9.5 Organizational Instructions for One Inch Tape

A number of organizational instructions are provided to cope with special situations which will arise in some magnetic tape programs. Many of the operations which these instructions perform are usually required near the beginning or end of the program, and they are then best left to the job description or the 1117 (end program) instruction. One exception to this rule is the instruction 1017 (Free Tape), which should be used before the 1117 instruction if the information on any titled tape is not required again.

Programs frequently require to use magnetic tapes as working space, without wanting to keep them after the job has been run. Such magnetic tapes should be requested under a heading TAPE COMMON in the job description.

For a long job, it may be desired to restart the job after a machine failure severe enough for common tapes to be lost; in such a case tapes should be requested under TAPE NEW. In exceptional circumstances a title may be written to a common tape, which is then kept by the user after the job has ended; the operator will be notified, and the Atlas installation may take action to discourage a user from doing this at all often.

The title stored in section 0 of the tape is referred to by extracodes 1014 and 1015 (Write Title & Search for Block 1, and Read Title or Tape Number). These refer by its main store address to a copy of the tape title, stored as one record; the 6-bit characters are packed 8 to a word and the last character is binary zero.

9.5.1 Mount Instructions

If a program requires to use magnetic tape, its job description must indicate the number of magnetic tape mechanisms required. Normally this is done by listing the magnetic tapes which are required to be mounted on these mechanisms initially: the Supervisor will then ensure that these tapes are mounted before the program is entered. The details of how the job description is prepared are given in Chapter 10.

If further magnetic tapes are required after a program has been entered, they should be listed in the job description and may then be called in by obeying "mount" instructions. However, the total number of mechanisms in use at any one time must not exceed the number reserved in the job description. A mount instruction should, if possible, be obeyed at least 2 minutes before the tape to which it refers is required; otherwise the tape may not be ready in time and the program will have to wait. Note that the program will be monitored if it calls for a new tape to be mounted at a time when none of its reserved mechanisms has been made free. If there is a spare tape mechanism, the tape may be mounted on it by the operator before the program calls for it.

The tape reference letter referred to in the mount extracodes is a letter associated with the tape in the job description (see chapter 10).

The mount instructions are as follows:-

1010 Mount

Allocate the number Ba to the tape whose tape reference letter is in the 6 least significant bits of n, in internal code. If this tape is not already available, instruct the operator to mount it on any available tape mechanism. If the tape is in the TAPE category, check its title; if in the TAPE NEW category, write into the section 0 the title given in the job description: if in the TAPE COMMON category, leave the tape untitled.

1011 Mount Free

Allocate the number Ba to a free tape, If no free tape is available, instruct the operator to mount one on any available tape mechanism.

This extracode should only be used in exceptional circumstances; normally all tapes required should be listed in the job description. The operator will be informed whenever this extracode is used, and an installation may take action if it is used overmuch.

1007 Mount Next Reel of File

Allocate the number n to the next reel of file Ba. If this tape is not mounted, instruct the operator to mount it on any available mechanism.

The following mount instructions refer to logical channel number (K = 0 to 3) of a given program. These logical channel numbers do not each refer to a fixed magnetic tape channel: they are merely a device to enable the program to separate on to different channels the magnetic tapes that it requires to operate simultaneously. On an installation with only one tape mechanism on each of the eight tape channels, there is no advantage in specifying channel numbers. On larger installations, the tape mechanisms are grouped together on pairs of channels, and the logical channel numbers then refer to these pairs.

A program's referring to logical channel number K has the following effect. If no channel has been previously designated logical channel K of the program, the new tape is mounted, if possible, on a channel different to any which has been previously designated a logical channel of the program; that channel is then designated logical channel K of the given program. If a channel has been previously designated logical channel K of the program, then the new tape is mounted on the same channel, if possible. A channel may also be designated logical channel K of the program by the Job Description; see chapter 10.

1012 Mount on Channel K

Allocate the number Ba to the tape whose tape reference letter is in the 6 least significant bits of n, in internal code. If this tape is not already available, instruct the operator to mount it if possible, on channel K of this program, where K is the most significant octal digit of n. If the Tape is in the TAPE category, check its title; if in the TAPE NEW category, write into section 0 the title given in the job description; if in the TAPE COMMON category, leave the tape untitled.

1013 Mount Free on Channel K

Allocate the number Ba to a free tape. If no free tape is available, instruct the operator to mount one, if possible, on channel K of this program, where K is the most significant octal digit of n.

Note that this extracode, as 1011, should be used only in very exceptional circumstances; normally all tapes required should be listed in the job description. The operator will be informed whenever this extracode is used and an installation may take action if it is used overmuch.

9.5.2 Other organizational Extracodes

1014 Write Title & Search for Block 1

Write to section 0 of tape Ba the title stored in location S onwards, overwriting any title that may be there. Inform the operator that this has been done.

1015 Read Title or Tape Number

If S is even (least significant bit is 0), then store in locations S onwards the title of tape Ba, i.e. that currently in section 0.

If S is odd (least significant bit is 1), then store in location S the tape serial number of tape Ba.

In 1014 and 1015 S is taken as a half word address. The tape serial number stored by 1015 will be in internal code, left justified, 8 characters in length.

1016 Unload and Store

Rewind tape Ba and disengage the tape mechanism on which it is mounted. Instruct the operator to remove the tape, ensure that the correct title is written on the spool, and store it for later use. If $n \neq 0$, the number of tape mechanisms reserved for the program is reduced by one.

1017 Free Tape

Overwrite the title on section 0 of tape Ba and return the tape to the Supervisor as a Free tape for general use. If $n \neq 0$ the number of tape mechanisms reserved for the program is reduced by one.

1020 Release Tape

Delete tape Ba from the allocation of this program and make it available for another program, which must call for it by its correct title. The tape is not freed and is not normally disengaged. If $n \neq 0$, the number of tape mechanisms reserved for the program is reduced by one.

1021 Free Mechanisms

Reduce by n the number of tape mechanisms reserved for use by the program.

1022 Re-number

Allocate the number n to the tape which was previously referred to in this program as tape number ba.
This enables a tape to be given a different number during a subroutine, and then to have its original number restored at the end of the subroutine. Note that in this instruction the tape number is ba and not Ba.

1023 How Long?

s' = Number of 512-word sections available on tape Ba (excluding section 0). The number of sections available on a standard-length tape is 4999, but this instruction may be of value if shorter tapes are used. The tape will be rewound.

1024 Where am I?

$s' = A$; $s^{*\prime} = W$ $(0 \leq W \leq 511)$.
This instruction places the address of the next section (going forwards) on tape Ba in the first half-word of the specified full-word address. After variable length transfers, the second half-word contains the position in the section of the next word to be used. After block transfers, the second half-word is zero.

9.6 Specification of the Atlas One Inch Tape System

9.6.1 Control

An Atlas installation may have as few as 8 one inch magnetic tape mechanisms, or as many as 32. Each mechanism is connected via one of eight channels which can operate simultaneously, each channel controlling one read, write, or search operation. Wind and rewind operations are autonomous and need the channel only to initiate and, if required, to terminate them.

The layout of the control system depends on the individual installation. When there are only 8 mechanisms, each mechanism has its own control channel. When there are more than 8 mechanisms, the 8 channels are grouped into 4 pairs and some or all of these pairs are adapted to control up to 8 mechanisms; any two of the mechanisms on one such pair of channels can then operate simultaneously at any one time.

9.6.2 The Tape Layout

The tape mechanism is the Ampex TM2, using one inch wide magnetic tape. There are 16 tracks across the tape, used as follows:

12 information tracks

2 clock tracks

1 block-marker track

1 reference-marker track (for Tape Addressing only)

Information is stored on tape in blocks or sections of 512 48-bit words, followed by a 24-bit checksum. Each section is preceded by a leading block marker and a section address, and terminated by a trailing block marker and a zero address. Tapes are tested and pre-addressed by a special run on the machine before they are put into use, and the fixed position of addresses permits selective overwriting of sections. Checksums are of 24 bits with end-around carry; they are used to check the accuracy of all reading and writing operations.

A 48-bit word is represented by four lateral stripes of 12 information bits, and a checksum by two stripes. Each 512-word section of information contains 2050 stripes and has an average length of 5.46 inches, with a gap of 2.3 inches between sections. Tapes are 3600 feet long and hold 5000 sections, or $2\frac{1}{2}$ million 48-bit words.

9.6.3 Performance

The normal tape speed is about 120 inches per second and there are 375 binary digits per inch on each track. This gives an instantaneous transfer rate of 90,000 6-bit characters per second, or one 48-bit Atlas word every 89 microseconds. Allowing for the gaps between sections, the effective transfer rate is about 64,000 characters per second. This is equivalent to one 512-word section every 64 milliseconds, or one word every 125 microseconds, on average. There are also fast wind and rewind operations at about 180 inches per second, and these are used for long searches along the tape.

There are independent write and read heads, separated by a gap of about .39 inches. When not operating, the tape stops with the read head roughly midway between sections, ready to read the next section address. It is possible to read when the tape is moving either in the forward or reverse direction, but writing is only possible when the tape is moving forwards.

9.6.4 Safeguards

A program is held up if it attempts to read from or write to a block of store which is involved in a magnetic tape transfer. The Supervisor may then enter another program until the transfer is completed.

If a magnetic tape block transfer cannot be initiated when it is requested, it is placed in a queue. If the queue is already full the program is held up.

A write-permit ring must be fitted to a reel of tape before that reel can be written on. Tapes containing permanent information will not have such a ring. A write-inhibit switch is also provided on each mechanism, which the operator can use to isolate the tape. It is only possible to write on a tape when the write-permit ring is on and the write current is switched on.

The address of the relevant section on tape is checked before all reading and writing operations, to make sure that the correct section is used. The information in each section is checked by means of a 24-bit checksum at the end of the section: this checksum is used to detect faulty writing or reading, which cause the operation to be repeated under the control of the Supervisor.

When a magnetic tape has useful information on it, a descriptive title of that information is stored in block 0 of the tape. This is in addition to the tape serial number, which is permanently associated with the tape and is unalterable by the user. This tape title can be up to 80 characters long, though the Supervisor prints out only the first 30 characters in operator messages. It is stored on tape in Atlas Internal Code with tab. and multiple spaces (including tab.) replaced by a single space; initial spaces, tabs, full stops, and commas are ignored. Shifts are ignored throughout the title. (see Chapter 10.)

9.7 Orion Tapes

It is possible to read into Atlas tapes prepared on the I.C.T. Orion Computer. This is accomplished by the use of the following two extracodes. P is a block address, and K an octal fraction, of the form P:0.K.

1046 Read Orion tape Ba forwards

A check is first made that tape Ba is in fact an Orion tape (if it is, zero will have been read from the first bit of the first block when the tape was mounted). Then, reading forwards, the next section is read into store blocks P onwards. K + 1 blocks will be reserved for the transfer - if this is not sufficient, the program will be monitored. Up to 4096 words may be transferred, but there is no automatic indication of the number of words actually read.

1048 Read Orion tape Ba backwards

This is very similar to 1046, the difference being that the first word transferred is placed in the last word of block P + K and the tape is read backwards.

Some of the organizational instructions listed in section 9.5 may be used with Orion tapes. These are given below.

1010 Mount
1007 Mount Next Reel of File
1012 Mount on Channel K
1015 Read Title or Tape Number
1016 Unload and Store
1020 Release Tape
1021 Free Mechanisms
1022 Renumber
1023 How Long?
1024 Where am I?

The section number only will be given; the second half word will be zero.

The title read by the 1015 instruction will consist of up to ten words separated by /, the words containing up to eight characters. The characters will be letters, digits or full stop.

In the Job description (chapter 10), Orion tapes must be listed under TAPE headings, and the title must be given in the form described for the 1015 extracode.

Chapter 10

PREPARING A COMPLETE PROGRAM

This chapter explains the way in which a program, and the input information it uses are prepared for running on Atlas.

10.1 Atlas Jobs

Each run of a program on Atlas is known as a job; it may range from a small job, for which there is no data outside the program itself, to a large job requiring several batches of data, possibly arriving on different media, e.g. punched cards and paper tape.

The various parts of a job may be submitted separately to the computer each on one deck of cards or length of paper tape, or two or more parts may be combined on a single deck or tape. In any case each part must be properly identified for the computer and for this purpose the concept of a 'document' has been introduced.

10.2 Documents

A document is a self-contained section of information presented continuously to the computer through one input channel (but see also 10.10.3). Typical examples of a document are a collection of data on a length of paper tape or the program itself.

Each document carries at its head suitable identifying information as detailed in 10.3.1. The end of a document is indicated by an 'end of document marker' which usually consists of the characters ***Z on a new line, or a 7, 8 punching in the first column of a new card which follows the last record of information of the document.

By means of the identifying information the Supervisor prepares a list of documents as they are accepted in the store and it also keeps a list of jobs for which further documents are awaited. A job may require several documents and only when all these have been supplied can execution begin. The Supervisor therefore checks the appearance of each document; when all have been entered work on the job may commence. Documents for a job may thus be fed to the computer in any order.

10.3 Document Headings and Titles

Every document is preceded by the identifying information mentioned above. This consists of a heading and the title of the document.

10.3.1 Headings

The heading indicates which type of document follows and must be one of a standard list. The most common types of heading are as follows:-

a) COMPILER (which is followed on the same line or card, after one or more spaces, by the name of a program language). The document following this heading is a program in the stated language. Available languages include Atlas Basic Language (ABL), Extended Mercury Autocode (EMA), ALGOL and HARTRAN (for Fortran).

For an Atlas Basic Language Program the heading will be

COMPILER ABL

b) DATA
The document following consists of data required by a program.

c) JOB
The following document is a request to the computer to execute a job and gives relevant facts about it.

The last type of document is called a 'job description'. It gives, for example, a list of all other documents required for the job and a list of output streams the program will produce. It is described in detail in sections 10.4, 10.5 and 10.6 below.

10.3.2 Titles

The title of a document consists of one line (or one punched card) immediately following the heading. It may be composed of any combination of characters obeying the rules of section 10.3.3 below. The prime consideration is, however, that it should be unique among all the documents stored in the computer at the same time. This is obviously made essential by the time-sharing facilities of Atlas, to avoid confusion between documents intended for different jobs.

A document will thus usually take one of three forms exemplified by the following, the second line of each document being its title:-

JOB F6479,SMITH;I.C.T. ------ ------ (Job Description) ------ ------ * * * Z	COMPILER ABL F1, SURVEY PROGRAM ------ ------ (Program) ------ ------ * * * Z	DATA F6479 BEETLE SURVEY ------ ------ (Data) ------ ------ * * * Z
Job Description	Program Document	Data Document

10.3.3 Rules for Title Preparation

Besides being unique document titles must obey the following rules:-

a) The title must begin with an identification of the person or organization which originated the document: normally this will take the form of an account number or name. For example, documents prepared for the I.C.T. Atlas Computing Service are identified by a letter F followed by an account number.

b) The number of characters must not exceed 80.

c) 'Backspace' must not be used.

d) A title must not contain three successive asterisks (***).

e) Titles must not begin with the word END or the word TAPE.

Furthermore:

a) The characters of a title are read in and stored in Supervisor records in Atlas internal code in the normal way, but the shift characters, 04, 05, 06, 07 are subsequently removed. This means for example that on the Flexowriter the titles

[Tape]

and

1 TAPE 3

are identical. For the same reason a length of run-out appearing on tape in the midst of a title would not become part of the title.

b) Any number of consecutive spaces and tabs are stored as one space.

c) Erases do not become part of the title.

d) Initial commas, spaces and full stops are ignored.

Documents used in the same job need not have related titles.

The job title itself normally contains the name and abbreviated address for the return of the results, but this is not necessary in the titles of data and program documents.

If necessary the title of a data document or job description (but not a program document) may be on the same line, or card, as the heading, provided sufficient room remains to accommodate all the title on that line or card. In this case the heading and title must be separated by 'comma', 'space' (one or more), or 'tab'.

10.4 The Input and Output Sections of the Job Description

After the heading and title a job description is divided into various sections each one describing a particular aspect of the job, e.g. input documents, store used and so on. These sections may be assembled in any order and are dealt with individually below in sections 10.4.1, 10.4.2, 10.6, 10.7 and 10.8.

10.4.1 The Input Section

This section begins with the word

INPUT

which is followed by the titles of the data documents used in the job, each preceded by the number by which the program refers to them. These numbers must be in the range 0 to 15. The program document itself usually is given number 0, but is in fact always taken to be the lowest numbered data document. Thus, if a program operates on two data documents which it refers to as inputs one and two respectively, the job description would contain

```
INPUT
0 (Title of Program)
1 (Title of Data 1)
2 (Title of Data 2)
```

To take a concrete example:-

```
INPUT
0 F1, SURVEY PROGRAM
1 F6479/2 BEETLE SURVEY DATA/62
2 F6479 BEETLE POPULATION 1961
```

The data document "F6479/2 BEETLE SURVEY DATA/62" could then be selected by the programmer by the instruction

```
1050        0        0        1
```

Data may be placed on the same tape as the program, where it becomes a part of the lowest numbered input stream. With an ABL program such data must come between the enter directive and the end of document marker that terminated the program stream. If the 1050 extracode is not used to select a given input stream, the lowest numbered stream is assumed and one obtains the data which followed the program.

The input section may also contain a reference to a magnetic tape on which an especially large document has previously been stored. This way of handling large amounts of input is explained in section 10.12.1.

10.4.2 The Output Section

This section of the job description specifies the type of peripherals to be used for output. Possible types of equipment are:-

```
LINE PRINTER
SEVEN HOLE PUNCH
CARDS
FIVE HOLE PUNCH
ANY
TAPE
```

Here CARDS means the card punch. Each Atlas installation will specify which types of equipment may be used for output ANY e.g. ANY may produce output on LINE PRINTER or SEVEN HOLE PUNCH. TAPE is used when a private magnetic tape is called for to hold an especially large amount of output (see 10.12.4).

The output section begins with the word

```
OUTPUT
```

which is followed by a list of output mechanisms, each preceded by a programmer's number in the range 0 to 15. For example one might have

```
OUTPUT
2 ANY
1 CARDS
0 LINE PRINTER
```

In this case, in order to send output to the card punch, the programmer would first have to select this form of output by the instruction

```
1060        0        0        1
```

A request will be made to the operator to mount special stationery for a given output stream if an asterisk is placed in front of the word LINE PRINTER. Thus if output stream 3 is to be printed on special stationery, the output section should contain

```
*LINE PRINTER m LINES
```

The type of equipment should normally be followed by a limit on the amount of output, specified as so many lines. One line is the output produced by one use of the 1065 (end current record) or the 1067 (output one record) extracodes. Thus one line means one printed or blank line on 5 or 7 track tape and on the line printer, or one punched card. One writes

```
OUTPUT
0 LINE PRINTER m LINES
```

The maximum amount of output may also be specified as n BLOCKS. A block contains 4096 characters. The number of characters allowed for must in general be larger than the number actually printed or punched. On the average each line output to punched paper tape or line printer requires an additional six characters (the maximum possible is 8) to be allowed for. Exactly 8 additional characters must be allowed for to punch one card (making 88 all told). Furthermore each use of 1065 (end this record) to produce a blank line generates 8 characters to be held in the output well.

If the number of blocks is omitted, one block only is allotted, and if the whole output section is omitted

```
OUTPUT
0 ANY 1 BLOCK
```

is understood.

10.4.3 Output : General Notes

a) Output 0 Output 0 is used by the Supervisor and some compilers, but is still available for normal output from the program. It is on this output that such information as the number of instructions done in compiling and executing the program, the number of store blocks in use when the program ended, the number of blocks accumulated for each 'stream' and other such items are printed. It is also used for fault information if the program goes wrong.

If no output 0 is mentioned in the output section of the job description

```
0 ANY 1 BLOCK
```

is assumed.

Similarly if no output stream is selected by the 1060 'select output n' extracode, any subsequent output will go to output 0.

b) Atlas can readily accept two or more streams of output from a program for the same type of equipment, even though only one such equipment may exist. The streams are accumulated independently within the computer and eventually output one after another.

In fact all output is accumulated and none will be printed until all computing has ceased unless the extracode

```
1071 Break output n
```

is used. In this case all of the output stream n accumulated so far will be sent to the peripheral.

In either case the output information with programmer's number n will always be preceded by

```
Supervisory Number / Date.    Time
OUTPUT n
(Title of Job)
```

The last line output gives the number of blocks sent to that output.

10.5 A Complete Job Description

We are now in a position to give an example of a complete job description and for the sake of illustration we include the documents of the job.

```
JOB
F64,  J. Smith, I.C.T. London, METALS        Job Title

INPUT
0 F64,    ANALYSIS PROGRAM                   Program Title
1 F64/A,  IRON CONTENT                       Data Tapes
2 F64/B,  COPPER CONTENT

OUTPUT
0 LINE PRINTER                               output
1 SEVEN HOLE PUNCH 3 BLOCKS                  streams
* * * Z                                      End-of-tape marker
```

Job Description

```
COMPILER ABL
F64, ANALYSIS PROGRAM
------
(Program)
------
------
* * * Z
```

Program Document

```
DATA
F64/A, IRON CONTENT
------
(Data)
------
------
* * * Z
```

Data 1

```
DATA
F64/B, COPPER CONTENT
------
(Data)
------
------
* * * Z
```

Data 2

10.6 The Magnetic Tape Section of the Job Description

Magnetic tapes are used with Atlas in two ways. Firstly, they are used by the Supervisor for such purposes as storing input and output. These are called System Tapes, and under normal circumstances need not concern the programmer at all. Their operation is quite automatic (see also section 10.12.3.)

Secondly, the programmer may use magnetic tapes in his program either:

a) For private input and output purposes

or

b) by magnetic tape extracodes.

The use of magnetic tapes for private input and output purposes will normally only be necessary if there is a very large amount of input or output. Full details of the way in which such tapes are employed are given in section 10.12.

The most common use of magnetic tapes is by extracodes within the program. The tapes required may be mounted on a tape mechanism (a 'deck') before running the program or during the actual execution.

The tapes mounted before the job begins must be listed in the job description. Normally the tapes that are to be mounted while the program is running are also listed in the job description. However, in programs which require a not-easily-predicted number of tapes it is possible to get tapes mounted which are not listed in the job description. (Sections 10.6.1. and 10.6.2. show how to list tapes in the job description. The extracodes for mounting tapes are described in Chapter 9.)

Full information on magnetic tapes can be found in chapter 9. However, for the sake of completeness some of the relevant facts are repeated here.

a) Information is stored on magnetic tapes in blocks of 512 48-bit words.

b) The first block is known as block 0, and is not available to the programmer. Block 0 contains the serial number of the tape, and the title if the tape has one. The title of the tape must obey the rules of document titles given in 10.3.2 but also, if the tape serial number is not listed in the job description along with the tape title, the first 30 characters of the title must identify the tape uniquely. Only those 30 characters are printed by the Supervisor when calling for the tape, but up to 80 characters are stored and checked.

c) In preparation for use each tape is mounted on a tape deck by an operator who receives instructions for the purpose from the Supervisor. When mounted, each tape is positioned by the Supervisor at the beginning of block 1, the first block available for storing information.

10.6.2 Single Tapes

Each tape required for a job is specified in the job description by 2 lines of printing:- a heading, and a description. The heading is one of three:-

TAPE - a tape belonging to the user and already having a title;

TAPE NEW - An untitled tape, not previously belonging to the user, to be titled and kept when the job is over;

TAPE COMMON - an untitled working tape to be retained by the system when the job is over.

The description consists basically of the programmer's number (in the range 0 to 99), the tape serial number (preceded by *) and the title of the tape:

21 *F3699 F1000, LONDON SALES, 1963

This format applies with headings TAPE and TAPE NEW. That for TAPE COMMON is described later in this section.

The programmer's number, 21 in the above example, is the number by which the tape will be referred to in the program. If the tape is not required to be mounted before the job begins, but rather will be called for in the course of execution by a "mount" instruction (see Chapter 9), the programmer's number must be replaced by a "tape reference letter" consisting of a single letter:

D *F3700 F1000, LIVERPOOL SALES, 1962

The mount instruction will then refer to the tape by its tape reference letter and assign to it a programmer's number. In the example just given the tape title will consist of the 28 characters starting with F1000 and ending with 1962 (i.e., the two double spaces will have been replaced by single space).

The tape serial number, F3699 in the first example, is permanently associated with the tape. If possible, the Supervisor will call for the tape by its serial number, under which the tapes are filed; the tape number should therefore be included when possible even when its inclusion is not made obligatory by a particular Atlas installation. If it is omitted, the * is omitted also.

Thus in the example

19 T9824, WOLVERHAMPTON SALES; 1962

the title is the sequence of 31 characters starting with T and ending with 2.

The title in the description is the complete title stored in block 0 of the tape. Under a TAPE heading, this will be used to check that the correct tape has been mounted; under TAPE NEW, the title will be written to block

0 when the tape is mounted. In either case if the tape serial number is absent the Supervisor refers to the tape by the first 30 characters of its title.

The description for TAPE COMMON consists of a programmer's number or tape reference letter only.

TAPE COMMON

36

All the programmer's numbers and tape references in the job description must, of course, be different. A programmer's number must lie in the range 0 to 99 inclusive; a tape reference can be any letter.

The description may refer to logical channel K of the program; the effect is the same as in extracodes 1012 and 1013. This is done by adding .K to the programmer's number.

Thus:

21.3 *F3699 F1000 LONDON SALES

would request that the tape be mounted on a 'channel' which can then be referred to in the program as channel 3. Tapes on different channels can be written to or read from simultaneously. The extracodes 1016 and 1012/1013 allow further tapes to be mounted on the same channel. (see Chapter 9.)

If a tape in the TAPE category requires file protection, i.e. no write permit ring and/or no write current, an asterisk should be written immediately before the description:

TAPE

*1 *F3002 F1002 PARIS SALES

With file protection block 0 cannot be written to, so it will not be possible to change the title of such a tape. The program will be monitored if the 1014 (change title) extracode is used on a tape with file protection.

10.6.2 Files

A collection of information may extend over several tapes although the programmer may wish to treat it as a single unit. Such a collection of tapes is called a file and only one of the tapes needs to be mounted at any one time. Each tape of a file is specified in the Job Description by a modified tape heading as follows:

TAPE/m

n (tape serial number) (Title of tape)

where m is the number of the tape within the file, counting from 1 upwards. The programmer's number n will be the same for all m of this file. The final tape of the file is entered as

TAPE/m END

n (tape serial number) (Title of tape)

For example, suppose a file of information extends over three magnetic tapes which have respectively as serial numbers and titles in block 0

```
Q1432   F1012   BIRMINGHAM SALES
Q1003   F1012   LONDON SALES
Q1100   F1012   MANCHESTER SALES
```

These would be referred to in the Job Description as

```
TAPE/1
3 *Q1432 F1012   BIRMINGHAM SALES
TAPE/2
3 *Q1003 F1012   LONDON SALES
TAPE/3           END
3 *Q1100 F1012   MANCHESTER SALES
```

These tapes would then form file 3. The Birmingham tape alone would be mounted and allotted programmer's number 3, i.e. the same as the file number, before the job begins.

The remaining tapes are mounted as required by use of the extracode 1007 (mount next reel of file Ba and allocate number n to it). The tape sections on the second and subsequent reels are not numbered consecutively from the preceding reel, but start again at section 1.

10.6.3 Deck Allotment

As stated above the first tape of every file and each single tape that is given a programmer's number in the job description, will be mounted on separate decks before the start of the job. These decks will then be available to the programmer throughout the course of the job. He can by extracodes cause any of the original tapes to be unloaded and new ones, given a tape reference in the job description, to be mounted in their place.

If however, at some stage in the program he requires more decks than will be allocated to him in this fashion, he must mention the number he will need in the job description. This is done by writing:-

DECKS d

where d is the maximum number of mechanisms that will be in use at any one time. Thus a programmer requiring 3 tapes to be mounted at some stage in the program but only 1 at the beginning, would give the single tape a programmer's number, give the other 2 tapes a tape reference and put in his Job Description:-

DECKS 3

10.7 Time Estimates for a Job

10.7.1 Computing Time

Since each program on Atlas will normally be time sharing with others there can be no direct control of an individual program by an operator. Thus, there is no means by which an operator may tell if a faulty program has entered an infinite loop and is thereby wasting machine time.

To overcome this problem a time limit for a program is placed in the job description and if the program exceeds this limit it is stopped by the computer.

The limit appears in the job description as:-

COMPUTING p.q SECONDS (or MINUTES, or HOURS)

where p.q is a decimal number. This time is found by allowing two microseconds to each instruction obeyed and adding to this the expected compiling time. The ABL compiler obeys 1500 to 1700 instructions per instruction compiled.

Alternatively the limit on computing time may be specified as

COMPUTING m INSTRUCTIONS

but one 'instruction' in this context means one instruction interrupt, equal to 2048 basic instructions obeyed. In fact the Supervisor actually times the program in terms of these units of 2048 instructions. Conversion from estimates given in terms of seconds, minutes or hours is made on the basis of 256 'interrupts' per second. Furthermore each multiplication instruction is counted as 2 instructions and each division instruction as 4.

For example a program requiring at most three million instructions, and having a compiling time of one and a half seconds would have the entry

COMPUTING 7.5 SECONDS

or

COMPUTING 1832 INSTRUCTIONS

If the computing time is less than 20 seconds this entry may be omitted completely from the job description. In this case a standard allowance of 20 seconds is made (5120 instruction interrupts).

10.7.2 Execution Time

If a program uses magnetic tapes it may be held up at some stage while a block of information is brought from tape to store, a time of 64 milliseconds. This wait can be eliminated in many cases by calling for a block 64 m.s. before it is needed, or by using a sufficiently large variable length transfer buffer or by resorting to branching. Some tape waiting time, however, may be inevitable and if it is likely to occur it must be shown in the job description.

This is done by the heading:-

EXECUTION p.q SECONDS (or MINUTES or HOURS)

where the time estimate is this case is found by adding an upper limit for the tape waiting time to the COMPUTING time.

For example, if the program quoted in 10.7.1 above was expected to be held up at most 200 times the job desoription would include:-

COMPUTING 7.5 SECONDS

EXECUTION 20.3 SECONDS

If the EXECUTION section is omitted for the job desoription the execution time is taken to be the same as the computing time.

10.8 Store Allocation

An estimate of the amount of store needed by the program is also required by Atlas to prevent a faulty program from monopolising the store by producing a large amount of useless information. This is done by the job description entry

STORE s BLOCKS

where s is the maximum number of 512 word blocks used by the program at any one time. No distinction is made between core and drum store. The word "BLOCKS" may be omitted if desired.

This section may be omitted, if the store requirement is less than 32 blocks, in which case 32 blocks will be allocated and charged for. If the estimate for store is exceeded at any time the program is stopped by the computer. One extra block should be allowed for each input and output stream. The blocks used by the compiler are not counted unless the computer is retained in the main store after the program is entered.

10.9 Job Description Format

10.9.1 Order of Sections

Separate sections of the job description may be listed in any order. For instance the OUTPUT section could precede the INPUT section, STORE could be followed by COMPUTING. However, to make it easier to read it is perhaps advisable to keep to a fixed order, for example the order in which the sections have been introduced. That is:-

INPUT
OUTPUT
MAGNETIC TAPES (this is not an actual section heading)
DECKS
COMPUTING
EXECUTION
STORE

10.9.2 Case Changes

Throughout this chapter the sections of the job description have been written in capital or upper case letters since these are common to all forms of input. Changes of case are ignored however, and if the job description is on seven-hole tape, lower case letters could be used well.

10.9.3 Backspace

Throughout the job description 'backspace' is an illegal character.

10.10 Composite Documents

10.10.1 Job Description Combined with Program

A job description may be combined with a program to form one composite document. In this case the last item of the job description will be followed, not by an end of tape marker (***Z) but by the program heading:-

COMPILER (Program Language)

and then by the program itself. No further title will be necessary since the composite document takes the title of the job description. This will be the usual form taken be small programs which are only run once.

If this procedure is adopted, no input zero is mentioned in the INPUT section of the job description and the computer will compile and execute the program immediately following the job description. If there are no separate data documents the INPUT section may be omitted completely.

The examples given below illustrate these facilities.

Example 1

```
JOB
F196, J. BLOGGS:  NUMBER FREQUENCY
INPUT
1 F196-TABULATED DATA
OUTPUT
0 LINE PRINTER
COMPILER ABL
------
------
------
------
***Z
```

Job Description/Program

```
DATA
F196-TABULATED DATA
------
------
------
------
------
------
------
***Z
```

Data Document

In this example the job required one other data document so the INPUT section must be included.

Example 2

```
JOB
F324/1, W. BROWN, FERRANTI:  PRIME NUMBERS
OUTPUT
0 LINE PRINTER
1 CARDS 2 BLOCKS
COMPILER ABL
----------
----------
----------
----------
***Z
```

Here no further information is required and the INPUT section has been omitted.

10.10.2 Job Description Combined with Data Document

It is also possible to combine the job description with a data document. This is particularly useful when the same program is to be run more than once, using different data each time. In this case the INPUT section of the job description must include:-

SELF = n

where n is the programmer's number for the data which follows on the same document. The program itself is specified as the lowest numbered input stream, in the same way as when the job description is a separate document, and the last item of the job description is followed by the heading

DATA

and then the data itself. No title is needed to identify the data.

For example, consider a wage calculation carried out each month using one fixed set of data, say a list of P.A.Y.E. codes and a second set of data consisting of a list of hours worked by each member of the staff. The second set of data would, of course, vary from month to month and could be combined with the job description while using the same program and P.A.Y.E. code tapes.

The program and data documents would be:-

```
COMPILER ABL
F900, WAGE CALCULATION

------
------
(Program)
------
***Z
```

```
DATA
F900, P.A.Y.E. CODES

------
------
(Data)
------
***Z
```

The job description could then be combined with the second data document thus:-

```
JOB
F900, J. SMITH LTD:  WAGES OCTOBER 1964
INPUT
0 F900, WAGE CALCULATION
1 F900, P.A.Y.E. CODES
SELF = 2
OUTPUT
0 LINE PRINTER 50000  LINES
1 CARDS 2000  LINES
DATA
------
------
(Data)
------
***Z
```


10.10.3 Data Files

It may be easier, for the purposes of some programs, to treat several distinct paper tapes or stacks of cards as a single data 'document'. Such a combination is called a data file. Each separate document is given a modified data heading of the form

DATA/m

where m is the number of the document within the file. All documents composing the file have the same title and each is ended by a ***Z end-of-tape marker, or a 7, 8 card punching. The last member of such a series must have the heading:-

DATA/m END

These documents may then be fed to the computer in any order and on any peripherals and the computer will combine them as required. In the INPUT section of the job description they will be referred to as one document with the title which each of them bears.

For example, if the data called U21, IRON CONTENT is on two distinct paper tapes these may be headed as follows:-

```
DATA/1
U21, IRON CONTENT

--------
(First part
 of data)
--------
***Z
```

```
DATA/2 END
U21, IRON CONTENT

--------
(Second part
 of data)
--------
***Z
```

If the programmer wishes to refer to the file as input two the INPUT section of the job description will contain:-

2 U21, IRON CONTENT

10.11 Tape and Card Markers

So far in this chapter only the marker ***Z and the 7, 8 punching have been considered as an end to a document. These are in fact the most common markers but there are others which are dealt with below.

On punched tape all the markers consist of *** followed by a single letter. On cards *** is not acceptable, and is replaced by punching the 7 and 8 positions in the first column of the card and the letter in the last column. The other 78 columns can contain anything at all. If the last column is blank, 7, 8C is assumed.

10.11.1 The Tape Markers ***Z, C, T and A.

a) ***Z indicates not only the end of a document but effectively also the physical end of the tape. The peripheral equipment concerned is disengaged by the computer and when re-engaged by the operator a new document will be read.

b) ***C indicates that the end of the current document has been reached but that another follows on the same paper tape. The end of the document is noted by the computer and reading is continued for the next document without interruption.

c) ***T indicates a temporary stop. When this marker is encountered the peripheral equipment is disengaged by the computer and when next engaged by the operator a continuation of the same document is read. Thus, if a document consists of two tapes the first part can be ended with a ***T. When this has been fed to the computer the second part is read by the same peripheral with no document heading and the computer will treat the two parts as one document.

   If the document is data it is better to use the data file system given in 10.10.3 since the parts may then be fed to the computer in any order, on any peripheral.

d) ***A is used only by a machine operator and is an instruction to the computer to abandon the previous incomplete document and disengage the equipment. It is required if part of a document is damaged before input is complete and the operator requires the computer to disregard the information it has already received.

10.11.2 The Binary Tape Markers ***B, E and F

Each of these markers indicates that a binary tape follows.

a) ***B. When this marker is encountered the computer reads the information following on the same document in binary, instead of internal code, to the physical end of the tape. There is no test for end of tape markers. The last 2 or 4 characters from a paper tape read in this fashion will be overwritten in the store by the 12 bit character 0707 (octal), which replaces any spurious characters generated as the end of the tape passes through the reader (this does not apply to punched cards).

b) ***F causes the mechanism to be disengaged. When it is next engaged by the operator the new paper tape is read to its physical end in binary. Thus ***F combines the effects of ***T and ***B.

c) ***E causes the input following on the same tape to be read in binary but a check is made for ***A, Z or C. When one of these is encountered, binary reading ceases and the appropriate action is taken. If the end of a reel of tape is encountered after ***E and before any of these *** sequences, the last 2 or 4 characters on the tape will be denoted by the 12 bit character 0707 (octal); the next tape will then be read as a continuation of the same binary document. When ***A, Z or C is encountered, it is itself stored in binary. On cards following 7,8E a card bearing 7,8A, Z or C is also stored in binary.

Note that if it is required to read to the physical end of a 7-track tape it is necessary to precede the ***B, E or F by ***P, as described below. If this is not done the tape may be rejected because of a spurious tape parity fault when the end of tape passes through the tape reader.

The marker ***B can also be used to read a fixed number of characters in binary. This is done by prefixing the marker by *n thus:-

*n***B

where n is the number of characters to be read. When the n tape or card characters have been read and stored, reading continues in internal code.

When reading punched cards an alternative way of causing the card to be read in binary is available. If the first column of a card is a standard code the contents of the card are converted to internal code. If the first column is not a standard code the card is assumed to be in binary and is stored as such. Naturally, after a 7,8 B, E or F has been read all cards are taken as binary regardless of their first column.

10.11.3 The Tape Marker ***P

When reading 7-track paper tape in internal code or in binary the Supervisor normally checks the parity of each character (an odd number of holes for correct parity) and rejects the document if a character with wrong parity (blank tape for example) is encountered. This parity checking may be suppressed by punching ***P, but it will be restored again at the end of the given document. If the input is binary and ***P has been punched, wrong parity characters will be recorded as punched, but if the input is internal code they will be replaced by the fault character 7.7 (inner set).

Note that to suppress parity checking on binary input it is necessary for ***P to appear before ***B, E or F; ***P after ***B, E or F will not be recognised.

10.11.4 Card Markers

The same markers are available for use with cards, but with *** replaced by a 7,8 punching in the first column of the card and with the letter in the last column.