervisor's path. On entry to this path the system requires a password to be input. Having finished issuing supervisor commands the video can be returned to normal data entry by requesting the IDE path catalogue and selecting the data entry path.

5.1 Starting a session

Before starting the first data entry session the supervisor should have carried out the IDE set up procedure described in Appendix 2. This involves creating the files described in section 2.6.2.

Thereafter each session is started as set out in section 5.1.1.

5.1.1 Loading the IDE system

Before starting a session make sure that the following items are available:

1 A TE library diskette containing T2AP31. This is the terminal executive used to run IDE

2 A diskette containing the path library file IDEPATHS

3 A diskette(s) containing the entry and transmit files that will hold the data input.

Note: The transmit file TRANS01 and the COPY information file IDECNTRL must be empty before each data entry session. See section 6.7 for a description of how to empty these files

4 A diskette containing the other IDE system files (see Appendix 2)

5 A console PID for use during the setting up of T2AP31

To set up the terminal system for T2AP31 proceed as follows:

1 Switch on the processor

2 Switch on the lowest numbered video

3 Insert the TE library diskette containing T2AP31 into drive 1 of the dual floppy disc drive or drive 2 if drive 1 is inoperable. Select position 3 on the rotary switch of the processor

4 'Genres' then 'activate' using the appropriate switch on the processor. The diskette menu select display will then appear

5 (a) Use the blue numeric key(s) to enter the appropriate *TE No* for T2AP31

(b) Press the SEND key

The message 'T2AP31/*nnn*' will appear on the screen

where

nnn is the issue number of the TE

6 Remove the TE library diskette from the drive

7 Insert a console PID into the PID reader

8 This step is optional and can only be used before the first "GO" command is issued

(a) Press the LOCAL INPUT key

(b) Type CE *n*

where *n* is a numeric value in the range 1 to 4 and represents the number of videos as explained below

(c) Press the SEND key

Note: CE configures T2AP31 for use with up to four videos. If CE is not used then all the videos on the system will be configured for use with the TE. The maximum number of videos that can be configured is four. CE is intended for use with a value *n* less than the number of videos available. The purpose of this is to provide the videos configured with a greater proportion of the available TPL space

9 This step is optional and is used if a file creation date is to be written to the data file(s). This date is displayed only when the Display Index function described in Appendix 1 is used

(a) Press the LOCAL INPUT key

(b) Type SD *date*

where *date* is six alphanumeric characters, for example, 010779

(c) Press the SEND key. All empty files, opened during this session for writing, will have this date written to their file headers.

If the SD command is not used then spaces will be assumed

10 Insert the diskettes containing the IDE system files (see Appendix 2), the path library file, normally IDEPATHS, and the entry and transmit files into the diskette drives

11 Use the GO command as follows:

(a) Press the LOCAL INPUT key

(b) Type GO

(c) Press the SEND key

The following messages are displayed:

(a) USER LIBRARY IDEPATHS OPENED

This message is displayed on the console when the file is opened

(b) SYSTEM READY

This message is displayed on all other videos when the file is opened

(c) USER LIBRARY NOT AVAILABLE

This message is displayed on the console if IDEPATHS cannot be found

The terminal system will now be ready for path selection before off-line data entry use. The video used as a console during this setting-up can be made available for normal use with T2AP31 by removal of the console PID.

5.1.2 Start up procedures

Having loaded the IDE system and removed the console PID the start up procedure for the session should be continued as follows:

1 Select the path catalogue

(a) Press the LOCAL INPUT key

(b) Type PC

(c) Press the SEND key. If successful the path catalogue will be displayed on the screen. If not error A will be displayed indicating failure to load the path catalogue. This may be the result of an unsuccessful GO command. If so check the files are all available and repeat the GO sequence specified in section 5.1.1

2 (a) Select the supervisor path by typing S into the field marked *Path?*

(b) Press the SEND or TAB key. If the supervisor path is being used at another video the following error message will be displayed:

SUPERVISOR PATH IN USE PRESS CTRL+Z TO END USE OF THIS PATH

Otherwise the first format of the supervisor path will be displayed

3 (a) Input the supervisor password. (The initial password is defined when IDE is set up as described in Appendix 2)

(b) Press TAB. If the file SPACNTRL, which contains the current supervisor password, is not loaded the error message

LOAD FILE SPACNTRL

will be displayed. If the diskette containing this file has simply not been inserted into a drive this should be done before continuing. However if the file does not exist CTRL and Z should be pressed to end the use of this path so that the file SPACNTRL can be created. For instructions on creating IDE system files see Appendix 2

(c) If the password has been input incorrectly the message

INCORRECT PASSWORD

will be displayed. The password should be corrected and TAB pressed

(d) After correct input of the password the system replies with the supervisor function list display as shown in Figure 5.1

4 Type 1 into the field *Function Required* and press TAB. The system checks that the IDE system files are on-line. If not one or more of the following messages could occur:

LOAD FILE IDECNTRL

LOAD FILE TRANS01

LOAD FILE BATCHDET

LOAD FILE OPERATOR

In this case action key Z should be used to exit from the path. The appropriate files should then be loaded and the sequence recommenced from item 2 (for instructions on creating IDE system files see Appendix 2).

The system also checks that the transmit file TRANS01 and the file to contain the batch copy information IDECNTRL are empty. If they are not one or both of the following messages will be displayed:

TRANS01 NOT INITIALISED PRESS CTRL+Z TO END USE OF THIS PATH

IDECNTRL NOT INITIALISED PRESS CTRL+Z TO END USE OF THIS PATH

These errors indicate that the data in the transmit and/or copy information files was not erased at the end of the last transmission session. Action key Z should be used to exit from the path and empty files should then be created as specified in section 6.7.

If no errors are present the start of session display will appear as shown in Figure 5.2

Figure 5.1 Supervisor function list display

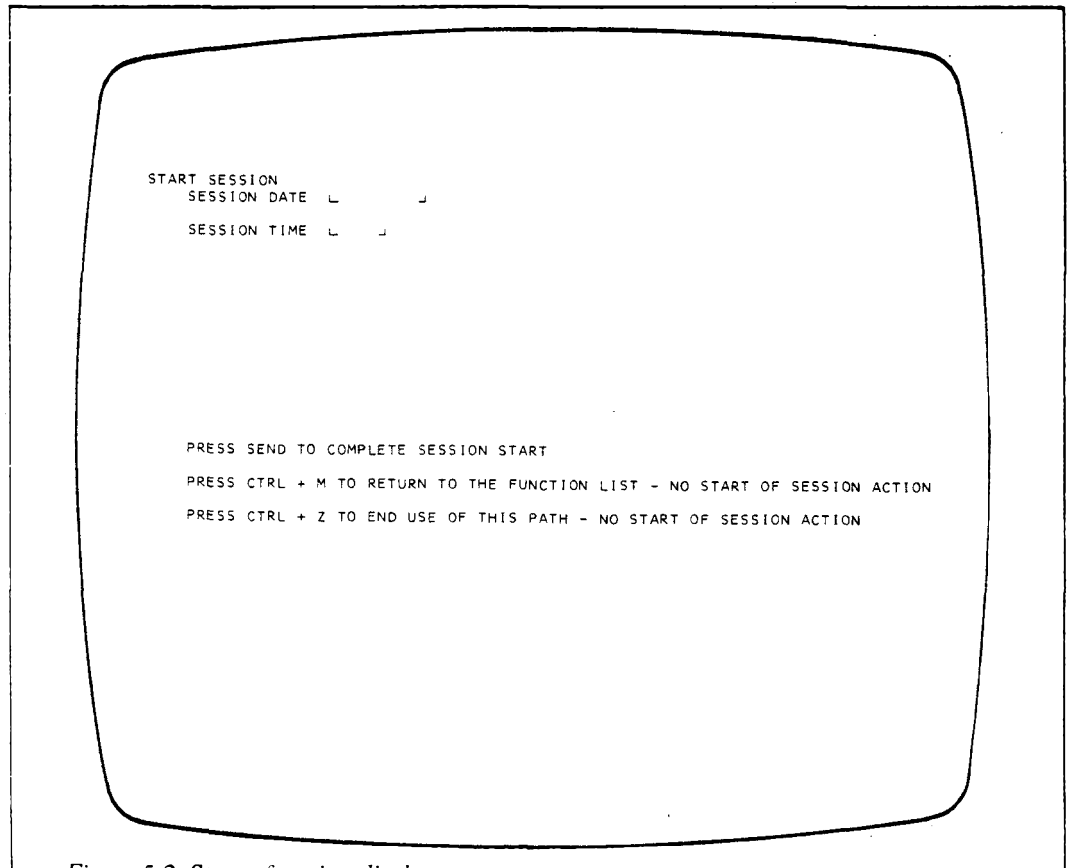


Figure 5.2 Start of session display

5 (a) Input the session date and press TAB. The date can be in the form DD/MM/YY or DDMMYY. If left blank the message

ENTER DATE

will be displayed; if invalid the message

INVALID DATE

will appear. The date should then be reinput and TAB pressed

(b) Input the time. The 24 hour clock notation should be used for the time, that is 0000 to 2359. If outside this range the message

INVALID TIME

will be diaplayed. The time should be corrected and TAB pressed

(c) Press SEND

Other error messages that could occur are:

Message	*Meaning/operation action*
ERROR S 01 01 XX	Consult ICL
INVALID FUNCTION	A function number in the range 1 to 18 has not been input, correct and continue
ERROR S 11 03 XX	Consult ICL
ERROR S 11 04 XX	Consult ICL
ERROR S 11 08 XX	Consult ICL

A successful start of session returns the supervisor to the supervisor function list display. Data entry can now commence on all videos in the system. Chapter 4 describes the data entry phase.

To use the supervisor's video for data entry type CTRL and Z to display the IDE path catalogue. Continue as described in Chapter 4.

5.2 Controlling the data entry session

In order to allow the supervisor to control the data entry session the IDE system provides various supervisor functions.

All the functions available are on the supervisor function list display as shown in Figure 5.1. This is the first screen displayed after successfully entering the supervisor path.

The steps required to display the supervisor function list are:

1 Obtain the IDE path catalogue. In order to check on the method of obtaining the path catalogue the supervisor should consult the parts of this publication dealing with the current operations of the video

2 Type S to obtain the supervisor path

3 Input the supervisor password. The supervisor function list will then be displayed

The individual functions are entered by typing the appropriate function number on the screen. An error message

INVALID FUNCTION

will occur if a function number outside the range 1 to 18 is input. The message

START SESSION

will occur when attempting to call all functions except 1 and 3 if the data entry session has not been started. See section 5.1.2 for the start of session routine.

Having completed a function the supervisor function list can be redisplayed by pressing CTRL and M. The IDE path catalogue can be redisplayed by pressing CTRL and Z.

Most functions can be aborted before completion by using the action keys M and Z before pressing SEND.

5.2.1 Function 2 NEW TRANSMIT FILE

This routine is only available with terminal systems that have four diskettes (two dual floppy disc drives) and where the transmit file occupies a single diskette (see Appendix 2 for details of setting up the transmit file). With these systems it is possible to continue with a data entry session after the original transmit file has been filled. This is done by changing the transmit file diskette for another diskette containing an empty transmit file. The diskette containing the new transmit file must have already been set up with the file name TRANS^{n}+1 where ^{n} is the sequence number of the current file (see Appendix 2 for details).

Before this function is called the supervisor must ensure that no verification is in progress and no verification must start until this procedure has been completed.

To use this function the supervisor should proceed as follows:

1 Remove the diskette containing the current transmit file and replace with the new diskette

2 Select the function by typing 2. The system will check that an empty transmit file has been provided. The system will be initialised to use this file and the message INITIALISATION COMPLETE will be displayed

3 Use action key M or Z to exit from the function. In this case the use of these action keys does not abort the function

Errors that may occur relate to the transmit file and indicate that the file has not been loaded, that it is not empty or that all the nine transmit files have been used, nine being the maximum number available. The error messages are specified in Appendix 5.

5.2.2 Function 3 RESTART

In the event of a machine failure or should it be necessary to reload the terminal executive after a session has been started this function must be used.

Note: This must be the first action in these circumstances. Selecting this function causes the necessary tidying up of the IDE system to take place.

The supervisor is not required to input any data to this function. When all the necessary tidying has been done (this may take some time) the message, RESTART COMPLETE, is displayed. Action keys M or Z can then be used to exit from the function. In this case the action keys do not abort the function.

Error messages will be displayed if any of the system files are not present. See Appendix 5 for all the error messages.

The message: USE START OF SESSION will be displayed when this function is called if the machine failure occurred before the session has been started. The message ERASE FILES AND USE START OF SESSION will be displayed when this function is called if the machine failure occurred during the session start routine.

5.2.3 Function 4 BATCH STATUS DETAILS

When this function is called the screen shown in Figure 5.3 is displayed. Enter the batchname and press TAB. The system will reply with all the batch details except the password. This can be displayed by using action key P. Appendix 5 lists the errors.

5.2.4 Function 5 SELECTIVE BATCH LIST

This function lists selected batches. The criteria for selection of the batches are:

1 BATCH STATUS That is D for data entered, V for data verified , C for data transmitted, and I for data that is in the process of being verified

2 VT NUMBER This is the number of the video that entered the batch

The criteria for selection of the batches are typed into the screen shown in Figure 5.4. Only those batches fulfilling the criteria input are listed. It is not necessary to specify both criteria and if neither is entered all the batches in the system will be listed.

The information supplied on the batch list screen is: batchname, status and the file it is being held on (either ENT*n* for an entry file, *n* being the number of the video to which the entry file relates, or T for the transmit file). One screen will list 22 batches; pressing SEND causes any continuation screens to be displayed

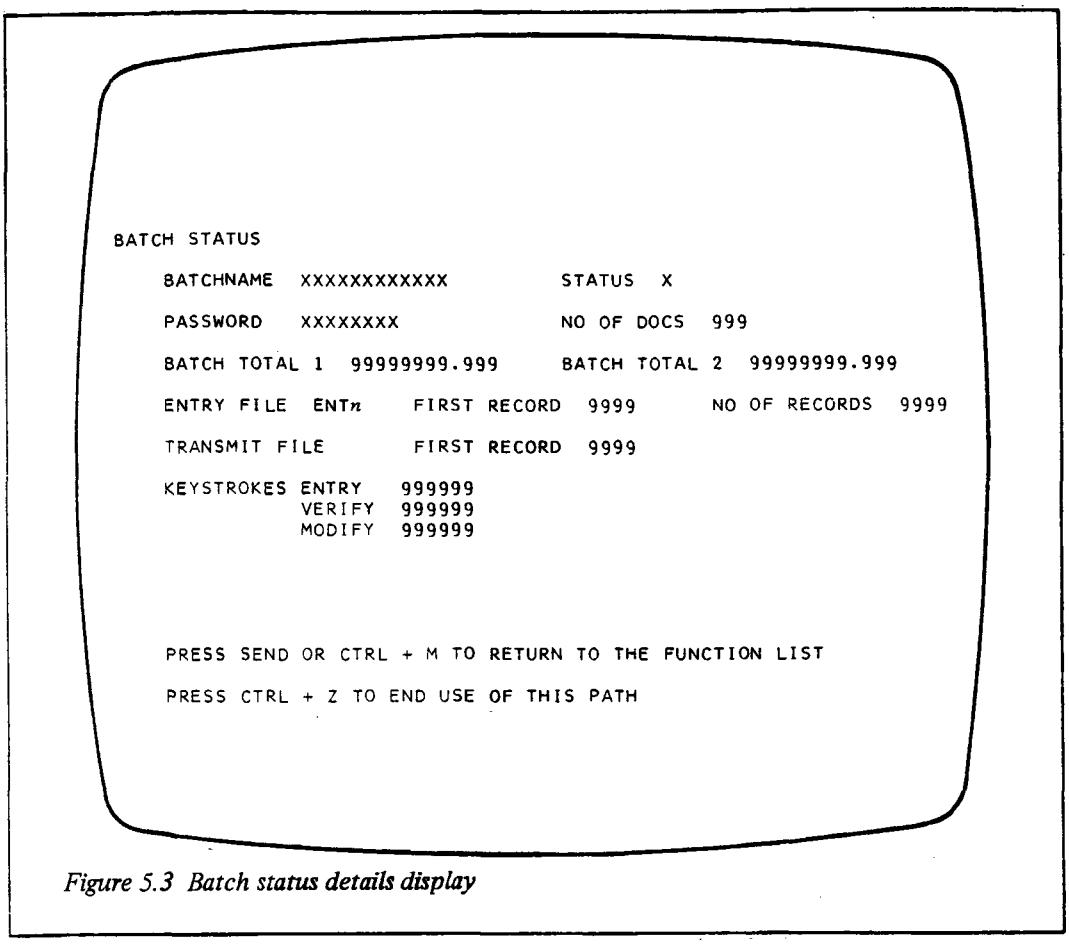


Figure 5.3 Batch status details display

5.2.5 Function 6 ERASE BATCH

This function erases all the IDE system references to a particular batch. It should only be used when it is certain that the batch is no longer required. Normally batch references will only be erased after the batch has been transmitted and successfully processed at the mainframe; that is when even the security copy of the batch is no longer required. Once batch references have been erased IDE can not access that batch again even from a security copy (unless the IDE system files relating to that batch have also been security copied, see section 6.7). It is recommended that batch references are retained in the system for a full working week after the batches have been transmitted. See section 6.7 for the recommended use of this function in relation to transmitting batches.

This function can also be used to erase batches that have not been verified or transmitted. Erasing the batch references for batches that have not been verified creates empty space on the entry file which can be used again.

The function is used as follows:

1 Enter the batchname and press TAB

2 Enter the batch password and press TAB. The current status of the batch will be displayed

3 If this status is COPIED (C), which indicates that the batch has been transmitted to the mainframe, press CTRL and E to erase the batch

4 If the status is not C the status field will be set flashing. Use CTRL and E to erase the batch

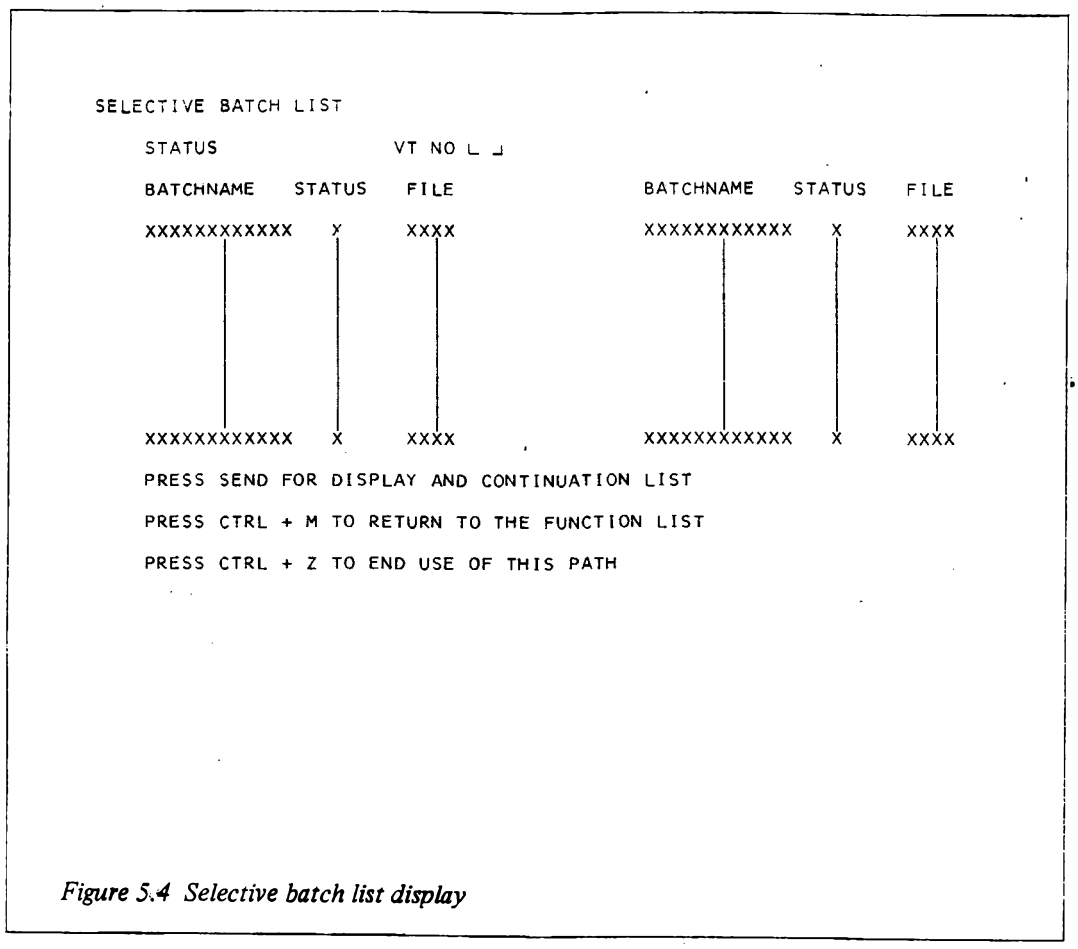


Figure 5.4 Selective batch list display

5.2.6 Function 7 BATCH LIST

When this function is called the system displays a list of all the batches in the system giving the batch number, name and status. Twenty-four batches can be listed on one screen.

Pressing SEND causes further screens to be displayed until all the batches in the system have been listed.

5.2.7 Function 8 DISPLAY/ERASE COPY

This function displays the COPY information for a batch; that is the details of the mainframe file to which this batch is to be written. It will also optionally erase this COPY information.

On entering the batchname the COPY information is displayed as shown in Figure 5.5. Action key E will delete these details but will not free that space for the input of additional COPY information, see section 5.2.8. Alternative COPY information should be supplied before transmission if there is sufficient space on the COPY information file IDECNTRL. Pressing SEND will display the COPY information of the next batch in the sequence.

5.2.8 Function 9 COPY

This function should be used for every batch before it is transmitted. The function provides the means of specifying the details of the mainframe file into which a batch is to be written. Batches can be writtn to a *document* on the mainframe spoolfile or to a named user file. Batches specified for the same destination are effectively merged when they are written to the file.

Batches that are destined for the same user file or spoolfile document are written to these files in the order in which the COPY information is supplied. The COPY information for all the batches to go to one document or user file need not be supplied at the same time. The COPY details can be interspersed with COPY details of batches going to other destinations.

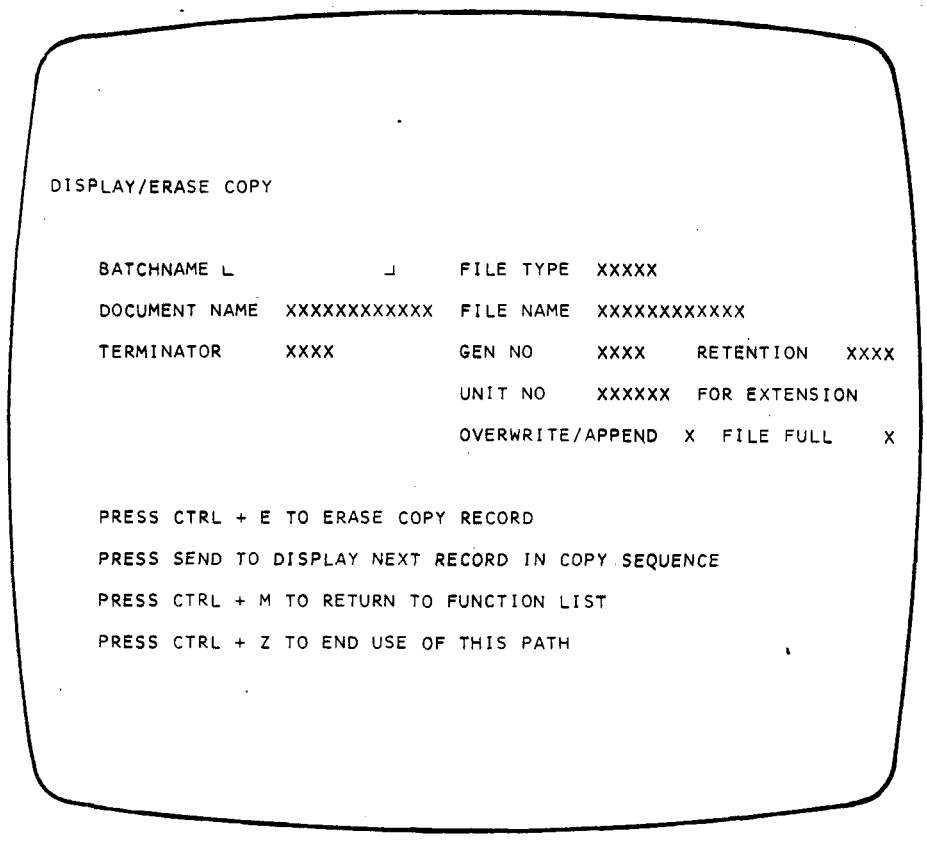


Figure 5.5 Display/erase copy display

Only 99 batches can have COPY information supplied for one session. This restriction applies even if some batches have had this COPY information erased. If batches cannot have their COPY information supplied at the terminal because the 99 limit has been reached or because the file details are not known, these details can be supplied by the mainframe operator as described in section 6.8.

On entering the COPY function the system will display the screen shown in Figure 5.6. This should be completed as shown below.

Where more than one batch is to be transmitted to the same user file or the same spoolfile document, only one COPY screen need be completed if the batches in the group to be transmitted have similar names. A group COPY is achieved by typing in the group copy indicator and specifying a partial batchname. On pressing TAB the system will display the full batchname of the first batch found in the group. If this batch is not the first of the group to be written to the destination file, action key S must be used to skip that batch. The system will then display the next batch found in the group. When the required batchname is displayed the COPY details are completed as normal. Pressing SEND causes these details to be written to diskette for the batch whose batchname is being displayed and causes the next group batchname found to be displayed. CTRL and S can be used to skip any batches so that the group COPY information is supplied in the order in which the batches are to be written to the destination file. After the last batch in the group is found the next batchname displayed will be the first in the group that has not had COPY information written to diskette. The system will continue to loop around the batches in the group until either all the batches have had COPY information supplied or action keys M or Z are used to exit from the function. The COPY information for a group need not all be supplied at the same time but must be supplied in the order in which the batches are to be written to the destination file.

Example

BATCHNAME input as INV

The batches copied could be INV01, INV02, INVAB, INVOICE, etc.

Note: Only batches starting with the partial batchname will be copied.

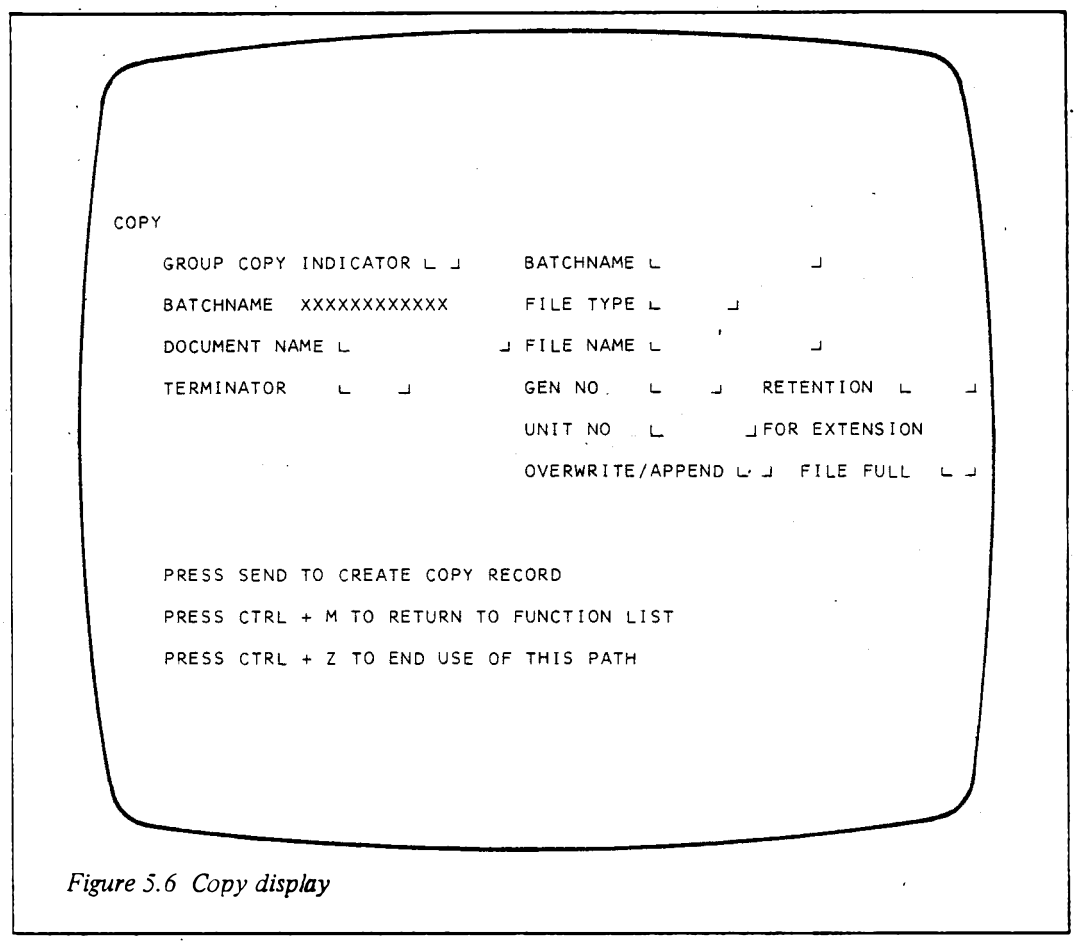


Figure 5.6 Copy display

The COPY screen should be completed as follows:

Field	*Content*
GROUP COPY INDICATOR	Type G if a group copy is required otherwise leave this blank
BATCHNAME	Enter the batchname of the batch to be copied or the partial batchname of the group of batches to be copied
FILE TYPE	The possible values for this field are SPOOL or USER. SPOOL indicates that the verified batch is to be transmitted to the spoolfile at the mainframe; USER indicates that the batch is to be written to a mainframe user file
DOCUMENT NAME	The name by which the batch will be known at the mainframe. This should only be completed for batches whose FILE TYPE is SPOOL and is mandatory for these batches
TERMINATOR	If a batch terminator other than **** is required this should be entered here. This may only be entered for batches whose FILE TYPE is SPOOL
FILENAME	Any user file name up to 12 characters in length. This field is mandatory for batches whose FILE TYPE is USER and specifies the name of the user file into which the batch(es) will be copied
GEN NO	This can be a number between 1 and 4095. It is the generation number of the user file to which the batch is to be copied. If this field is left blank the batch will be copied to the highest generation of FILENAME on-line
RETENTION	This should be input as a number in the range 0 to 4095. If it is omitted a retention period of 0 is assumed for the user file specified in FILENAME

UNIT NO FOR EXTENSION	Input the CSN of the disc to be used to extend the file. This will be used only if the user file needs extending to accept the batch and there is no room on the disc holding the last used file area. If this field is left blank extension of the user file will only be attempted on the current disc
OVERWRITE/APPEND	Enter the letter O if the batch is to overwrite the data in the user file. If more than one batch is to be copied to a user file, only the first overwrites, subsequent batches are appended to this first batch. Enter A if the batch is to be appended to the data in the user file
FILE FULL	Enter E if the user file may be extended when the batch is copied. In this case the user file will be extended in units of 96 blocks if necessary. Extension will take place initially on the current disc and then on the alternative disc specified in UNIT NO FOR EXTENSION. Enter A if the user file is not to be extended. In this case the copy will be abandoned if the batch will not fit into the existing user file and the copied data will be lost from the system completely. It can then only be retransmitted into a a larger file from the security copy of the transmit file after specifying new COPY commands. If the field is left blank E is assumed

Pressing SEND causes the screen to be written to diskette. A blank screen is redisplayed, except in the case of a group COPY (see above). Action keys M or Z can be used to exit from the function as described in section 5.2. If used before pressing SEND the current screen will not be written to diskette.

The errors that may occur during input to this function are as follows:

Message	*Meaning/supervisor action*
COPY DATA PRESENT	No additional COPY information for this batch can be supplied in this function
BATCH IN USE	The batch is being used. COPY information can not be supplied until the use of the batch has finished
IDECNTRL FILE FULL	COPY information has been supplied for 99 batches. No additional COPY information can be supplied
INVALID DATA	
NO SUCH BATCH	Batchname incorrect; reinput correct name and continue
ENTER SPOOL OR USER	The field FILE TYPE has been completed incorrectly. File type should be entered
ERROR S 19 12 XX	Consult ICL
ERROR S 19 13 XX	Consult ICL
ERROR S 19 14 XX	Consult ICL
ERROR S 19 15 XX	Consult ICL

5.2.9 Function 10 CHECK COPY

When this function is called the IDE system checks that each batch on the transmit file has had COPY information supplied. For any batch that has not, the system displays the screen shown in Figure 5.7 into which the COPY information should be typed. The screen should be completed as described for the COPY function in section 5.2.8.

When the CHECK COPY function has checked that all the information is present or the IDECNTRL file is full (99 batches having had COPY information supplied) a screen is displayed which gives the batchname and destination file type (S for spool and U for user) for all the batches with COPY information present. The batches are displayed in the order in which they will be written to the files. This screen should be used to check that the COPY information has been supplied in the correct order for batches that are to be written to the same user file

or spoolfile document. If the order is incorrect the appropriate COPY information can be erased using Function 8 described in section 5.2.7 and new COPY information can be supplied in the correct order using Function 9 described in section 5.2.8.

Note: Only 99 sets of COPY information can be input in one session. Deleting erroneous COPY information does not reduce the set count.

5.2.10 Function 11 CHANGE SUPERVISOR PASSWORD

Calling this function causes a display to appear containing the current password as shown in Figure 5.8. The new password of any eight characters should be input to this screen and TAB pressed. The system responds with the message:

PASSWORD TO BE XXXXXXXX

Pressing SEND causes the new password to be accepted and the system displays the message:

PASSWORD IS XXXXXXXX

Action keys M or Z can be used to abort the function (if used before pressing SEND) or to exit after acceptance of the new password. Action key M will display the supervisor function list. Action key Z will display the IDE path catalogue.

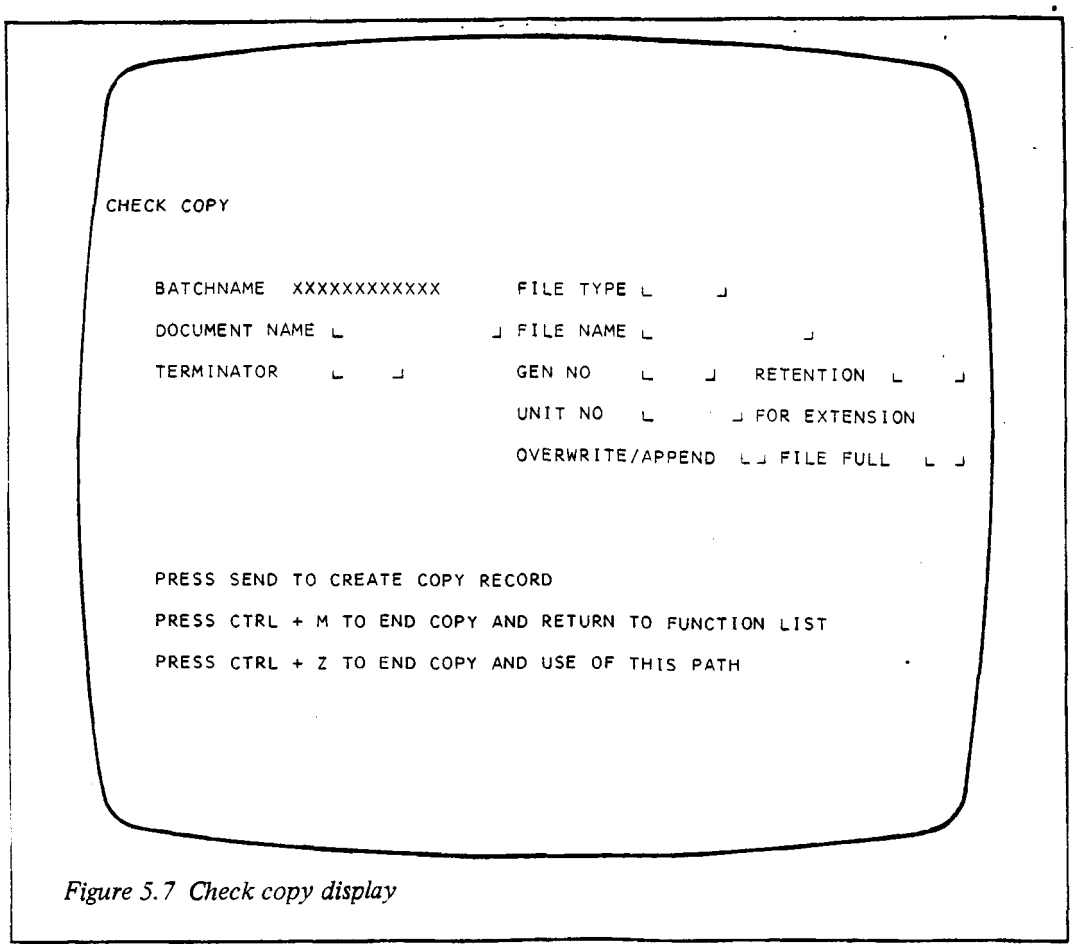


Figure 5.7 Check copy display

5.2.11 Function 12 CHANGE BATCHNAME/PASSWORD

This change is only possible for a batch that has been fully entered but not yet verified.

It is not necessary to change the batchname and password together. Either may be changed as follows:

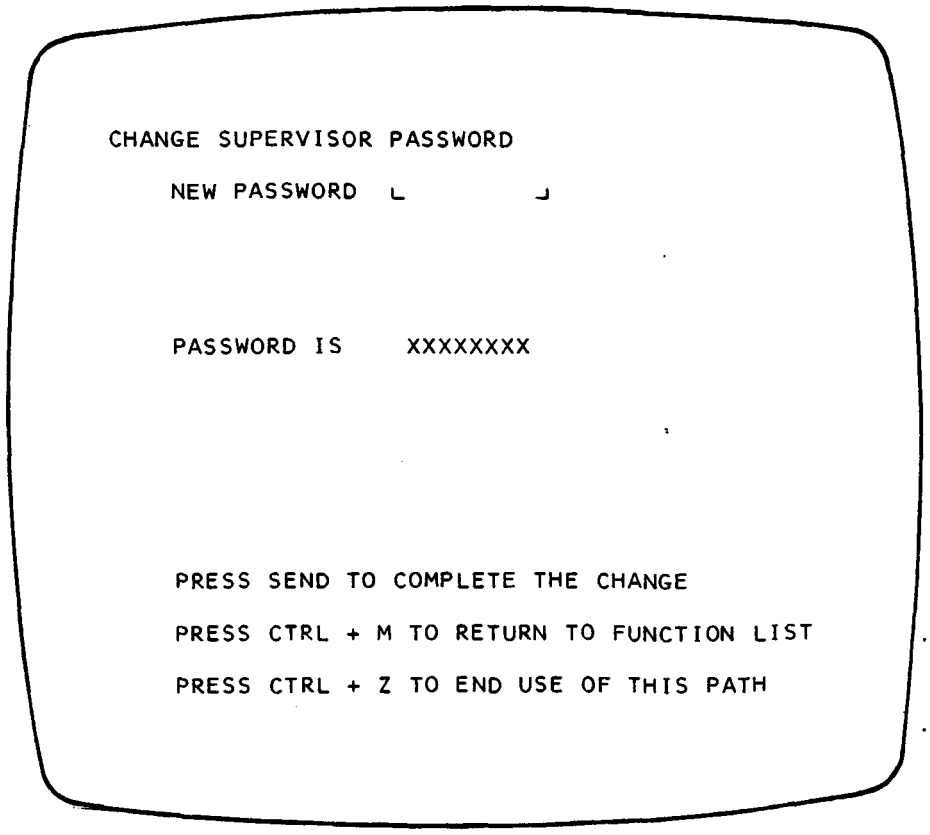


Figure 5.8 Change supervisor password display

1 Input the existing name of the batch and the current password

2 Input the new batchname and/or password into the fields NEW NAME and NEW PASSWORD

3 Press SEND

The system replies with the accepted new batchname and/or password. Action keys M or Z are then used to exit from this function (see Appendix 3). Appendix 5 lists the errors.

5.2.12 Function 13 COPY BATCH FROM TRANSMIT FILE

This function can only be used by a terminal system with two dual floppy disc drives as it requires two transmit files on separate diskettes and other IDE system files on a third diskette all to be loaded in the drives together.

The function is used to copy batches from the security copy of the transmit file back into the current transmit file so that these batches can be retransmitted. It is used when data has been lost or corrupted at the mainframe. Section 6.7 describes the method of creating security copies.

This function can only be employed for batches whose references have not been erased (see section 5.2.5).

Before calling this function the diskette containing the appropriate security copy must be placed into one of the drives. The current transmit file and IDE system files should remain on-line. The data entry files may be removed providing the videos for those files are not being used to enter data and that no data is being verified which was entered at those videos.

The function is called in the usual way from the supervisor function list screen. The system then displays the screen shown in Figure 5.9 into which should be typed the batchname and password of the batch required and the name of the security copy file containing that batch.

When the batch is found it is copied to the current transmit file. While the copy is underway the supervisor's video cannot be used for any other work.

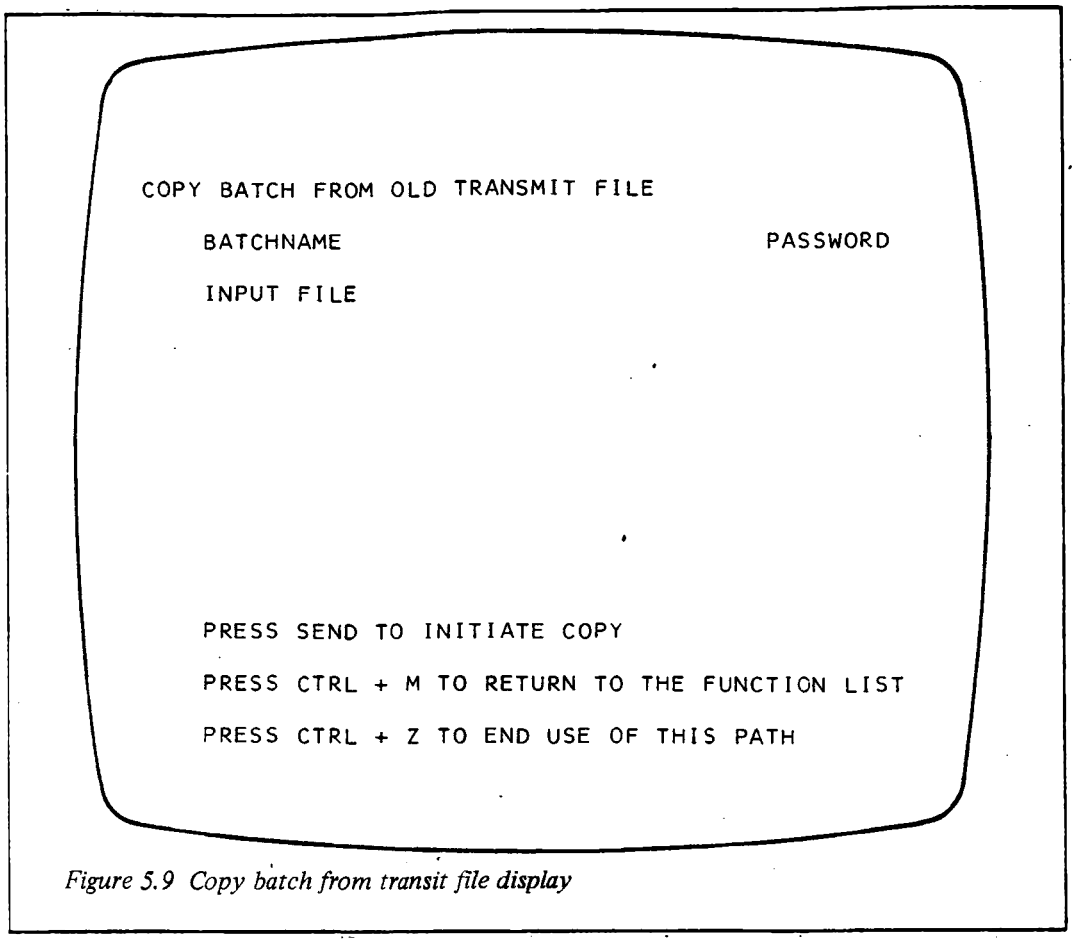


Figure 5.9 Copy batch from transit file display

Having successfully copied the batch to the current transmit file the system will display: *Batchname* COPIED. To redisplay the supervisor function list use action key M.

After the copy is complete and before the next transmission session COPY information must be supplied for this batch as described in section 5.2.8.

The errors that may occur during entry to this function are:

Message	*Meaning/operator action*
NO SUCH BATCH	A batch does not exist with that name in the IDE system. Reinput the correct name and continue
INCORRECT BATCH STATUS	The batchname input belongs to a batch that has not yet been transmitted. Change the batchname and continue
INCORRECT PASSWORD	The password is incorrect. Correct this and continue
INPUT FILE NOT ONLINE	The security copy is not loaded. Load the security copy and continue
BATCH NOT ON INPUT FILE	The batch named is not on the security copy loaded
TRANSMIT FILE FULL	There is not sufficient space on the transmit file to take this batch. The copy will have to wait until the transmit file has been emptied after transmission, or a new transmit file should be loaded
ENTER BATCHNAME	The batchname has not been entered. Enter this and continue
ERROR S 23 04 XX	Consult ICL
ERROR S 23 03 XX	Consult ICL
ERROR S 23 05 XX	Consult ICL

5.2.13 Function 14 CREATE/DELETE OPERATOR STATS

This function allows the supervisor to introduce new operators to the system by name so that statistics on keystrokes can be collected. It also allows the supervisor to delete operators from the system.

A maximum of 12 operators can be specified to IDE at any one time.

The function is completed by typing in the operator's name and the ACTION required. This can be either C for create or D for delete.

The following error messages may be displayed on input to this function:

Message	*Meaning/supervisor action*
NO SPACE ON FILE	A new operator cannot be added as 12 are already known to the system
OPERTOR ALREADY EXISTS	A record for this operator name already exists
NO SUCH OPERATOR	When deleting an operator name no record for this operator has been found
OPERATOR IN USE	An operator name cannot be deleted while the system is updating that operators record
INVALID ACTION	An incorrect action code has been input. It must be corrected before continuing
ERROR S 24 03 XX	Consult ICL
ERROR S 24 02 XX	Consult ICL

5.2.14 Function 15 OPERATOR STATISTICS DISPLAY

This function causes the system to display the keystroke statistics collected for the operator whose name is input. If a number in the range 1 to 4 is input the system will supply the statistics for that numbered video.

5.2.15 Function 16 TRANSMIT/ZEROISE OPERATOR STATS

This function is used by the supervisor to transmit all the operator statistics information to the mainframe and/or to zeroise the operator statistics held at the terminal.

A two character code is input on the screen shown in Figure 5.10 to specify the action required:

1 TO Input to transmit the records only

2 ZO Input to zeroise the records only

3 TZ Input to transmit and zeroise the records

As soon as this function is called, if the statistics are to be transmitted they are written to a record on the transmit file; if they are to be zeroised they are cleared. These actions are performed at the time that this function is called.

If the function is called to transmit the records and the transmit file is full an error message will be displayed and the function will not be performed. Appendix 5 lists the error messages.

5.2.16 Function 17 OPERATOR LIST

Entering this function causes the system to display a list of all the operators known to the system.

5.2.17 Function 18 SYSTEM STATUS

This function displays the free space in each file in the system. This can be used to determine when it will be necessary to transmit data to the mainframe.

The information displayed is shown in Figure 5.11.

The file sizes are specified as the number of records.

5.3 Error conditions

Certain error messages not detailed in section 5.2 will be displayed during operation of the supervisor path. In general these are self-explanatory but for a complete list of error messages see Appendix 5.

Other error messages may occur at any stage during the terminal operation. These messages will be in the form:

ERROR X 99 99 XX

In general these are irrecoverable and will require the supervisor to use the RESTART function. Appendix 5 describes these errors in more detail.

Errors that occur during the loading of any terminal executive can normally be overcome by repeating the procedure exactly as specified in this publication. For a full list of these errors, which are in the form ERROR XX, see *7501/7502 Software Operating*.

5.4 Closing a data entry session

There is no close down procedure necessary. Once data entry, modification and verification have ceased the transmission session can start. Transmission is detailed in Chapter 6.

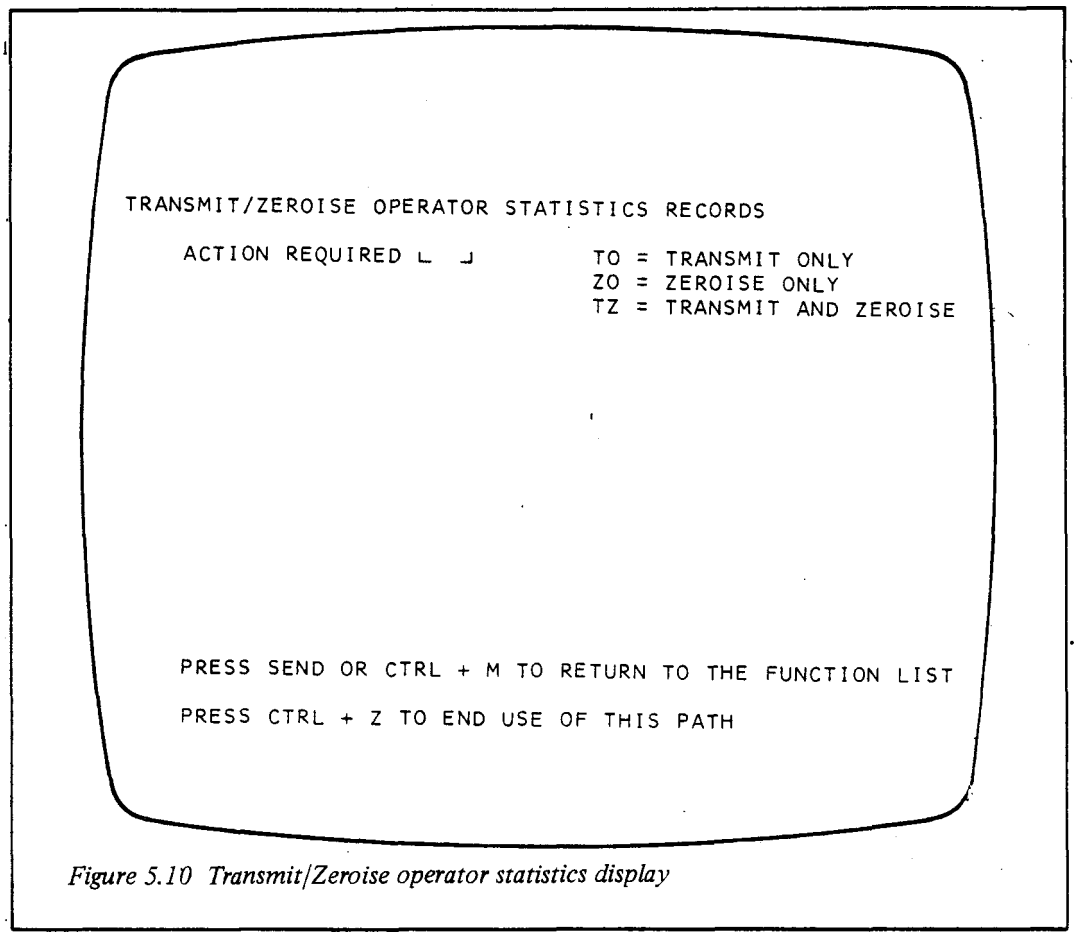


Figure 5.10 Transmit/Zeroise operator statistics display

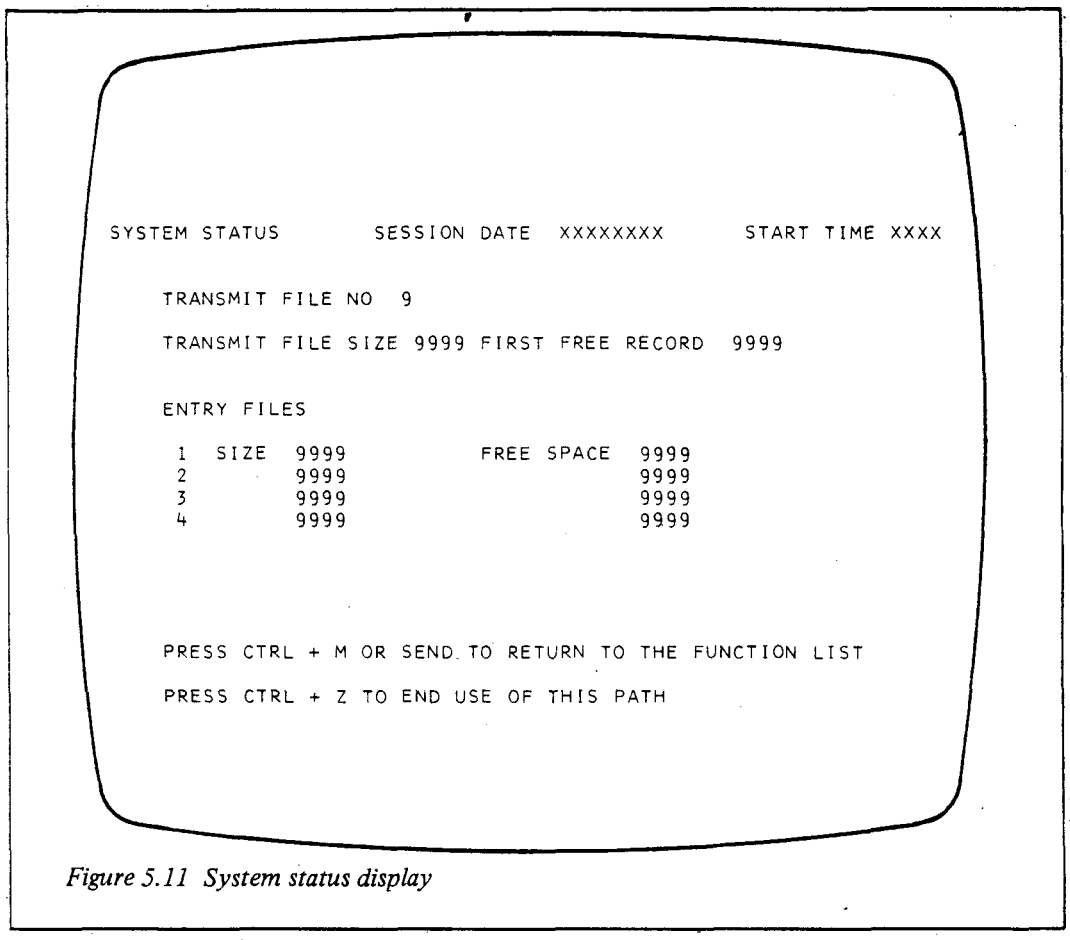


Figure 5.11 System status display

Data will usually be transmitted on a regular basis, normally at the end of the day. However each installation will have to judge the best time for transmission and this will be dependent on the size of the transmit file or files and the mainframe workload.

Before starting a transmission session all data entry, modification and verification must cease. Also all the verified batches must have had COPY information supplied so that the system can write these batches to the correct mainframe spoolfile document or user file. This is done using the supervisor function 9 described in section 5.2.8.

In order to set up a transmission session the following tasks must be completed at the terminal:

1 Establish a communications link between the mainframe and the terminal

2 Ensure that the program #XJXV is running in the mainframe

3 Load the TE T2RB31. This is the terminal executive used for transmitting data. It does not support data entry

4 Call the appropriate package in #XJXV to transmit the data

5 Open the IDE files IDECNTRL containing the COPY information and TRANS01 containing the batches of data to be transmitted

In addition the mainframe operator will load the program #XIDE at the end of the transmission session. This is the program that reads the mainframe spoolfile and processes all the batches, writing each one to the appropriate file. See section 6.8 for a description of #XIDE.

Transmission is controlled by the supervisor using one video as a console. The console is created by inserting the console PID, numbered 255, into the video's PID reader.

At the end of the transmission session it is recommended that a security copy of the transmit file is taken, see section 6.7.

6.1 Establishing the communications link

The terminal (or terminals) will be connected to the mainframe either directly (via a private line) or over a switchable telephone line. The supervisor must be aware of how the terminal is connected to the mainframe, the possible transmission speeds, the protocol used, the mode of transmission and whether a standby connection is possible. Many of the variables associated with a connection are mainframe dependent and are set at installation time. The supervisor should always consult the mainframe operator if in doubt to decide the connection used and to ensure that the mainframe communications equipment is available.

The following section gives a brief description of how 7501 and 7502 systems may be connected to the mainframe.

6.1.1 Terminal to mainframe connection

The type of connection between the terminal and mainframe is set at installation time. There are two basic types of connection:

1 DIRECT CONNECTION The terminal is connected directly to the mainframe without using modems. Direct connection is used for 7501 or 7502 systems local to the mainframe site. The maximum distance that the 7501 and 7502 can be sited from the mainframe is dependent on the transmission speed and the communications facilities at the mainframe

2 TELEPHONE CONNECTION The terminal is connected remotely to the mainframe by a PTT line using modems. The connection can be one of two types:

(a) Leased to the user by the PTT authority. This connection is sometimes known as a *dedicated* line

(b) *Dial-up* connection over the Public Switched Telephone Network (PSTN). A dial-up connection is often available as standby to a leased connection. Some modems allow a *double dial-up* connection to be used, providing full duplex operation over the PSTN

The procedure for getting on-line for a direct or leased telephone connection is described in section 6.1.3 and for a PSTN telephone connection in section 6.1.4.

6.1.2 Preparing the terminal

To prepare the terminal before getting on-line proceed as follows:

1 Ensure the processor and the required terminal equipment are connected to the mains supply

2 Ensure that a cable is connected to the MODEM 1 socket on the rear panel of the 7502 processor, or to the MODEM socket on the rear panel of the 7501 processor. The cable should go to the mainframe if a direct connection is used, or to the modem if a telephone connection is used

6.1.3 Direct connection and leased telephone line connection

The procedure for getting on-line when the terminal is directly connected to the mainframe or is connected by a leased telephone line is as follows:

1 Set the DATA SET switch on the processor to DATA

2 Ensure the mainframe communications equipment is switched on

3 Switch on the processor. The DSR lamp on the processor should light

The processor is now ready to be loaded with a TE as described in section 6.3.

6.1.4 PSTN telephone connection

The procedure for getting on-line when the terminal is to be connected to the mainframe by a PSTN line is as follows:

1 Switch on the processor

2 Set the DATA SET switch on the processor to VOICE. Use the telephone handset usually provided with the modem to dial the number of the mainframe location

3 When the call is answered by the mainframe operator ask for the mainframe communications equipment to be set up and connected. Agree with the mainframe operator type of connection and the speed of transmission

4 Set the DUPLEX switch on the processor to HALF. If a double dial-up connection is being used set the switch to FULL

5 Set the STANDBY switch on the procesor to the type of connection decided in step 3

6 Set the DATA RATE switch on the processor to HIGH or LOW depending on the modem in use and the transmission speed decided in step 3

7 Set the DATA SET switch to DATA. The DSR lamp should light to indicate that the connection is established.

If the DSR lamp does not light repeat the procedure above from step 2. Consult the fault finding procedures in *Hardware Operating*. If the DSR lamp repeatedly fails to light

The processor is now ready to be teleloaded with a TE as described in section 6.3.

6.1.5 Setting the control switches on the processor

When setting the control switches on the processor for data transmission the following points should be noted:

1 DATASET switch is always set to DATA

2 When operating in full duplex mode the STANDBY switch is always set to NOT

3 If a PSTN connection is being used and the STANDBY switch is set to select, the DUPLEX switch must be set to HALF

4 On multidrop circuits, the DUPLEX switch is always set to HALF

The following is a list of modem types with possible combinations of the DUPLEX, DATA RATE and STANDBY switches for each modem. Characteristics of the connection may be given in the *Comment* column.

Modem	*Control switch settings*			*Line speed*	*Comment*
	DUPLEX	*DATA RATE*	*STANDBY*	*(bps)*	
7B	FULL	HIGH	NOT	2400	Leased line, four wire point to point
	HALF	HIGH	NOT	2400	Leased line, multipoint
	HALF	HIGH	SELECT	1200	PSTN
	HALF	LOW	SELECT	600	PSTN
7C	HALF	HIGH	NOT	2400	PSTN, two wire, phase modulated
	HALF	LOW	NOT	1200	PSTN, two wire, phase modulated (see note 1)
	HALF	HIGH	SELECT	1200	PSTN, two wire, frequency modulated
	HALF	LOW	SELECT	600	PSTN, two wire, frequency modulated
1 or 20	FULL	HIGH	Not	1200	Leased line, four wire
	HALF	HIGH	applicable	1200	Leased line, multipoint asynchronous
	HALF	LOW		600	PSTN, two wire
2 or 13	FULL	Not applicable		300	Asynchronous
11	FULL	HIGH		4800	Leased line, four wire
	FULL	LOW	Not	2400	point to point
	HALF	HIGH	applicable	4800	Leased line multipoint or
	HALF	LOW		2400	two wire PSTN
12	FULL	HIGH	NOT	2400	Leased line, four wire, point to point
	FULL	LOW	NOT	1200	Leased line, four wire point to point (see note 2)
	HALF	HIGH	NOT	2400	Leased line, multipoint
	HALF	LOW	NOT	1200	Leased line, multipoint (see note 2)
	HALF	HIGH	SELECT	2400	PSTN, two wire
	HALF	LOW	SELECT	1200	
24	FULL	HIGH		4800	Leased line, four wire or
	FULL	LOW	Not applicable	3200	double dial-up (see note 3)
	HALF	HIGH		4800	Leased line, multipoint;
	HALF	LOW		3200	or PSTN
OEM	FULL	HIGH	As	As	(See note 3)
	HALF	HIGH	specified	specified	

Notes:

1 Phase modulation may give better throughout than frequency modulation with this modem at 1200 bps

2 If the line characteristics are suspect this combination of switch settings can be used

3 Modem 24 and some OEM modems allow a four wire PSTN connection to be used. This is known as *double dial-up* and provides full duplex operation over the Public Switched Telephone Network

6.2 Ensuring that #XJXV is running on the mainframe

The program #XJXV is an overlaid utility program which runs in the mainframe C: program slot and controls the transfer of data to and from 7501 and 7502 terminal systems.

#XJXV is constructed from several packages of which two are used for the transmission of data to the mainframe:

1 CONSOLE MANAGER (CM) This package provides general program facilities and is mandatory

2 SPOOLER INPUT (SI) This package allows batches to be input to the mainframe from terminal systems

When a transmission session is about to start the terminal supervisor should ask the mainframe operator to load #XJXV before T2RB31 is loaded on to the terminal.

To load and initialise #XJXV the mainframe operator:

1 Ensures that the file ICLKXJXVFILE is on-line

2 Loads #XJXV into the C: program slot using the command:

FI#XJXV

The program displays the message:

DISP-XJXV/*xx*

where *xx* is the program identifier, and copies its overlays into ICLKXJXVFILE. When initialisation is complete, the program file is released and the following message is displayed:

DISP-INITIALISED

After initialisation the program waits to be informed by Executive of any on-line terminals loaded with suitable terminal executives. (For IDE transmission the terminal is loaded with T2RB31). As each terminal comes on-line the message:

DISP-**L-*ll*T-*tt*P-*x*

is displayed, where

ll is the line number, in decimal

tt is the terminal address

x is either NONE or a string of two-character identifiers indicating the packages which have been selected at that terminal.

A similar messae is displayed whenever the state of the terminal changes.

For further details of the mainframe operation of #XJXV the operator should consult *Remote Job Entry to the 2903 Range*.

6.3 Loading T2RB31

Before the start of the session using T2RB31, make sure that the following items are available:

1 A terminal executive library diskette containing T2RB31

2 The diskette(s) containing the files to be transferred to the mainframe

3 A console PID numbered 255

Also make sure that #XJXV is running on the mainframe.

The operating procedure for T2RB31 is detailed below. Only the console video and disc drive are supported by this terminal executive. The console video has to be the lowest numbered video.

1 Switch on the console video if this has not already been done

2 Insert the terminal executive library diskette containing T2RB31 into drive 1 of the dual floppy disc drive or drive 2 if drive 1 is inoperable. Select position 3 on the rotary switch of the processor

3 *Genres* and *activate* using the appropriate switch on the processor. The diskette menu select display will then appear

4 (a) Use the blue numeric key(s) to enter the appropriate *TE No.* for T2RB31

(b) Press the SEND key

The message 'T2RB31/*nnn* will appear on the screen where *nnn* is the issue number of the terminal executive

5 Remove the terminal executive library diskette and insert the diskette(s) containing the files to be transmitted into the drive(s)

6 Insert the console PID into the vidoe used to control the terminal executive

7 The mainframe should display the message **#XJXV/*xx* INITIALISED on the console

6.4 Calling the #XJXV Spooler Input package

The #XJXV command *Set Package* (SP) must be entered at the terminal. This enables the terminal to use the Spooler Input (SI) package to transfer the data from the transmit file to the mainframe. The command should be used as follows:

1 Press the INTERLOCK key

2 Type CM=SP SI

3 Press SEND

When the SP command has been accepted the mainframe replies to the terminal with the following messages:

O.K.←

**SI: CR-FIX

The files for transmission should then be opened as described in section 6.5.

Notes:

1 Response to the SP command may incur a slight delay since it may cause #XJXV to expand in size

2 A package remains associated with a terminal until #XJXV has been deleted and reloaded and the message **XJXV/*xx* INITIALISED is again output

For further details on #XJXV operation see *Remote Job Entry to the 2903 Range*.

If the diskettes containing the files are not in the drives they should be loaded at this point.

6.5 Opening the files for transmission

The files IDECNTRL and TRANS01 must be opened using the OF command as follows:

1 Press the LOCAL INPUT key

2 Type OF *filename,class*

where

filename is the name of the file to be opened

class is B

3 Press the SEND key.

If the file is opened successfully the following message is displayed:

BUF FILE *filename* OPENED

where BUF is *Basic Unit File* which is the type of file used by the IDE system.

and *filename* is the name of the file

Transmission will then start automatically.

Notes:

1 IDECNTRL should be opened first

2 After IDECNTRL has been transmited as described in section 6.6 TRANS01 should be opened. If there is more than one transmit file, set up as specified in Appendix 2, these will be automatically opened and transmitted in sequence provided that the next transmit file in the sequence is loaded prior to the completion of tramsission of the current file. Diskettes containing further files in the sequence may be loaded before or during transmission provided that not more than 20 files are on-line at the same time. If the next file in the sequence is not on-line

transission will cease. It can be restarted by loading and opending the next transmit file

6.6 Transmitting data

When transmission of the file starts, the message displayed at the terminal is:

BUF FILE TRANSMISSION STARTED

**SI:DOC:*d*

where *d* is the name of the document being transmitted:
IDECONTROL Or IDEUSERDATA1 to 9.

Transmission of the file will proceed until either:

1 The end of the file is reached

2 The operator abaondons the transmission

3 An error occurs whilst reading from the diskette

At which time the messages displayed are:

**SI: CR-FIX

BUF FILE TRANSMISSION ENDED

N RECORDS READ

where *N* is the total number of records read from the file.

The file will then be closed automatically by the terminal executive, the message displayed being:

BUF FILE *filename* CLOSED

If transmission ceases because of reaching the end of file, then one of the following applies:

1 If the previous filename was IDECNTRL, then the terminal executive will do nothing until the operator opens TRANS01

2 If, however, the previous filename was TRANS*nn* and TRANS*nn+1* is on-line, then it will be opened automatically and transmissin will proceed as described above

3 If the next file is not loaded, then the following warning message is displayed:

BUF FILE *filename* NOT FOUND

If another transmit file is to be transmitted the operator should then load the diskette containing the file and type in the OF command. In this case the filename may be omitted from the OF parameter (that is OF ,B may be typed)

4 If the file fails to open for any other reason than in 3 then the following messages are output

ERROR*xx*

BUF FILE *filename* FAILED TO OPEN

where *xx* is the normal two character error code covered in *7502 Operating* or *Software Operating*

6.6.1 Abandoning a file

Transmission of a file may be abandoned at any time as follows:

1 Press the LOCAL INPUT key

2 Type AB

3 Press the SEND key. Messages will be output as for normal closure.

Transmission will cease, and the file will be closed automatically. No attempt will be made to open the next file in sequence where this is applicable

6.6.2 Error messages

If transmission of a file ceases because of an error then an error message is output, followed by automatic closure of the file. No attempt is made to open the next file in the sequence.

A transmission failure error message has the following format:

BUF FILE ERROR TF ON T*tt* S*ss*

where

TF indicates transmission failure

tt is the absolute track number on the diskette

ss is the relative sector number of the track

Other error messages associated with T2RB31 can be found in *7502 Operating* or *Software Operating*. Other #XJXV messages can be found in *2903 Remote Job Entry*.

6.6.3 Closing transmission session files

It is not necessary to close any of the files opened for transmission before removing the diskettes.

6.7 Taking security copies, creating an empty transmit file and retransmitting data

When transmission is complete the transmission link with the mainframe can be broken, the diskettes can be removed and the terminal switched off. However before commencing the next data entry session an empty transmit file (TRANS01) and an empty COPY information file (IDECNTRL) must be created. It is also recommended that security copies are taken and maintained long enough to ensure that all the batches of data transmitted have been sucessfully processed at the mainframe.

Creating these security copies and the empty files for a new data entry sesion should be carried out at the end of the transmission session using the utilities described in Appendix 1. There are two methods recommended depending on the terminal system being used:

1 TERMINAL WITH TWO DUAL FLOPPY DISC DRIVES To create the security copy it is recommended that, rather than write the contents of the transmit file to another file, the transmit file is *renamed* to form the copy. Then to create the new empty transmit file for the next data entry session an old transmit file security copy should be renamed TRANS01 and this new file *tidied* and *erased*. If there is more than one transmit file, security copies of all these files should be created. It is not necessary to keep a security copy of IDECNTRL but an empty version is required for the next data entry session. Therefore at the end of the transmission session IDECNTRL should be tidied and erased.

The functions of renaming,tidying and erasing files are all carried out using the utility commands RF (RENAME FILE), TF (TIDY FILE) and ER (ERASE FILE) described in Appendix 1.

Should it be necessary to retransmit any data, supervisor function 13 described in section 5.2.12 should be used to select the batches for retransmission and write them to the current transmit file. Supervisor function 9 described in section 5.2.8 must the be used to add COPY information for these batches to the current IDECNTRL file

2 TERMINAL WITH SINGLE DUAL FLOPPY DISC DRIVE In this case it is recommended that the complete diskette containing the files TRANS01 and IDECNTRL is copied to an empty diskette, retaining the correct filenames. This copy should be carried out using the VC (VOLUME COPY) utility command specified in Appendix 1.

Note: Because the transmit file and some entry files will be on the same diskette it is not possible to rename the transmit file to create the security copy as described for a terminal with two dual floppy disc drives since the entry files would then only be on the security copy diskette.

After the volume copy has been completed, the original files TRANS01 and IDECNTRL should be tidied and erased using the utility commands TF (TIDY FILE) and ER (ERASE FILE) described in Appendix 1. These commands will cause the files to be emptied ready for the next data entry session.

In order to retransmit data the security copy containing the data should be loaded in the place of the current TRANS01 and IDECNTRL files. A transmission session should then be initiated. With this terminal system it is not possible to selectively retransmit batches. All the batches on the security copy will be retransmitted with their original COPY information

Notes:

1 Before attempting to use any of the utility commands the files on each diskette to be used should be opened using the OV (OPEN VOLUME) command described in Apppendix 1

2 After the utility commands have been used all the files on the diskettes used should be closed using the CV (CLOSE VOLUME) command described in Appendix 1

6.8 Operating #XIDE

During the transmission session the batches of data on the transmit file and their associated information on the IDECNTRL file are written to documents on the mainframe spoolfile. Document IDECONTROL contains the IDECNTRL file and documents IDEUSERDATA*n* contain the transmit files, *n* being the number between 1 and 9 referring to file TRANS01 to 09. Program #XIDE runs on the mainframe and reads these spoolfile documents then writes the data to user files or spoolfile documents as specified by the COPY information supplied for each batch. (See section 5.2.8 for a description of the COPY information supplied.) #XIDE also writes batch statistics and operator statistics if present to the IDEBATCHSTAT and IDEOPERSTATS files.

#XIDE must be run after every transmission session.

To operate #XIDE proceed as follows:

1 Load #XIDE by typing:

FI:#XIDE#XXXX

where #XXXX is the name of the mainframe file containing #XIDE

2 Assign the transmission spoolfile documents to the program by typing:

```
AS CR10, IDECONTROL
AS CR1, IDEUSERDATA1
AS CR2, IDEUSERDATA2
.                  .
.                  .
.                  .
AS CR9, IDEUSERDATA9
```

Note: CR2 and CR9 need only be assigned if the respective transmit file has been transmitted

3 Load the files IDEBATCHSTAT, IDEOPERSTATS, IDEDWORKFILE and any user files that the batches will be written to

4 Type GO 20

5 If no COPY information has been supplied for a batch at the terminal, #XIDE displays the message:

Disp: COPY INFORMATION MISSING FOR *batchname*
Halt: LOAD ACCEPT DATA ON CR*n*

where *batchname* is the twelve character name of the batch and *n* is the number of the card reader.

In order to provide the missing COPY information a card format described in Appendix 4 should be created as specified in section 5.2.8 to supply this information. The facility exists to specify as COPY information that this batch should be *ignored*. In this case the batch is lost but batch statistics are written to the IDEBATCHSTAT file. If the batch is required later it must be retransmitted, see section 6.7.

The card should be inserted in the card reader and GO typed

6 If a required user file is not on-line #XIDE displays:

Disp: *filename* FGN*n* NOT ONLINE
Halt: COBOL ACCEPT SEE JOB SHEET

The program can be restarted in one of three ways:

(a) By putting the user file on-line

(b) By supplying different file information on the COPY information card described in Appendix 4

(c) By abandoning copy of any batches to this user file

In order to indicate which of the three choices has been made word 0 of the program should be altered to 1, 2 or 3 by typing:

AL∇0,*n*

where *n* is 1, 2 or 3 and 1 indicates that the file is now on-line, 2 indicates that new COPY information will be supplied and 3 indicates that any batches to be written to this file should be abandoned.

Then type GO

7 Having completed the run #XIDE displays:

END OF RUN

Other messages that may be displayed when running #XIDE are:

1 *Halt*: RUN ABANDONED

This may be preceded by

(a) :*batchname*
: INVALID REC TYPE *x*

(b) :DA ERROR *nnnn* ON *filename*

These errors should be reported immediately to the ICL representative. An attempt may be made to re-run #XIDE or to clear documents IDECONTROL and IDEUSERDATA*n* and retransmit before re-running #XIDE

2 :EOF ON CR*n*

Where *n* is the number of the card reader assigned to a transmit file document.

Ths will not cause the run to be abandoned but indicates that the transmit file document assigned to that card reader is empty. If there should be data on that file check with the terminal supervisor that the correct file was transmitted

3 :UNPAIRED RECORD ON
USER DATA FILE *n*

where *n* is a number in the range 1 to 9 depending on the transmit file number from which the record was read. If this message is displayed it will not cause the run to be abandoned but ICL should be informed

4 *Disp*: CARD INVALID
Halt: LOAD ACCEPT DATA ON CR*n*

This will occur if an invalid card has been used to provide COPY information. The card should be repunched and input again

5 *Halt*: ON LINE *filename*

This occurs if a system file is not on-line. The file should be loaded and the program restarted by typing GO

6 *Disp: batchname* BATCH HEADER MISSING
Halt: PRINT 0-3000 OF CORE

This error should not occur. The core print should be taken and the ICL representative informed immediately. An attempt may be made to clear documents IDECONTROL and IDEUSERDATA*n* and then to retransmit before re-running #XIDE

7 Disp: *filename* FILE FULL

where *filename* is the name of either the batch statistics file or the operator statistics file. The appropriate file should be extended and the program resumed by typing GO

8 The following messages indicate an error in writing the spoolfile

(a) SD PERI COMMANDS NOT SUPPORTED

(b) ITEM INDEX FULL

(c) INSUFFICIENT CORE FOR ASSIGNMNET

(d) INSUFFICIENT SPACE ON FILE FOR REQUEST

(e) CARD TRANSFER ERROR

(f) ERROR IN FUNCTION CODE

(g) FAILURE ON CLOSING DOCUMENT

For information on these errors see the publications relating to the mainframe executive in use.

#XIDE also prints an incident report on the line printer and provides an analysis of the batches it has written to the mainframe files. An example of the printed output is shown in Figure 6.1.

```
                    ANALYSIS OF IDE FAILES 31010930          USER DATA FILES          38/03/80

OUTPUT FILE      FGN    BATCH NAME          ACTION                                    IDE FILE

IDEUSERDATA1     0000   BATCH 1             CANNOT BE COPIED - NO DATA

IDEUSERDATA1     0000   BATCH 3             COPIED SUCCESSFULLY

IDEUSERDATA1     0000   BATCH 2             COPIED SUCCESSFULLY
```

Figure 6.1 #XIDE printout example

7501/7502 operating is described fully in the ICL publications *7502 Operating, Hardware Operating* and *Software Operating*. This appendix briefly describes the main operating features that will be required when using IDE.

A1.1 7501/7502 system components

ICL's 7501 and 7502 intelligent modular terminal systems provide a wide range of on-line and off-line facilities in an office environment at remote or local sites.

The 7501 modular terminal system is run under the control of the 7501 processor loaded with a terminal executive. Up to two videos can be linked in a 7501 system. The 7501 processor is integral with the first video which also includes a Personal Identification Device (PID) reader. One or two dual floppy disc drives can be connected to the system.

The 7502 modular terminal system consists of a processor, a mandatory video with a PID reader, up to three additional videos and up to two dual floppy disc drives.

Each dual floppy disc drive provides two direct access storage devices using small removable flexible magnetic discs called diskettes.

The 7501 or 7502 is controlled by one of a number of terminal executives (TEs). A TE must be loaded into the store of the processor by the supervisor according to the particular function she wishes to carry out. Switching from one function to another (apart from the loading and dumping processes) requires the loading of a different TE, and the TE that is loaded will only support those devices necessary for the function.

The TEs required for the different IDE functions are listed in section 2.5. These TEs will be issued by ICL to the mainframe installation. They must be *teleloaded* from the mainframe to the terminal where they should be held in a TE library file on a single diskette. Teleloading and the creation of a TE library diskette are described in *7502 Operating* or *Hardware Operating* and *Software Operating*.

A1.1.1 The processor

The front panels of the 7501 and 7502 processor are shown in Figures A1.1 and A1.2. A description of the function of each control and indicator can be obtained from *Hardware Operating*. For the purposes of IDE the control required most often is the *genres/activate* switch. This switch is a three-position switch used to initiate the function selected by the white *rotary* switch (described below).

When moved up it resets the entire processor and when pressed down it initiates hardware checks followed by the selection function. With IDE it is used to initiate the loading of a new TE from the TE library diskette.

The rotary switch shown on the front panel of both the 7501 and 7502 in Figures A1.1 and A1.2 provide the supervisor with a selection of loading and dumping routines. A full description of the function provided by each position is given in *Hardware Operating*. For the purposes of IDE the most used position is position 3; LOCAL LOAD. This provides for the loading of a terminal executive from diskette.

A1.1.2 The dual floppy disc drive

The dual floppy disc drive (see Figure A1.3) is located in the furniture unit directly above the MTP to which it is connected.

The drive provides two small random access storage devices which hold the data, the screen formats, the IDE software and the TEs on small exchangeable diskettes (see Figure A1.4).

The diskette is permanently enclosed in a sealed jacket and when loaded rotates inside its jacket. When the diskette is inserted by the supervisor and the door closed, the diskette starts to rotate and the drive is ready for the transfer of information.

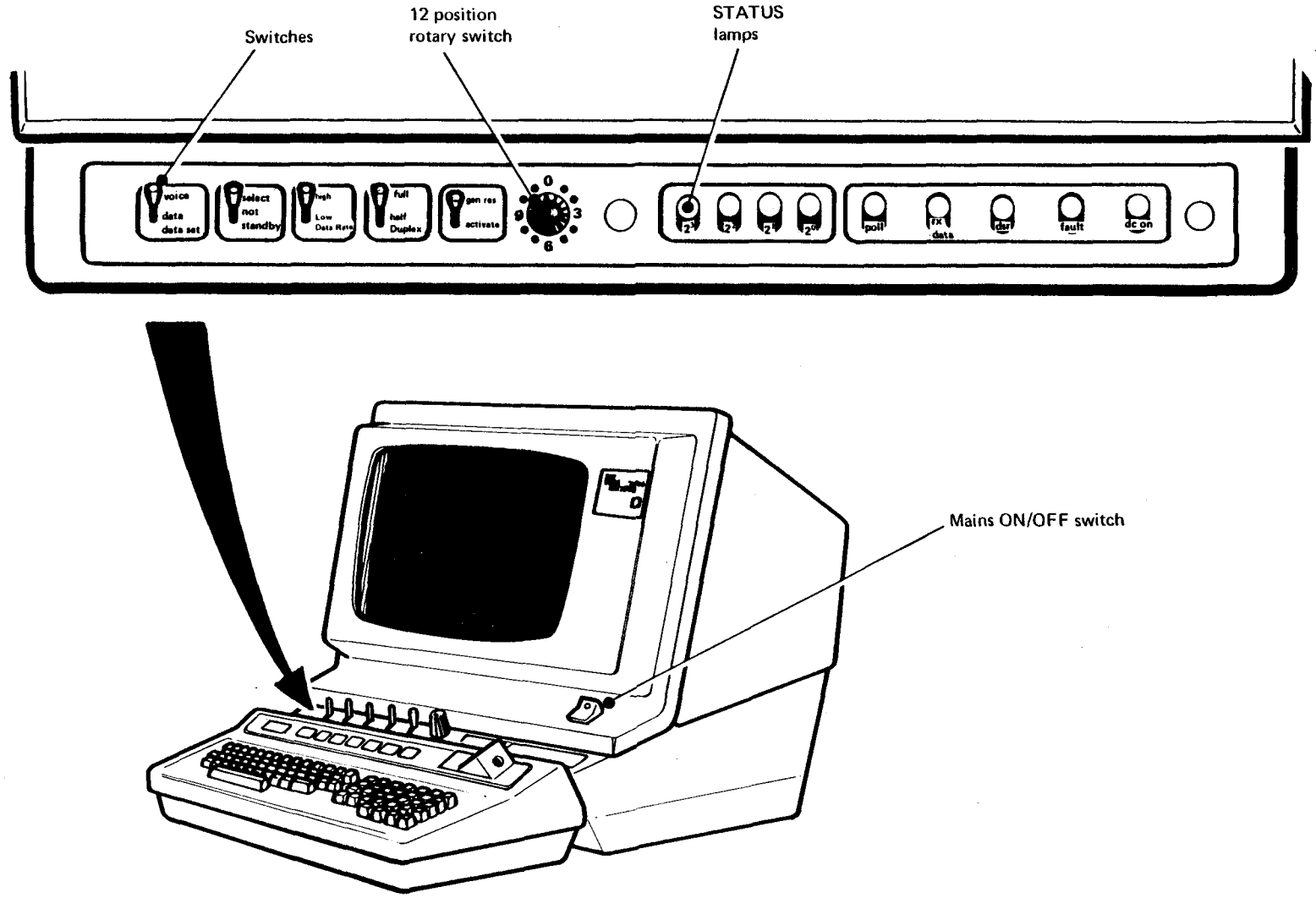


Figure A1.1 The 7501 MTP front panel

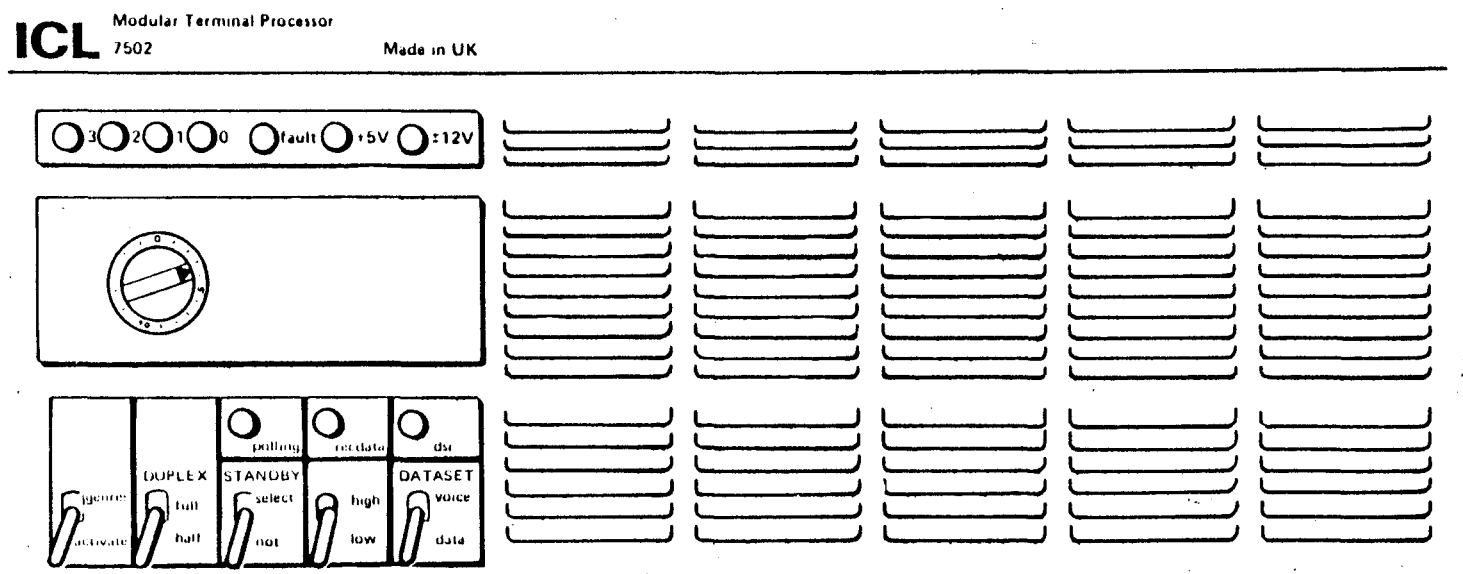


Figure A1.2 The 7502/05 MTP front panel

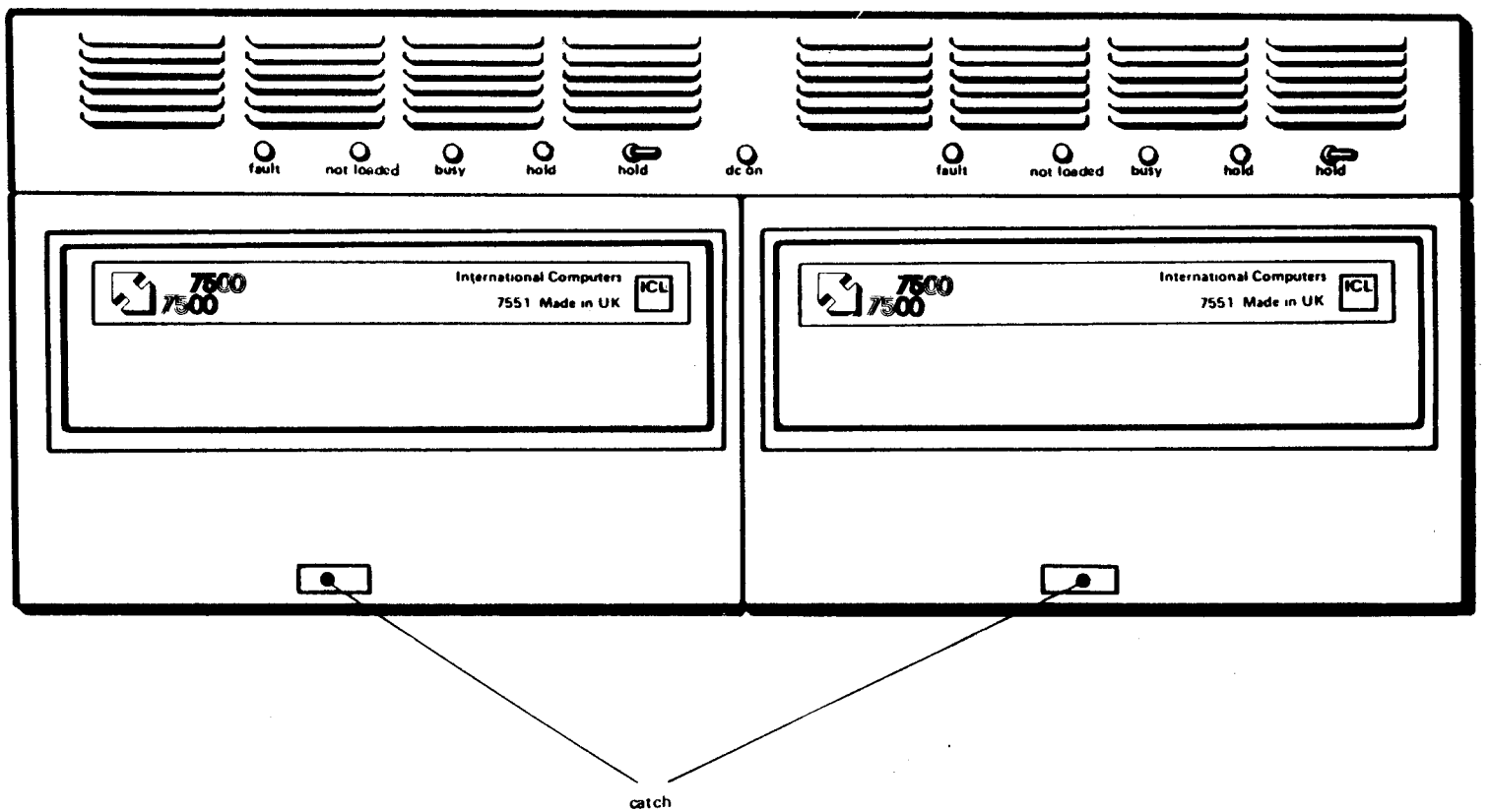


Figure A1.3 The dual floppy disc drive

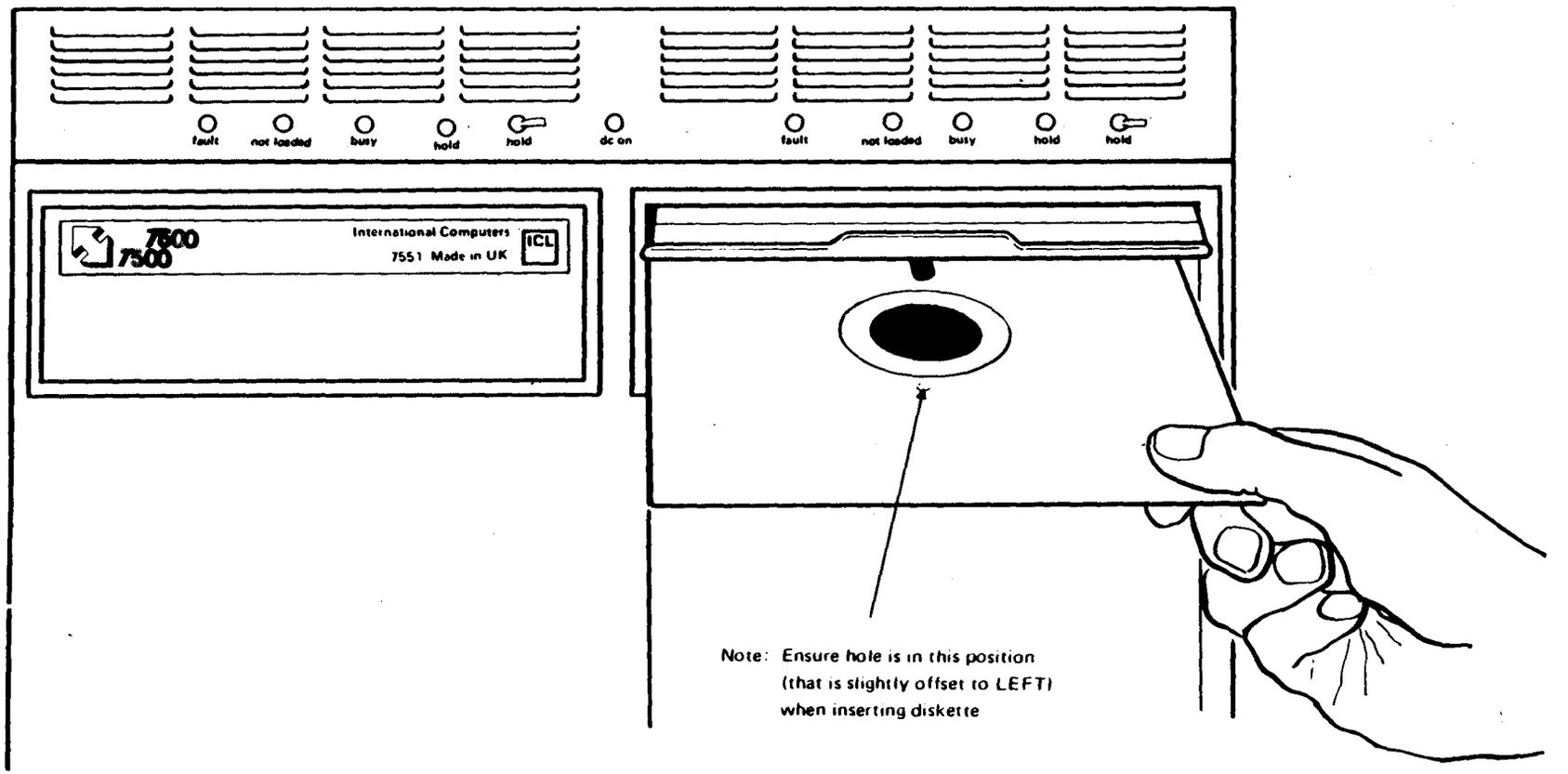


Figure A1.4 Loading a diskette

A 7501 or 7502 system may have one or two dual floppy disc drives attached thus providing for the use of two or four diskettes.

There are two controls on each dual floppy disc drive: a small release catch and a HOLD switch. The HOLD switch is described in *Hardware Operating* but should not be used. The release catch is used to open the spring-loaded flap prior to inserting the diskette in the drive, or removing it from the drive. However, the diskette should not be removed until it has been 'closed' as outlined in section A1.1.2.2.

Note: There must be no diskettes in any drive when the terminal is switched on or off.

A1.1.2.1 *Inserting the diskette*

Pressing the catch below the flap opens the flap. The jacket with diskette is inserted in the slot the correct side up (see Figure A1.4) and the flap is closed. Closing the flap presses the diskette onto a rotating conical spindle which centres the diskette and rotates it within the stationary jacket.

A1.1.2.2 *Removing the diskette*

Before removing a diskette the appropriate close down procedure must be followed. If it is not data may be lost or corrupted. The close down procedure differs depending on the IDE function being performed at the time. Closing the diskettes is described in section 3.3.3 (ending a development session), section 5.4 (closing a data entry session), section 6.6.3 (closing transmission session files) and section A1.4 (T2UxI operating).

A1.2 Use of the console facilities

One of the videos on the system is used as a console to input commands to the terminal executives and to provide a means of sending and receiving messages to and from the terminal and the mainframe. This provides the supervisor with a means of controlling the stand-alone operation, and generally controlling and setting up the system.

The console video is identical to the other videos connected to the terminal but before it will provide console facilities, the supervisor must fit a console Personal Identification Device (encoded 255) into the PID reader on the video.

If the supervisor wishes to use her console as an ordinary video for the purpose of keying into screen formats during stand-alone operation she has only to remove the console PID.

A1.2.1 Typing in commands to the TE

To type a command proceed as follows:

1 Press the LOCAL INPUT key

2 Type the command

3 Press the SEND key

4 If the command is not valid an error message will be displayed. For a list of error messages see *Software Operating*

5 If the command is successful, the message DONE is displayed

A1.2.2 Typing in messages to the mainframe

To type a message proceed as follows:

1 Press the INTERLOCK key (which moves the cursor to the bottom of the screen)

2 Type the message (which must not be more than 160 characters) on the bottom two lines of the screen

3 Press the SEND key

A1.2.3 Messages to the operator

Messages from the mainframe appear on the bottom two lines of the screen. Their format is dependent on mainframe software, and they are displayed immediatelv they arrive at the bottom of the screen. If a message arrives at the terminal

while the operator is typing in a message (or command) on the keyboard, the mainframe will hold its message until the operator has finished typing and pressed SEND. The message from the mainframe will then be displayed on the bottom lines of the screen.

A1.3 Loading TEs from diskette

The TEs required to use IDE must be held on a single diskette termed the TE library diskette. Instructions on how to create a TE library diskette can be found in *7502 Operating* or *Software Operating*.

When loading any of the TEs from this TE library diskette the loading program, which is permanently in the processor, loads the TE from track 1 on the lowest numbered drive. If there is no diskette on drive 1, or the flap to drive 1 is left open, it loads from drive 2. The first TE loaded will be the local menu selection TE T2U7I which when loaded displays the list of TEs that are available on this particular TE library diskette.

The procedure for loading is as follows:

1 Ensure that the diskette containing the TE library is installed in either of the drives

2 Select the rotary switch position 3 and initiate the load by using the *genres/activate* switch. All video screens are cleared and provided that no fault is found, the load is commenced. If a fault is found the *fault* lamp on the processor is lit. *Hardware Operating* lists faults

3 As the load takes place each block is checked and if an error is detected an indication of the error is displayed on the error lamps on the processor (see *7502 Operating* or *Hardware Operating*)

4 If, during the load, the *fault* lamp on the drive lights the operator should attempt the load from the other drive. If this does not clear the fault refer to the Fault Finding procedures in *7502 Operating* or *Hardware Operating*

5 Once T2U7I is loaded it will display a list of the TEs available from that diskette

6 Type the number of the required TE into the *TE No* field, press SEND and the required TE will be loaded and started automatically

A1.4 T2UxI operating

The T2UxI series of TEs allows the supervisor to perform a number of file-handling operations. Each operation is executed when the supervisor types in the corresponding command. This section provides information on the T2UxI commands used with IDE. The commands are listed in alphabetical order and each is followed by a list of error messages and their meanings. For a full list of the T2UxI commands available see *7502 Operating* or *Software Operating*.

The commands referred to above are typed in at the console video as described in section A1.2.

Before using some of the T2UxI series commands the files on each diskette to be used must be opened using the OV (OPEN VOLUME) command. This command opens all the files on a diskette. If an OV command has been used the diskette should be closed using the CV (CLOSE VOLUME) command after the required operation is complete.

In the following descriptions of the T2UxI commands optional parameters are enclosed in square brackets [ ], these brackets are not to be typed in. A space character must be used to separate the two-character command mnemonic from the parameters following. Commas and not spaces, must be used to separate parameters from each other.

A1.4.1 CV command (CLOSE VOLUME)

Purpose

Used to close down any diskettes currently in use.

Executive required

T2U2I, T2U3I, T2U4I or T2U6I

Procedure

1 Ensure that one of the required executives is loaded

2 Insert the console PID into the PID reader

3 Ensure that the diskette to be closed is in one of the disc drives

4 (a) Press the LOCAL INPUT key

(b) Type CV [*vol name*]

where

vol name is the six-character name of the diskette to be closed. If *vol name* is present, that diskette will be closed, otherwise all on-line volumes without files currently in use will be closed

(c) Press the SEND key

5 If the command is successful, the message

DONE

is displayed

Users should ensure that, where the *vol name* is used, the appropriate diskette is loaded.

Error messages

Where the CV command is used with the *vol name* parameter, there are three possible causes of failure, each indicated by an error message:

Message	*Cause of failure*
Error IV	The volume name is invalid
Error NV	The specified volume is not loaded
Error FO	The specified volume still has files open

A1.4.2 DI command (DISPLAY INDEX)

Purpose

Used to display the contents of the index track of a diskette. This can be used to check the creation date of files, when this date has been input using the set date (SD) command described in section 5.1.1. This also displays the name and state of the files.

Executive required

T2U4I, T2U2I, T2U3I or T2U6I

Procedure

1 Ensure that one of the required executives is loaded

2 Insert the console PID into the PID reader

3 Insert the required diskette into one of the disc drives

4 (a) Press the LOCAL INPUT key

(b) Type DI *drive no.*[,*file no.*]

where

drive no. is the number of the drive containing the volume; this number is marked on the front of the drive, and is in the range 1 to 4.

file no. is the number of the file from which the display commences. Where this parameter is 0 or absent, information about all the files on the volume will be displayed

(c) Press the SEND key. A display of the format shown in Figure A1.5 will appear on the screen

The significance of the fields displayed is as follows:

Field	*Meaning*
FILE NO	The number of the file(s) on the current volume. Up to 19 files can exist on one volume
FILENAME	The name of each file on the diskette. The file named FREEZONE contains all the free space on the diskette
EXTENT	The number of tracks occupied by the file
SPACE USED	The amount of space in the file that contains data. Thus 19/7 means that 19 tracks and 7 sectors of the file are filled with data (sector 8 may be partially filled)
TYPE	The data class of the file: APLIB = path library files: IDE, IDEDEVEL, IDEPATHS TELIB = terminal executive library file BUF = all the IDE files except those mentioned above
TRANSMITTED	If the file has been copied or transmitted, this field will contain the letter C
VERIFIED	Not used at present
O/C FLAG	If the file has been opened for write access and not yet closed, this field will contain the letter O
WRITE PROTECT	If the file is write protected (see the TF command), this field will contain the letter P
CREATION DATE	Date on which the file was created: six alphanumeric characters (no spaces)
EXPRIRY DATE	Date of expiry of file: six alphanumeric characters (no spaces)

VOLUME:- DUMP01 FLAWED TRACKS:- NONE

FILE NO.	FILENAME	EXTENT	SPACE USED	TYPE	TRANS-MITTED	VERI-FIED	O/C FLAG	WRITE PROTECT	CREATION DATE	EXPIRY DATE
1	DUMPAREA	20	0/0	DUMP						
2	MONDAYS	26	19/7	SPOOL	C			P	230178	300178
3	CUSTNOS	12	11/8	INDREC			O	P	010178	311299
4	CUSTOMY	12	11/8	INDREC				P	100178	120178
5	FREEZONE	3	3/0							

Figure A1.5 The DI command display (T2UxI)

Error messages

Messages	*Cause of failure*
Error DI	The disc drive is inoperable
Error IN	The specified file number is invalid
Error IU	The specified drive number is too large
Error MP	A parameter is missing or a comma needed to separate parameters is missing or misplaced
Error TF	A read error has occurred

A read error can occur for two reasons:

1 Failure to transfer to the screen the volume label or error map sector, in which case th display is abandoned .

2 Failure to read a particular file header, in which case the message

HEADER UNREADABLE

is displayed against the appropriate file number

A1.4.3 ER command (ERASE FILE)

Purpose

Used to erase the entire contents of a particular file, so that the file can be used for a new function.

Executive required

T2U4I

Procedure

1 Ensure that T2U4I is loaded

2 Insert the console PID into the PID reader

3 Insert the diskette containing the file to be erased into one of the disc drives

4 Perform step 4 of the procedure for the OV command (see this command)

5 (a) Press the LOCAL INPUT key

(b) Type ER *vol name,filename,class*

where

vol name is the six-character volume name of the diskette containing the file

filename is the name of the file to be erased

class is B for all IDE files except the path library files: IDE, IDEDEVEL and IDEPATHS (*class* A) and the terminal executive library file (*class* T)

(c) Press the SEND key

The ER command is only valid if the file to be erased has been made write protected.

Note: To use the ER command on files other than IDE files see *7502 Operating* or *Software Operating*.

Error messages

Message	*Cause of failure*
Error AE	Access Error, that is, file *filename* is not write protected
Error CE	The specified file class is invalid
Error DI	The disc drive is inoperable
Error IF	The specified filename is invalid
Error IV	The specified volume name is invalid
Error MP	A parameter is missing or commas used to separate parameters are missing or misplaced
Error NF	The specified file is not on-line
Error NV	The specified volume is not on-line
Error TF	A read/write error has occurred

A1.4.4 ID command (INITIALISE DISKETTE)

Purpose

Used to initialise a diskette.

Executive required

T2U2I

Procedure

1 Ensure that T2U2I is loaded

2 Insert the console PID into the PID reader

3 Insert the diskette to be initialised into one of the disc drives

4 (a) Press the LOCAL INPUT key

(b) Type ID *drive no.,vol name* [,*first defective track no.*] [,*second defective track no.*][,*D*]

where

drive no. is the number of the drive containing the diskette; this number is marked on the front of the drive and is in the range 1 to 4.

vol name is the six-character volume name, of which the first character must be alphabetic, of the diskette to be initialised.

first defective track no. is a two-digit number in the range 01 to 73, and refers to the lowest-numbered track to have become defective.

second defective track no. is similarly a two-digit number in the range 02 to 73.

D signifies that the diskette is to be dump enabled to permit dumping of store to diskette. This is not normally required for IDE files

(c) Press the SEND key. If the command is successful, the message

DISC INITIALISED

is displayed

The first defective track number must always be lower than the second. If track 00 is defective, the diskette must be replaced. A dump-enabled diskette must not contain defective tracks. Where dump enabling is specified, a twenty-track file called DUMPAREA is created on the diskette.

Error messages

Message	*Cause of failure*
DRIVE INOP	The disc drive is inoperable
PARAM??	An invalid parameter has been used. The first character of the offending parameter is caused to flash
WRITE ERROR TRACK *n*	A write error has occurred at track *n* on the diskette

A1.4.5 NF command (NEW FILE)

Purpose

Used to create a diskette file.

Executive required

T2U2I or T2U4I

Procedure

1 Ensure that one of the required executives is loaded

2 Insert the console PID into the PID reader

3 Insert the diskette on which the file is to be created into one of the disc drives

4 Perform step 4 of the procedure for the OV command (see this command)

5 (a) Press the LOCAL INPUT key

(b) Type NF *vol name,filename,class,filesize*

where

vol name is the volume name of the diskette on which the file is to be created.

filename is the name of the file to be created (one to eight characters, of which the first must be alphabetic).

class is one of the following:

A denotes an application programming path library file and is used when creating IDEPATHS and IDEDEVEL

B denotes a basic unit file and is used for all files other than IDEPATHS and IDEDEVEL

filesize is the size in tracks of the new file

(c) Press the SEND key. If the command is successful, the message

DONE

is displayed

Error messages

Message	*Cause of failure*
Error CE	The specified file class is invalid
Error DF	There is no space on the diskette for the file or the file header label
Error DI	The disc drive is inoperable
Error IF	The specified filename is invalid
Error IV	The specified volume name is invalid
Error MP	A parameter is missing or commas used to separate parameters are missing or misplaced
Error NF	The specified file already exists
Error NV	The specified volume (diskette) is not loaded
Error TF	A read/write error has occurred

A1.4.6 OV command (OPEN VOLUMES)

Purpose

Used to open all files on a diskette.

Executive required

T2U2I, T2U3I, T2U4I or T2U6I

Procedure

1 Ensure that one of the required executives is loaded

2 Insert the console PID into the PID reader

3 Insert the diskette(s) to be opened into the disc drive(s)

4 (a) Press the LOCAL INPUT key

(b) Type OV

(c) Press the SEND key. If the command is successful, the message

DONE

is displayed

If any file header label is unreadable, all files on the diskette will only be available for reading.

A1.4.7 RF command (RENAME FILE)

Purpose

Used to change the name of a file.

Executive required

T2U4I

Procedure

1 Ensure that T2U4I is loaded

2 Insert the console PID into the PID reader

3 Insert the diskette containing the file to be renamed into one of the disc drives

4 Perform step 4 of the procedure for the OV command

5 (a) Press the LOCAL INPUT key

(b) Type RF $filename_1,filename_2$

where

$filename_1$ is the name of the file to be renamed

$filename_2$ is the new name of the file

(c) Press the SEND key. If the command is successful, the message

DONE

is displayed

Error messages

Message	*Cause of failure*
Error DI	The disc drive is inoperable
Error IF	The specified filename is invalid
Error MP	A parameter is missing or commas used to separate parameters are missing or misplaced
Error NF	The specified file is not loaded
Error TF	A read/write error has occurred

A1.4.8 VC command (VOLUME COPY)

Purpose

Used to copy the entire contents of one volume to another volume.

Executive required

T2U3I

Procedure

1 Ensure that T2U3I is loaded

2 Insert the console PID into the PID reader

3 Insert the diskette to be copied into one of the disc drives

4 Insert an empty diskette into the other disc drive (this diskette must have been initialised; see the ID command)

5 Perform step 4 of the procedure for the OV command (see this command)

6 (a) Press the LOCAL INPUT key

(b) Type VC *vol name*$_1$,*vol name*$_2$

where

vol name$_1$ is the name of the diskette to be copied

vol name$_2$ is the name of the empty diskette

(c) Press the SEND key. If the command is successful, the message

DONE

is displayed

Error messages

Message	*Cause of failure*
Error DI	The disc drive is inoperable
Error IV	The specified volume name is invalid
Error MP	A parameter is missing or a comma used to separate parameters is missing or misplaced
Error NV	The specified volume is not loaded
Error TF Drive no *d* Track no *t*	A read/write error has occurred on drive *d* at track *t*

The IDE software is issued on a single diskette on which are the files DEVLIB, containing the Application Development System and IDE, containing the IDE software.

In order to install the IDE software so that it can be used to develop and run the data entry screen formats the Development System packages *Create Library, Clear File, Tidy Library* and *Sizing* must be used. Operating instructions for these are described in sections A2.2, A2.3, A2.4 and A2.5. For more details on these packages see *TPL Application Programming*.

The procedure for installing IDE is outlined in the steps below:

1 Teleload a terminal executive library containing the TEs:

(a) T2AP11

(b) T2AP31

(c) T2RB31

(d) T2U7I

(e) T2U2I

(f) T2U3I

(g) T2U4I

Then take a security copy of the TE library created using the T2U$_x$I series command volume copy (VC) described in section A1.4.8.

Note: Teleloading a TE library diskette is described in *7502 Operating* and *Software Operating*

2 Using the T2U$_x$I command new file (NF), described in section A1.4.5, create a file called IDEDEVEL to occupy a whole diskette, that is 73 tracks. This is not an indexed file. It is used to hold the IDE software while the Basic Data Entry path is developed

3 Load the Development System as described in section A2.1 and run the Create Library package described in section A2.2 for file IDEDEVEL. This enables the file to be used as the path library development file

4 Re-load the Development System and run the Tidy Library package (described in section A2.4) to copy the issued IDE software from the file IDE to the newly created file IDEDEVEL

5 Re-load the Development System and develop the data entry screen formats on IDEDEVEL as described in Chapter 3

6 Re-load the Development System and use the Clear File package to clear file IDE. Then use the Tidy Library package to copy the developed software from IDEDEVEL to IDE (see Note under 7)

7 Re-load the Development System and use the Clear File package to clear file IDEDEVEL followed by the Tidy Library package to copy the developed software from file IDE back to IDEDEVEL.

Note: Steps 6 and 7 tidy the verison of the software that has been used for development to remove any empty areas that have been created by amendments while developing. These steps return the tidied software to the large file IDEDEVEL on which further development can take place.

Steps 6 and 7 must be carried out every time a development session is closed even if development has not been completed

8 Take a security copy of the fully developed file IDEDEVEL using the T2U$_x$I command (VC)

9 Initialise the diskettes that will contain the remaining IDE system files. Section A1.4.4 describes the initialising of diskettes and section A2.6 indicates the number of diskettes required

10 Using the T2UxI command new file (NF) create the operational path library file IDEPATHS. This hold the fully developed operational version of the IDE software and is not indexed

11 Load the Development System and use the Create Library package for file IDEPATHS. This enables the file to be used as the operational path library file

12 Re-load the Development System and use the Tidy Library package to *shrink* the developed software from file IDE to IDEPATHS. The shrink option in the Tidy Library package is described in section A2.4. It causes the developed paths to occupy as small a space as possible

13 Re-load the Development System and run the Sizing package on the Basic Data Entry path to make sure that each screen format comes within the size defined for the current issue of IDE. Section A2.5 describes the operation of the Sizing package.

Notes:

(a) Should the processing of a screen format occupy more than the number of words defined for the current issue of IDE, the screen should be redefined in order to come within this limit. This redefinition can be done in a number of ways:

(i) The screen can be split into two or more screens

(ii) The validation criteria can be made less complicated

(iii) The COMPILE ALL facility can be used to reduce the size of the user action routines relating to this screen. This facility is described in *TPL Application Programming*

(b) The Sizing package can be run at any time during the development of the operational Basic Data Entry path to give an indication of the size of the processing of each screen format. However if the Sizing package is run before the file has been shrunk, the screen formats will register larger than their final size

(c) If alterations need to be made to the Basic Data Entry path these can only be made on a pre-shrunk file. Amendments should be made by returning to step 5 in this operational sequence and ignoring steps 9 and 10. Step 11 should be replaced by the use of the Clear File package on IDEPATHS rather than the Create Library package

14 Using the T2UxI command new file (NF) create the remaining IDE system files, none of which are indexed:

(a) OPERATOR This will hold the operator statistics

(b) SPACNTRL This will hold the supervisor password

(c) BATCHDET This will hold the batch details

(d) IDECNTRL This will hold the details of the mainframe file to which each batch is to be written

(e) ENT$_n$ These are the entry files that hold the batches as they are input; n is in the range 1 to 4 and refers to the number of the video associated with that entry file

(f) TRANS01 This is the transmit file. In certain cases there may be more than one transmit file in which case files TRANS02, TRANS03...TRANS09 should be created as required. Sections 2.4 and 5.2.1 describe the use of multiple transmit files. Section A2.6 indicates the recommended sizes of transmit files

15 Obtain the IDE path catalogue as described in section 5.1. Select the Set Up path by typing Q into the field *Path*. The Set Up path is used to inform IDE of the sizes of the system files that have been created in step 14. Input to the Set Up path is described in section A2.6.2

A2.1 Loading the Development System

In order to load the Development System for each step defined above proceed as follows:

1 Switch on the lowest numbered video

2 Insert the terminal executive library diskette into drive 1 of the floppy disc drive

3 Select rotary position 3. Lift and then press down the *genres/activate* switch on the front of the terminal processor. The diskette menu select display will then appear

4 Press the blue numeric keys corresponding to the display number of T2AP11 and then press the SEND key

5 Insert the diskette containing the Development System program T2AD01 issued in the file DEVLIB. This diskette also contains the IDE software in the path library file IDE

6 Insert the application development PID (numbered 254) into the PID reader

7 Remove the terminal executive library diskette from drive 1 and insert in its place the diskette containing the appropriate path library file for the step to be undertaken. For example the file IDEDEVEL must be loaded when developing the Basic Data Entry Path (step 5) and IDEPATHS should be loaded when shrinking the developed data entry path from the file IDE to the operational path library file IDEPATHS (step 12)

8 Press the LOCAL INPUT key and type CE$^{\nabla}$D to configure the terminal executive for the Development System

9 Press the SEND key

10 (a) Press the LOCAL INPUT key and type OV to open all files on-line

(b) Press the SEND key

11 Press the LOCAL INPUT Key

12 Assign the Development System file as follows:

(a) Type GA *filename*,D,R

where *filename* is the name given to the Development System file, issued as DEVLIB

(b) Press the SEND key

13 Assign the path library file(s) that are to be used with the Development System package(s) that will be called for the current step. These should be assigned as follows:

(a) Press LOCAL INPUT then type GA *filename*,X,X

where *filename* is the name given to the path library file and X,X can be one of the following:

(i) U,W This is used when the path library file is to be created using the Create Library package or when it is to be used for development using the development package. It is also used to assign the path library file before using the Sizing package

(ii) U,R This is used when the path library file is to be copied to another file using the Tidy Library package

(iii) T,W This is used when the path library file is to be cleared using the Clear File package or it is to receive the contents of another path library file via the Tidy Library package

(b) Press the SEND key to assign the file

Example: For step 7 in the installation of IDE, the assigning of files should be as follows:

Press LOCAL INPUT

Type GA IDEDEVEL,T,W

Press SEND then LOCAL INPUT

Type GA IDE,U,R

Press SEND

14 Press the LOCAL INPUT key

15 Type AD

16 Press the SEND key. The Development System path catalogue display will now appear on the screen

A2.2 Using the Create Library package

1 To use the Create Library package the file that is to be created as a path library file must first have been set up on the diskette using the T2U$_x$I series command new file (NF)

2 The Development System must be loaded with the file to be created, assigned as follows:

Press LOCAL INPUT

Type GA *filename*,U,W

Press SEND

3 When the Development System path catalogue display appears on the video, type D into the field marked *Path* and press the TAB key. The create library display will appear naming the file that has been created

4 The creation is completed by pressing SEND

The development session should then be closed as specified in section 3.3.3.

A2.3 Using the Clear File package

This is used to clear a path library file before another path library file is tidied to it.

To use the Clear File package proceed as follows:

1 Load the Development System assigning the appropriate file as follows:

Press LOCAL INPUT

Type GA *filename*,T,W

Press SEND

Note: If the Tidy Library package is to be run straight after the Clear File the other file required by the Tidy Library package should be assigned when the Development System is loaded

2 When the Development System path catalogue appears, type E into the field marked *Path*

3 Press the TAB key. The Clear File display shown in Figure A2.1 will then appear

4 Press SEND to clear the file

A2.4 Using the Tidy Library package

This is used to copy and shrink a path library from one file to another.

Note: If the Tidy Library package is simply used to copy one file to another the files may still be used for further development. If however the shrink option has been used when copying a path library, the new file cannot be used for further development.

When the Tidy Library package is used after the Clear File package the Development System need not be re-loaded.

The Tidy Library package requires the file to be copied to be assigned

GA *filename*,U,R

and the file to which the copy is to be made

GA *filename*,T,W

If these files were not assigned when the Development System was loaded proceed as follows:

1 Press INTERLOCK together with CLEAR SCREEN. When the screen is cleared, continue

2 Press the LOCAL INPUT key

3 Assign the first file

4 Press the SEND key and repeat from 2 to assign the second file

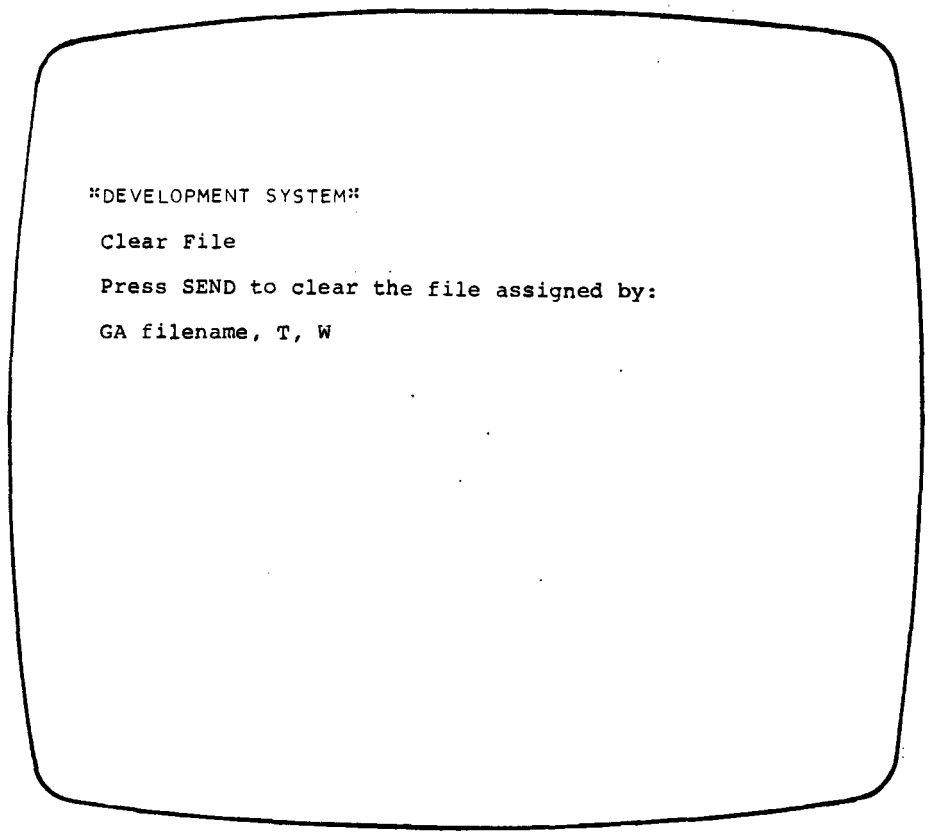


Figure A2.1 Clear file display

5 Press the LOCAL INPUT key

6 Type AD

7 Press the SEND key. The Development System path catalogue will now appear

Otherwise:

Press CTRL and 2 to display the Development System path catalogue

When the Development System path catalogue appears type B into the field *Path*. The Tidy Library display will then appear, as shown in Figure A2.2.

When the Tidy Library display appears, proceed as follows:

1 If the shrink option is to be used, type Y into the *Shrink?* field and press TAB, otherwise begin at step 2

2 If an individual path only is to be tidied, enter the identifier of the path into the *From Path* field and press TAB, otherwise perform step 4

3 Enter the new identifier of the path into the *To Path* field and press TAB

4 Press the SEND key

After the Tidy Library package has been used the development session must be closed as described in section 3.3.3.

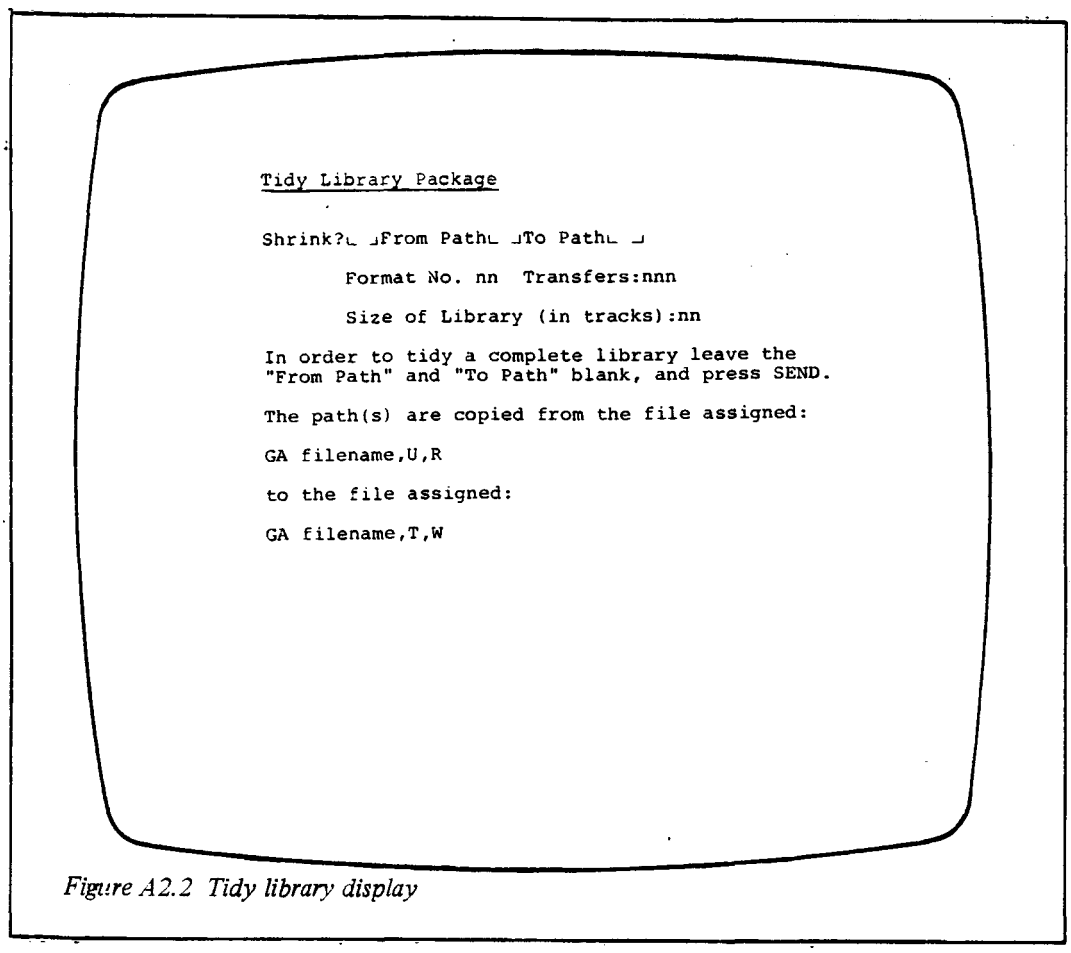


Figure A2.2 Tidy library display

A2.5 Using the Sizing package

Load the Development System assigning the path Library to be sized

Press LOCAL INPUT

Type GA *filename*,U,W

Press SEND

When the Development System path catalogue display appears type C into the *Path* field and press TAB. The sizing package display shown in Figure A2.3 will then appear.

When the Sizing Package display appears, proceed as follows:

1 Type D into the *Path Id* field. This is the identifier of the Basic Data Entry path for which sizing information is required

2 Press the TAB key. The relevant information will then be displayed

The total size for each screen format is given. The line containing the largest total value will be framed by two asterisk (*) characters.

The total size per screen format should be kept within a defined limit given with each release of IDE. An oversize screen should be reduced to this limit or below.

After using the Sizing package other Development System packages can be used by typing CTRL and 2 to return to the Development System path catalogue. Otherwise the development session should be closed as specified in section 3.3.3.

A2.6 Setting up the IDE system

A2.6.1 Creating the IDE system files

The IDE system files are created as described in steps at the beginning of this Appendix.

All the files must be created without indexes.

The following table lists the files and their sizes for both the single dual floppy disc system (two diskettes) and the two dual floppy disc system (four diskettes).

	SIZE IN TRACKS (maximum 72)	
Filename	*Two diskette system*	*Four diskette system*
IDEPATHS	50*	50*
OPERATOR	1	1
SPACNTRL	1	1
BATCHDET	5*	5*
IDECNTRL	4	4
ENT*n*	10*	36*
TRANS01	36*	72*

n is in the range 1 to 4 and indicates the number of the video with which each entry file (ENT*n*) is associated.

* indicates where the *size in tracks* is a suggestion only. For these files the sizes can be set to fit the installation's requirements within the limit of 72 tracks, that being a complete diskette.

The size of the path library file IDEPATHS is unlikely to exceed 50 tracks. Its final size depends on the number and size of the screen formats designed.

When step 12 is attempted, if IDEPATHS has been created too small to hold the developed path library, the Tidy Library package will fail and display an error message. In this case the diskette should be initialised again and IDEPATHS recreated larger.

The recommended size of BATCHDET is 5 tracks, this allows 130 batches to be known to the system at any one time. Each additional track allocated allows an extra 26 batches to be known to the system.

The size of each entry file should be determined for the most efficient use of the configuration. If fewer than the maximum number of videos are attached to the system then the entry files for the videos can be larger than suggested.

If one video were to be used considerably more than the others on the system the size of its entry file could be larger than for the other videos.

The size of the transmit file should be set to provide enough space to hold all the batches verified in the interval between transmission sessions.

In a four diskette system it is recommended that the transmit file occupies a full diskette. This allows multiple transmit files to be created. Multiple transmit files are used to continue data entry after the original transmit file is full, (see sections 2.4 and 5.2.1). To use these files during a data entry session they must be available before the data entry session starts and all the transmit files must be of the same size. They must be created with serial numbers increasing by one up to nine, that is TRANS02, TRANS03...TRANS09.

IDE system files can be spread among the available diskettes in the way most appropriate to the installation. The division is dependent on the estimated frequency of access to each file, on the speed of access desired and on the size of the files involved. The best method of organisation is one that satisfies the following conditions:

1 The diskette read/write head movement should be kept to a minimum

2 File access should be spread as evenly as possible between the disc drives

The table below indicates the recommended distribution of the files on the diskette volumes A, B, C and D.

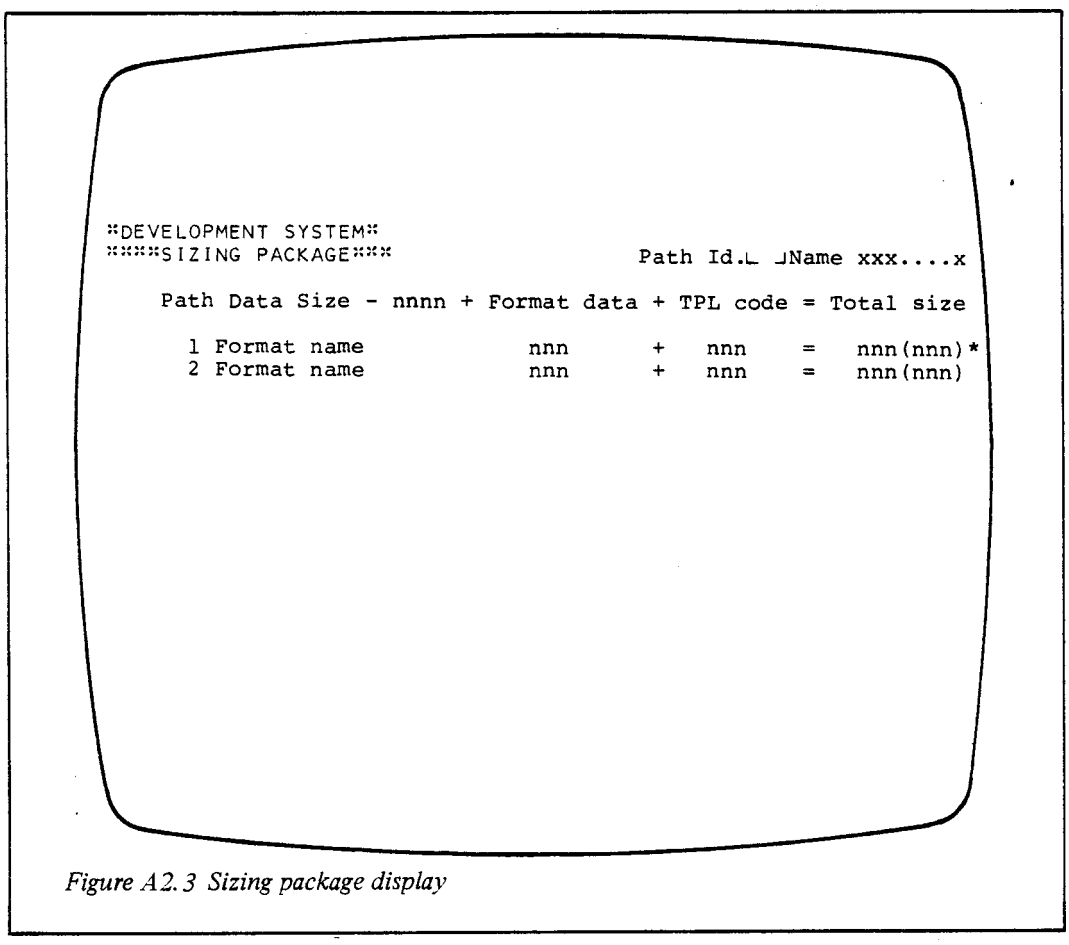


Figure A2.3 Sizing package display

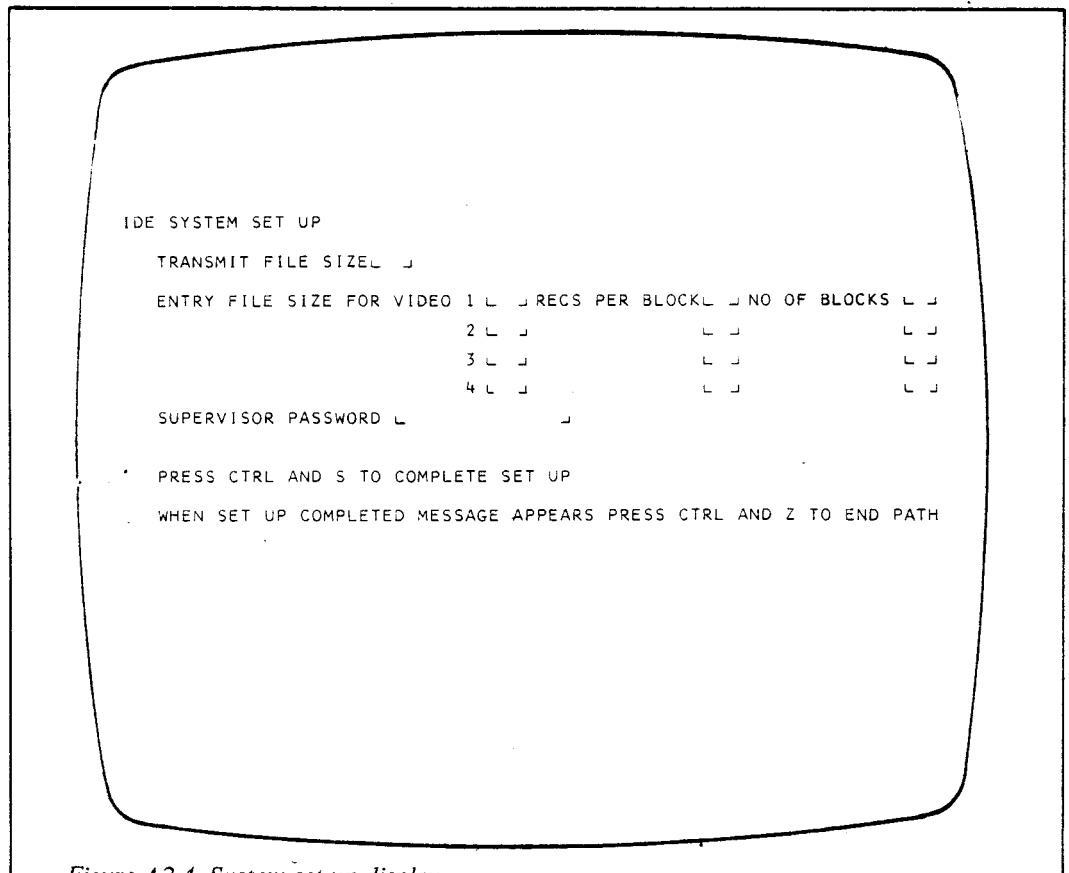


Figure A2.4 System set up display

Two diskette system diskettes		*Four diskette system diskettes*			
A	*B*	*A*	*B*	*C*	*D*
IDECNTRL	IDEPATHS	TRANS01	ENT1	ENT3	IDEPATHS
TRANS01	OPERATOR		ENT2	ENT4	OPERATOR
ENT1	SPACNTRL				SPACNTRL
ENT2	BATCHDET				BATCHDET
ENT3	ENT4				IDECNTRL

A2.6.2 Using the IDE Set Up path

When the files have been created the IDE system must be informed of the file sizes and of the initial supervisor password. This is done using the IDE Set Up path.

File sizes in the IDE system are specified as the number of records in the file. Each track contains 26 records.

In addition to the number of records the IDE system needs to know how the entry files are to be divided into blocks.

Each entry file is treated as a series of blocks of records with a maximum of 60 blocks per file. When space on the file is allocated to each new entry batch an integral number of blocks is allocated. So for efficient use of the file space available the number of records per block should be kept to a minimum.

For each entry file the IDE system requires the number of records per block and the number of blocks per file.

To use the Set up path proceed as follows:

1 Obtain the IDE path catalogue as described in section 5.1

2 Select the Set Up path by inputting Q. The set up screen display will appear as shown in Figure A2.4

3 Enter each field in the screen and press TAB

(a) TRANSMIT FILE SIZE Enter the number of records in this file

(b) For each entry file created complete:

(i) FILE SIZE The number of records in the file

(ii) The number of records per block

(iii) The number of blocks per file

Note: If fewer than four entry files have been created the three fields for the unused files should be specified as zero

(c) Enter the supervisor password. (Any eight characters)

4 Press CTRL and S. The message SET UP COMPLETE will appear

5 Press CTRL and Z to return the system display to the IDE path catalogue.

The error messages that may appear when using this path are as follows:

Message	*Meaning/operation action*
NOT NUMERIC	A field entered is not numeric. It should be corrected and TAB pressed before continuing
SIZE < RECS * BLOCKS	For an entry file the file size input is less than the size calculated by multiplying the number of records per block specified by the number of blocks specified. Correct one or more of these input fields (pressing TAB after each field) before continuing

SIZE NOT =REC * BLOCKS	For an entry file the size input is larger than the size calculated by multiplying the number of records per block specified by the number of blocks specified. This is a warning message only. The fields need not be corrected before continuing
SPACNTRL ERROR	This file is not on-line. It must be loaded before continuing
OPERATOR ERROR	This file is not on-line. It must be loaded before continuing

A2.7 Installing a new issue of the IDE software

When a new version of the IDE software is released it will be issued on a diskette in file IDE.

In order to copy the developed screen formats into the new issue of the IDE software proceed as follows:

1. Load the Development System with the files IDEDEVEL (containing the pre-shrunk current Basic Data Entry path) and IDE (the new issue)

2. Use the Tidy Library package to copy path D (the Basic Data Entry path) from IDEDEVEL to path E of the new issue of IDE

3. Use the Part Copy package described in *TPL Application Programming* to copy each format from path E to the new issue of path D in file IDE

4. Using the control keys described in section 3.3.4 call up all parts of the hierarchy shown in Figure 3.6 and *store* all the screens

5. Delete path E as described in section 3.3.4

6. Re-load the Development System and run the Clear File package on IDEPATHS. Then use the Tidy Library package to shrink the new version of IDE from file IDE to IDEPATHS

Action key	*Effect*
A	Abandons the current function in data entry mode
C	In the data entry path this closes a batch. Depending on when it is used it may or may not perform batch totalling
D	In the data entry path under modify mode this deletes the information displayed on the screen from the diskette file
E	In the data entry path this enables the operator to continue input after an unsuccessful attempt has been made to close a batch. In the supervisor path this action key will erase a batch when used with function 6. It also erases COPY information when used with supervisor function 8
F	Load a new format
M	In the data entry path this will display the mode selection screen. In the supervisor path it will display the supervisor function list
P	In the supervisor batch status function this displays the password
S	Skips records in modify mode. In the set-up path it completes the set-up. In the supervisor function this action key will skip a batch when group COPY information is being entered
Z	This will display the IDE path catalogue. Depending on when it is used it may or may not abort the current operation

The format of the card depends on the type of file to which the batch is to be written.

When the batch is to be written to a user file the format is as follows:

Columns	*Content*
1 to 12	Filename
13 to 16	Generation number
17 to 20	Retention period
21 to 26	Unit number for extension
27	Overwrite/Append indicator
28	File full action
29	Record type. This must be U

When the batch is to be written to a spoolfile document the card format is as follows:

Columns	*Contents*
1 to 12	Document name
13 to 16	Terminator
17 to 28	Unused
29	Record type. This must be S

The contents of the card are detailed in section 5.2.8.

The majority of IDE error messages are fully detailed in the text of this publication. Where they are not the errors fall into two categories: errors that are self-explanatory and errors that indicate a software failure.

Software failure error messages have the following format:

ERROR▽*x*▽*nn*▽*mm*▽XX

where *x* is the path name:

D for Basic Data Entry path

S for Supervisor path

G for Global path

nn is the format number

mm is the action routine number

When a software error of this type occurs ICL should be consulated.

The other IDE errors that may occur are listed below:

Validation error messages

FIELD MUST BE PRESENT
NOT ALPHABETIC
NOT NUMERIC
NOT ALPHANUMERIC
NOT FULL NUMERIC
NOT INTEGER
INVALID DATA
SPACES PRESENT
VALUE OUT OF RANGE
INVALID DATE

Other error messages that may occur during data entry

SESSION NOT STARTED
BATCH NEARLY FULL
NO ROOM FOR THIS DOCUMENT - CLOSE THIS BATCH
INVALID FORMAT

Error messages on closing a batch

TOTALS MISMATCH
TOTALS MISMATCH PRESS CTRL + C TO IGNORE
BATCH CANNOT BE CLOSED IN VERIFY MODE

Error messages relating to the modify mode header

BATCH IN USE
NO SUCH BATCH
INCORRECT BATCH STATUS
INCORRECT PASSWORD
LOAD FILE ENT*n*
ENTER BATCHNAME

Error message on mode selection

INVALID MODE

Error messages relating to the entry mode header

BATCHNAME IN USE
ENTER NO OF DOCUMENTS
NO SPACE ON FILE ENT*n*
LOAD FILE ENT*n*

INVALID FORMAT
ENTER BATCHNAME

Error messages relating to the verify mode header

BATCH IN USE
NO SUCH BATCH
INCORRECT BATCH STATUS
INCORRECT PASSWORD
TRANSMIT FILE FULL
LOAD FILE ENT*n*
ENTER BATCHNAME

Error messages in the supervisor path

SUPERVISOR PATH IN USE PRESS CTRL + Z TO END USE OF THIS PATH
LOAD FILE SPACNTRL
INCORRECT PASSWORD
INVALID FUNCTION
START SESSION

Error messages when starting a session

LOAD FILE IDECNTRL
IDECNTRL NOT INITIALISED PRESS CTRL + Z TO END USE OF THIS PATH
LOAD FILE TRANS01
TRANS01 NOT INITIALISED PRESS CTRL + Z TO END USE OF THIS PATH
LOAD FILE BATCHDET
LOAD FILE OPERATOR
ENTER DATE
INVALID DATE
INVALID TIME

Errors in Function 2

ALL 9 FILES ALREADY USED
LOAD FILE TRANS*nn*
TRANS*nn* NOT INITIALISED PRESS CTRL + Z TO END USE OF THIS PATH

Errors in Function 3

LOAD FILE SPACNTRL
LOAD FILE IDECNTRL
LOAD FILE OPERATOR
LOAD FILE TRANS*nn*
LOAD FILE BATCHDET
USE START OF SESSION
ERASE FILES AND USE START OF SESSION

Errors in Function 4

ENTER BATCHNAME
NO SUCH BATCH
COPY DATA PRESENT

Errors in Function 5

INVALID DATA

Errors in Function 6

ENTER BATCHNAME
NO SUCH BATCH
INCORRECT PASSWORD
BATCH IN USE

Errors in Function 8

NO BATCH BATCH
BATCH IN USE
NO COPY DATA PRESENT

Errors in Function 9

IDECNTRL FILE FULL
INVALID DATA
NO SUCH BATCH
ENTER SPOOL OR USER
COPY DATA PRESENT
BATCH IN USE

Errors in Function 10

IDECNTRL FILE FULL
ENTER SPOOL OR USER
INVALID DATA

Errors in Function 12

BATCH IN USE
NO SUCH BATCH
INCORRECT BATCH PASSWORD
BATCH ALREADY EXISTS
ENTER BATCHNAME

Errors in Function 13

NO SUCH BATCH
INCORRECT BATCH STATUS
INCORRECT PASSWORD
INPUT FILE NOT ONLINE
BATCH NOT ON INPUT FILE
TRANSMIT FILE FULL
ENTER BATCHNAME

Errors in Function 14

OPERATOR IN USE
INVALID ACTION
OPERATOR ALREADY EXISTS
NO SPACE ON FILE
NO SUCH OPERATOR

Errors in Function 15

NO SUCH OPERATOR

Errors in Function 16

TRANSMIT FILE FULL
INVALID ACTION

Error messages in the Set Up path

NOT NUMERIC
SIZE < RECS * BLOCKS
SIZE NOT = REC * BLOCKS
SPACNTRL ERROR
OPERATOR ERROR

Error messages for program #XIDE

: *batchname*
: INVALID REC TYPE *x*

: DA ERROR *nnnn* *filename*

: EOF ON CR*n*

: UNPAIRED RECORD ON
USER DATA FILE *n*

: COPY INFORMATION MISSING FOR BATCH *batchname*
: LOAD ACCEPT DATA ON CR*n*

: CARD INVALID
: LOAD ACCEPT DATA ON CR*n*

: ONLINE *filename*

: *filename* FGN*n* NOT ONLINE
: COBOL ACCEPT SEE JOB SHEET

: INVALID REPLY-TRY AGAIN
: COBOL ACCEPT SEE JOB SHEET

: *batchname* BATCH HEADER MISSING
: PRINT 0-3000 OF CORE

: FILE FULL *filename*

SD PERI COMMANDS NOT SUPPORTED

ITEM INDEX FULL

INSUFFICIENT CORE FOR ASSIGNMENT

INSUFFICIENT SPACE ON FILE FOR REQUEST

CARD TRANSFER ERROR

ERROR IN FUNCTION CODE

FAILURE ON CLOSING DOCUMENT

Error messages of the form

Error *xx*

may be displayed when inputting commands to the terminal executives T2AP31 and T2RB31 or during transmission under T2RB31. For a list of these errors see *Software Operating*.