ATLAS COMPUTER LABORATORY

USERS'

HANDBOOK

SCIENCE RESEARCH COUNCIL

APRIL 1969

AMENDMENT CHECK LIST

1	21	41	61	81
2	22	42	62	82
3	23	43	63	83
4	24	44	64	84
5	25	45	65	85
6	26	46	66	86
7	27	47	67	87
8	28	48	68	88
9	29	49	69	89
10	30	50	70	90
11	31	51	71	91
12	32	52	72	92
13	33	53	73	93
14	34	54	74	94
15	35	55	75	95
16	36	56	76	96
17	37	57	77	97
18	38	58	78	98
19	39	59	79	99
20	40	60	80	100

ATLAS COMPUTER LABORATORY
USERS' HANDBOOK

Contents

1. GENERAL INFORMATION

1.1 INTRODUCTION

The Atlas Computer Laboratory is one of the laboratories of the Science Research Council. Its main equipment is a large I.C.L. (formerly Ferranti) Atlas I Computer, supported by the services and ancillary machinery needed to operate a powerful computer efficiently. The machine was ordered in the autumn of 1961 and has been in regular operation since October 1964. The Laboratory is financed by the Treasury and its services are available to all universities and to Government-supported research laboratories and institutes.

The Atlas Laboratory makes no charge to universities or colleges for any of its services; the Government-supported users are charged at cost and the Laboratory is allowed to sell machine time, if there is any available after meeting the prime needs, to industry, charging at the current commercial rate. The Laboratory is not subject to any security restrictions.

1.2 SITUATION

The Laboratory is situated on the west side of the A.34 (Birmingham/Southampton) trunk road, 14 miles south of Oxford (there is a map at Appendix 1).

The nearest station is Didcot (5 miles); London Airport is 45 miles away; the City of Oxford No. 12 (Oxford/Newbury) 'bus passes the site.

The correct postal address is:

Atlas Computer Laboratory,
Science Research Council,
Chilton, Didcot, Berkshire.

Telephone Number: Abingdon 1900

(some extensions which may help users are given in Appendix 3)

Telex: 83303

Telegrams: ATLAB, Telex, Abingdon.

1.2.1 Laboratory Building

The Laboratory is in two parts: a computer Block (total area 15,000 sq.ft.) and an office block, which is itself in two wings, (total area 15,000 sq.ft.) linked by circulating area and stairways.

The computer is on two floors: the operating equipment (input/output and magnetic tape decks) on the upper floor and the bays of electronics (central

processor, stores and co-ordinators) below. The reception and data prepara-
tion area and the ancillary equipment room surround the computer operating
area on the upper floor. There is a "think room" just outside the entrance to
the reception area, where users can look over their work before handing it in
for processing or after getting results off the machine.

The office block houses the Laboratory's permanent staff and contains a
lecture room and the library and reading room. It also provides accommoda-
tion, mostly in small single offices, for users who would like to spend some
time in the Laboratory developing a program or supervising a big production
job.

Appendix 2 shows the approach to the Laboratory and gives a plan of the
building.

1.3   EQUIPMENT

At the time of writing (May 1968) the following is installed and working:

1.3.1   Atlas Computer (the Atlas word is 48 bits, organised in 8 6-bit characters)

Central Processor:	with 128 index registers (B-lines)
Main Store:	48K ferrite cores, 2 μs cycle time
	96K magnetic drums, transfer time 2ms for 512 words, average access time 7.8ms
Magnetic Disc:	Data Products Model 5045, capacity 16 million words, transfer time 6.5ms for 512 words, average access time 80ms
Magnetic Tape:	16 Ampex TM.2 decks (1 inch tape) 1 IBM 729 Mark IV deck (½ inch tape)
Console:	Engineers' console with input and output peripherals.
	Direct access typewriter for operators' messages and enquiries.

Input:	Card readers:	2 ICT — 600 card/minute
	Paper tape readers:	1 Ferranti — 300 character/ sec
		1 Elliott — 1000 character/ sec
Output:	Printers:	2 Anelex — 1000 lines/ minute
		120 character/ line
	Card punch:	2 ICT — 100 card/minute

Paper tape punch:	3 Teletype — 110 character/sec

N.B. The paper tape equipment will handle 5, 7 or 8 track tape.

Ancillaries

Graphical Output:	SC 4020 : see below
Card Sorter:	1 ICT — 750 card/minute
Reproducers:	2 ICT — 100 card/minute
Interpreters:	3 ICT — 22 character/second

Data Preparation Machines

Card Punch:	8 IBM Type 029
Card Verifiers:	4 IBM Type 059
Flexowriter:	6 (7-track Atlas code)
Teleprinter:	2 (5-track Mercury code)

1.3.2 Multi-Access Time-Sharing Consoles

Initially six Teletype consoles, distributed throughout the Laboratory, communicating with Atlas via the disc and a small satellite computer, an SDS Sigma 2 with core store of 32K words of 16 bits. (Later the number will be increased, probably to twelve.)

1.3.3 Graphical Output

Stromberg-Carlson 4020 microfilm recorder, driven by IBM ½-inch tape produced by Atlas. For high-speed production of printed material, plots, graphs, charts, microfilm and cinematograph film. Standard routines are available, in both Fortran and Algol, to simplify the use of this equipment. Full details of this equipment and of the Fortran routines are given in a special manual, SC 4020 Plotter Manual, Ref. 5.5.1.

The Algol routines are referred to in paragraph 3.3 and described in another Laboratory publication, Ref. 5.3.6.

1.3.4 Speed of Atlas

The instructions using the index registers take $1.6 - 2.2 \ \mu s$ and those using the accumulator take $1.75 - 3.2 \ \mu s$, except for multiplication which takes $5.9 \ \mu s$ and division which can take anything from 15 to 30 μs, averaging about 20 μs. The mean rate at which the machine obeys instructions, over an extended period, during which much input/output and magnetic tape activity takes place, is about 350,000 per second.

The following times were measured for some simple subroutines, written in machine language:

a. Compute e^x (using the extracode 1703)   154 microseconds

b. Sort 5000 floating point numbers into
    ascending order   1 second

c. Invert a 100 x 100 matrix (computation only)   17 seconds

1.3.5 Operating System

The term "job" will occur frequently throughout this manual. It is not easy to define but is readily understood. It refers to a run on the machine, with input, processing and output. A job can occupy the machine for seconds or hours, and can be completely self-contained or can use library or system programs. The machine does useful work only by processing complete jobs.

The machine is normally under the control of a large resident program called the Supervisor, which organises and monitors every activity. Input, output and magnetic tape transfers are time-shared with computation, and the Supervisor, by interleaving several programs, attempts to maximise the throughput of complete jobs. One action of the Supervisor is to produce a report on every job as soon as it has been completed, giving the starting and finishing times, amounts and kinds of input and output, numbers of machine instructions obeyed and other information. The accounting and the operational statistics for the machine are derived from this report.

The operating system requires that every program presented to the machine shall be accompanied by a Job Description, which itself must start with the word 'JOB' followed by a Job Number. The Job Description contains, amongst other information, statements of the language in which the program is written, of the amounts and types of input and output, and gives estimates of the amount of store needed and of the running time. Job Numbers are issued by the Laboratory in response to requests for time on the machine and users have to write their own Job Descriptions. Both are discussed in Section 2.

1.3.6 Programming Languages

The general principle is that programs are accepted in any language for which a compiler has been written. All the available compilers are held on the disc and are called in by means of simple statements in the Job Description (see Section 3 and paragraph 2.2).

Amongst others, compilers for the following languages are available:

ABL	(Atlas Basic Language — assembly code)
ASP	(Atlas Symbolic Programming — based on a mnemonic form of ABL)
FORTRAN	

ALGOL

Atlas Autocode (AA and AB) (Manchester University)

Extended Mercury Autocode   (I.C.T.)

SOL   (Simulation language)

LISP 1.5   List Processing

IPL-V   List Processing

Compiler Compiler   (Brooker-Morris, Manchester)

A full list is given in Section 3.

1.4   ORGANISATION OF THE LABORATORY

The total staff is about 100, organised as follows:

Director: Responsible to the Science Research Council for the Laboratory.

Operations Group: Responsible for all the activities involved in getting work through the machine. These include receipt and despatch, punching of programs and data, operation of magnetic tape library, main frame operations and offline operations such as card reproduction and use of the plotting equipment.

Programming Group: Responsible for all basic software-systems programs, compilers, development of library programs and packages; and for any special projects which the Laboratory has decided to undertake.

Support Group: Responsible for helping users to make the best use of the operational services and programming resources provided by the Laboratory. Users are required to write their own programs, but the Laboratory, through this group, will give help and advice and provide educational services such as programming courses.

Research Appointments: Individual research workers (not concerned with the day-to-day demands of the computing service.)

Administration Group: Responsible for the general running of the Laboratory and all the non-technical services, such as stores, accounts, accommodation and transport. The Laboratory draws on the services provided by the adjacent Rutherford High Energy Laboratory, for example for general engineering support and for Personnel and Finance work.

1.5 SERVICES PROVIDED

The Laboratory has been set up to provide a powerful computing service to a wide variety of users and it will do all it can to help its users to get their work done quickly and efficiently. To this end it is staffed in the way described in paragraph 1.4. The services offered are described in the following paragraphs. Users wishing to work in the Laboratory must send in a Visit Request Form, which is reproduced on the next page.

1.5.1 Computer Operation

Normally work will be put through the computer by the operator teams. There is usually no advantage in the programmer running his own work, and it is more efficient to leave this to the professional operators: but in special circumstances the opposite can be true, and whilst this will be discouraged there will not be a complete embargo on the programmer running or monitoring his work personally. Anyone wishing to take advantage of this should consult the Deputy Head of the Operations Group (Extension 515).

1.5.2 Evening and Night Priority Runs for Development

This is intended to help users who are developing large programs. A maximum of 8 people may have priority treatment outside normal working hours — that is, roughly, between 5.30 p.m. and 7.30 a.m. — and each may have up to 4 runs of not more than 3 minutes duration. Users give their cards or tape to the Shift Supervisor whenever they want a run and he will put them in the machine as soon as possible — which will often mean immediately. Priority must be booked in advance with the Receptionist (Extension 6296), who is not permitted to accept bookings for more than 4 users in any one shift; the Shift Supervisor is not permitted to give priority to anyone whose name is not on the list for that shift. Users will appreciate that unless priority treatment is very tightly controlled it becomes valueless.

1.5.3 Card and Tape Punching

The Laboratory will punch programs and data from manuscript: details of this service are given in paragraph 2.4. Normally this is done free, but the Laboratory reserves the right to raise a charge if the amount of punching is very great, or to require the user to get the punching done himself. This would be discussed with the user in advance.

1.5.4 Library and Reading Room

This is on the upper floor of the Office Block, above the main entrance. It is very much specialised to mathematics, computer science and computer applications, but contains also a comprehensive collection of general reference books. Users are welcome to read here, but are asked not to use it for working on programs, for which there is other accommodation in the Laboratory. The Librarian will be glad to help them.

VISIT REQUEST

Name:　1. _____

　　　　2. _____

University, College or Department _____

Address: _____

_____

_____

Telephone No: _____　　Extension: _____

Brief Reasons for Visit:

_____

_____

_____

_____

Job Number: _____

Is any special Programming or Punching assistance desired (with brief details)?

_____

_____

_____

Is Office accommodation required in the Laboratory?　　　　　YES / NO

Proposed dates of visit:　　Arrival - _____　Departure - _____

　　　　　　　　　　　　　　　　　　　　　　　　　　　　Time of Arrival

Will Transport be required from:　　　Oxford Station　　　_____

　　　　　　　　　　　　　　　　　　Didcot Station　　　_____

　　　　　　　　　　　　　　　　　　London Airport　　　_____

　　　　　　　　　　　　　　　　　　　　　　　　　　(Flight No._____)

　　　　If Laboratory transport NOT required,
　　　　approximate time of arrival　　　　　　_____

Are Hotel bookings to be made on visitors behalf?　　　　　YES / NO

　　　　AREA: (If any preference)　　_____

(Note: It may not be possible to arrange hotel accommodation unless the Laboratory has a clear week's notice)

　　　　　　　　　　　　　　　　　　Signature: _____

　　　　　　　　　　　　　　　　　　Date: _____

For Internal Use

Receptionist　　　　　　　　　　　　　　　　　　Operations Group
Hotel
Transport
Room Number
Visitors from

OSL

AL.16(4/68)

1.5.5 Accommodation

The "think room", immediately outside the machine room, is reserved for visiting users and is available to them without formality. Its furniture includes an Anita electronic desk calculator, to help in checking programs.

A suite of small offices is provided for users who wish to spend a significant amount of time in the Laboratory; they are booked in advance with the Receptionist.

1.5.6 Hotel Bookings

The Receptionist will make bookings for users. A telephone request must be confirmed in writing as quickly as possible.

1.5.7 Transport

The Laboratory will transport users to or from local hotels, Didcot Station or London Airport. Bookings are made with the Receptionist.

1.5.8 Telephone Calls

The users' offices all have telephones, from which local calls may be made freely; but long-distance calls will be charged. There is a telephone (Extension 312) for general use in the entrance hall. The telephones in the data preparation and machine rooms, are strictly reserved for the staff, as they carry very heavy traffic. Anyone with any special communications problems should seek the help or advice of the Receptionist.

1.5.9 Meals

The Rutherford Laboratory Restaurant is available and serves breakfast, lunch and evening meal every day, including Sunday. Users are welcome to have tea and coffee from the Atlas Laboratory's regular service and will be asked to pay for this. They may use the kitchen during the evening and night shifts — not during the daytime, when only the kitchen staff may use it — but must clear up thoroughly after any food preparation. Supplies of coffee, milk, etc. are left in the kitchen and anything used must be paid for. The Receptionist will explain all these arrangements.

2. PUTTING A JOB
ON THE MACHINE

This section is particularly important and should be read with care and attention before any application is made for work to be done in the Laboratory.

2.1 THE JOB NUMBER

2.1.1 Introduction

This is the vital reference number for any piece of work put on the computer, used in all costing and accounting operations. It must form part of the title in the Job Description. It is intended for identification of a complete program or project rather than of an individual run. Job Numbers must have a specified form and are issued only by the Laboratory.

Job Numbers are made up of 5, 6, 7 or 8 alpha-numeric characters of which the first is usually one of the letters V, T, E or I and differentiates between the main groups of users.

> Thus V denotes a university user
> and T one from a College of Technology; the two are dealt with in basically the same way and will not be distinguished in what follows
>
> E (External) denotes a government, Treasury-supported or commercial user
>
> I denotes a member of the Atlas Laboratory staff

Occasional use is made of other identifiers for special purposes. The first five characters of the Job Numbers are obligatory, but the user has the option of adding a further one, two or three characters if he chooses. These can be used to label any sub-division of the work which the user finds convenient; but they are disregarded by the costing and accounting programs and therefore the Laboratory cannot distribute, say, the total cost of a project amongst the sub-divisions represented by these final digits.

The method of getting a job number varies for the different user groups.

2.1.2 University and Technical College Users

> Job numbers have the form:
>
> $$V \, \alpha xxxyyy$$
>
> or $\quad V \, \alpha\alpha xxyyy$
>
> (or the same format preceded by T).

where the α's are letters identifying the university

the x's are digits 0-9 identifying the job and assigned by the Laboratory

and the y's are characters (letters, digits, or other valid symbols) assigned by the user for his own purpose.

Thus VC987 would identify a job from a user at Cambridge. It would be valid as it stands, but the user could identify sub-divisions of his work by employing, for example, one of the following variations:

VC9871   VC98704   VC987025   VC987AI   VC987A/B   VC987 AB
VBR would identify a job from the University of Bradford and could be correspondingly extended. Note that the separators and spaces are each counted as one character. The accounting program would treat all these sub-divisions as a single job, in this case VC987;   but individual runs will be labelled with the full identifier.

In every university there is a representative for the Atlas Laboratory who is either the Director of the University's own computer centre or someone nominated by him.   He holds stocks of all literature issued by the Laboratory, including the form on which University users must apply for time on the machine. The form for university users is reproduced on the next page. That for Technical College users is almost the same. The information asked for on this form is *essential* to the running of the service. The form must be countersigned by the university representative, who is asked by the Atlas Laboratory to satisfy himself that

(a)   the work is worth doing;

(b)   the amount of time asked for is reasonable;

(c)   the work cannot reasonably be done on the University's own computer.

Completed forms should be sent to:

External Reception,
Atlas Computer Laboratory,
Science Research Council,
Chilton,
Didcot,
Berkshire.

University users must apply to their local representative for any literature they need. The Laboratory will not deal with individual requests.

Forms not countersigned or otherwise incomplete are returned to the applicant.

On receiving a properly completed application form the Laboratory will issue

Science Research Council

ATLAS COMPUTER LABORATORY

Chilton, Didcot, Berks.

Telephone Abingdon 1900
        Extension 6284

UNIVERSITY   USERS

JOB APPLICATION FORM

NAME:-

UNIVERSITY OR COLLEGE:-

        DEPT. OR SCHOOL:-

        FULL POSTAL ADDRESS:-

TELEPHONE NO.:-
EXTENSION:-

SHORT DESCRIPTION OF JOB:-

		Rate per month (mins.)	Period (mths.)	Total (hrs)
ESTIMATE OF COMPUTING TIME (Please state approx. rate in mins. per month for the period the job is expected to take. State total in hours.)	Starting Date			

CLASSIFICATION (Please mark appropriate box):-

MATHEMATICS	1.	MEDICINE		8.	
PHYSICS	2.	METEOROLOGY		9.	
CHEMISTRY	3.	ECONOMICS		10.	
BIOLOGY	4.	SOCIOLOGY		11.	
ENGINEERING	5.	PSYCHOLOGY		12.	
CHEMICAL ENGINEERING	6.	ADMINISTRATION		13.	
METALLURGY	7.	ALL OTHER WORK		14.	

IS THE JOB SUPPORTED BY FUNDS FROM A NON-UNIVERSITY BODY?   YES ☐   NO ☐

If 'YES' please give details below or on a separate sheet:

SIGNATURE OF APPLICANT: _____

* APPROVAL SIGNATURE: _____

* All applications must be signed by the appointed University representative, usually the Director of the University
  Computing Laboratory. For list of those with power of approval see overleaf.

AL8. REVISED.

a Job Number. This is, in effect, the applicant's authority to use the Laboratory's services and facilities, and he may then deal directly with the Laboratory. It is for each university representative to decide how closely he wishes to watch the progress of any work he authorises.

When there is real urgency an applicant can get a Job Number by telephoning Abingdon 1900, Extension 6284, provided that the local representative has been consulted and agrees to this. A completed form should be sent as quickly as possible; all work on the job will be stopped if this has not been received after two reminders have been sent.

The university representative should be told if any work which he has authorised seems likely to need more time than was asked for. The Laboratory reserves the right to ask for a further authorisation, and also to hold up work on a project which has taken more time than was authorised. Users can however rely on helpful treatment, consistent with the Laboratory's obligation to give a good service to all its users.

The user is invited to get in touch with the Support Group if he expects his job to require a large amount of machine time, or if he wishes to use any unusual facilities.

Users must apply for a new job number when:

(a)   the nature of the project changes significantly;

(b)   they move from one university to another.

They should also inform the Laboratory when their project is complete, as the job number may be cancelled.

A job number must be used only by the person to whom it was issued.

2.1.3   External Users

Job Numbers here have the form

$$Exxxxyyy$$

where the x's are digits 0-9 identifying the job and are issued by the Laboratory, and the y's are characters assigned optionally by the user. As before, the accounting program disregards the y characters and assigns everything to the number Exxxx, but the individual runs are labelled with the full identifiers.

External users, who have been almost wholly Government Departments or Institutes supported with Government funds, are charged for the use they make of the Laboratory. It is therefore necessary for the Laboratory to know that the applicant has authority to incur the charge. Intending applicants are very welcome to discuss their problem with members of the Laboratory, but the first step in obtaining approval and hence a Job Number must be a formal letter to the Director.

Establishments of the Science Research Council are treated as External users but are not charged.

2.1.4   Internal Users

Job Numbers here have the form

$$Ixxxxyyy$$

with the x's and y's representing digits as before.

All internal work is authorised within the Laboratory.

2.2   JOB DESCRIPTION

2.2.1   Introduction

The operation of Atlas is controlled by a built-in program called the Supervisor. This, amongst other things, arranges that input and output operations and magnetic tape transfers go on simultaneously with computation. In order to run the whole installation with maximum efficiency it is necessary to hold several programs in the machine available for execution and for the Supervisor to have information about the requirements of each one. This information is given in the *Job Description,* which is normally punched on the first few cards or the first part of the input tape, whichever is the medium used. The

information required is:

> The Job Number and title (for identification)
>
> The expected duration of the run
>
> The amount of store needed
>
> Definition of input streams
>
> The amount and form of the output streams
>
> The numbers and titles of any magnetic tapes to be used
>
> The numbers and titles of any disc areas to be used
>
> The name of the compiler.

This information is picked off during input and stored in directories, which are used by the Supervisor in organising the operation of the complete installation. The Job Description is clearly an exceedingly important part of every program, and must be constructed with care. *Prospective users are strongly urged to read the I.C.T. publication "Preparing a Complete Program for Atlas I" which gives more information on the Job Description.* Ref.5.1.2.

It is convenient here to define a *Document*. A program with its data is thought of as made up of one or more documents, each of which is a self-contained section of information and is read into the machine through one input channel. The actual method of splitting up a program into documents is arbitrary and the different documents of a single program need not be presented on the same input medium — cards and the various forms of paper tape can be mixed in any way. But it is usually convenient to keep the number of documents down and to have them all on the same medium. Every document has to be identified by a heading such as JOB or DATA, and a title, and must be terminated by an end of document marker, which may take several forms, as described in 4.6. The Job Description must show how many documents make up the program; the Supervisor will not start to process the program until all the documents have been supplied.

2.2.2   Make-up of the Job Description

It should include these sections:

> JOB TITLE   STORE
>
> COMPUTING   COMPILER
>
> OUTPUT

and *may* include these:

> EXECUTION   TAPE
>
> INPUT   DISC

The various sections are described below.

2.2.2.1 Job Title

This must begin with the word JOB, followed on the same or a new line by the Job Number (paragraph 2.1), the programmer's name and a descriptive word or phrase. The complete title (that is, the word JOB, the number, the descriptive word or phrase and all spaces) must be not more than 80 characters long and it must not contain any back-spacing characters. The title will be reproduced as a heading on the output, and must therefore be the same as that written on the Operations Request Card so that the machine-room staff can collate output and input documents correctly. The same title should not be used for different programs.

2.2.2.2 Computing

This is an estimate of the amount of central processor time required by the run. It can be given in SECONDS (or abbreviated to SECS), MINUTES (or MINS) or HOURS,

> e.g. COMPUTING 40 SECS
>
> COMPUTING 1.6 MINS
>
> COMPUTING 1 HOUR or 1.3 HOURS
>
> (but not 1 HOUR 18 MINS)

Alternatively the computing time can be given in units of *instruction interrupts,* which are groups of 2048 basic machine instructions, roughly equivalent to 7 milliseconds computing time. This is abbreviated to INSTRUCTIONS, so one can write

> e.g. COMPUTING 15000 INSTRUCTIONS

Only integral numbers are allowed here and no commas or spaces must be used. An adequate rule is that approximately 10,000 instructions give 1 minute of computing.

2.2.2.3 Output

A job may have up to 7 output streams, which are to be identified by numbers from 0 to 15 inclusive. Diagnostic information produced by the Supervisor is always allocated to Stream 0. If Fortran cards are to be produced these must be assigned to Stream 15, and if there is to be graphical output this will use Streams 12, 13 (and 14 for IBM tape). With these restrictions, programmers may number their output streams as they wish.

The forms of output are specified as

> LINEPRINTER
>
> FIVE HOLE PUNCH
>
> SEVEN HOLE PUNCH
>
> CARDS

The word ANY can be used, but at Chilton is automatically interpreted as LINEPRINTER.

It is possible to produce 8-hole paper tape but it will be done only by special arrangement.

Stream 0 must always be defined as LINEPRINTER. If a binary deck from a HARTRAN program is required, this must be declared on Stream 15 as CARDS. For example:

                    OUTPUT

                    0 LINEPRINTER 500 LINES

                    2 ANY 200 LINES

                    6 SEVEN HOLE PUNCH 4 BLOCKS

                    15 CARDS 200 LINES

LINE means one line on the printer, which can have up to 120 characters. In the CARDS stream, LINE means one symbolic or about half a binary card.

BLOCK means about 4,000 characters, equivalent to about 24 binary or 46 non-binary cards. The Atlas block is 4,096 characters but the Supervisor uses a number of these for various tags and labels.

2.2.2.4 Store

Estimates of the amounts of main store required should be given, for both compilation and execution. The form is

                    STORE M/N BLOCKS

A block is 512 Atlas words, M is the number required for execution, N for compilation. The maximum amount available is 215 blocks. Programmers should always try to keep their demands below 120 blocks, as this allows more efficient multi-programming and therefore speeds up the machine's throughput. The End-of-Job Statistics (see paragraph 4.1) printed out with the results of every run will enable the programmer to make progressively more accurate estimates of the store demand.

If the compilation store N is not given, the Supervisor will allocate a standard amount, according to the compiler used by the program. A table of these amounts is given in paragraph 2.2.2.5.

If in the store request N is less than M, the Supervisor will allocate the larger amount M for both execution and compilation.

2.2.2.5 Compiler

The statement

                    COMPILER X

or   USE X

gives the Supervisor the information that the compiler for language X is required. If the input medium is paper tape this must be terminated with a new-line character. The following table of the most commonly used compilers gives the standard allocation of compilation store, referred to in paragraph 2.2.2.4.

AA	80 blocks
ALGOL	100 blocks
CHLF3	80 blocks
EMA.	80 blocks
HARTRAN	120 blocks

This completes the list of obligatory sections of the Job Description. As an example, the following would be accepted:

2.2.2.6 Example

```
JOB VPQ234 SMITH INTEGRALS
COMPUTING 30 SECS
OUTPUT  0 LINEPRINTER 500 LINES
              1 ANY          100 LINES
            15 CARDS         100 LINES
STORE 30/125 BLOCKS
COMPILER HARTRAN
```

2.2.2.7 Input

As with OUTPUT, input streams can be identified by numbers from 0 to 15; a single program can declare up to 16 streams. The program itself is assigned to Stream 0, which need not be declared in the Job Description. Very often the data will be part of the same document, in which case there is no need for any INPUT statement at all. But if data are to be read in as separate documents, each separate document must be identified by a title and declared in the Job Description by INPUT followed on the same or a new line by a stream number and a title:

```
e.g. INPUT
        3 SAMPLE VALUES A
        4 SAMPLE VALUES B
```
or `                INPUT 2 OBSERVATIONS`

The word INPUT need not be given more than once.

The corresponding data documents must then be headed:

> e.g. DATA
>
> SAMPLE VALUES A

or DATA SAMPLE VALUES A

etc.

Inside the machine input documents are identified by their title and not by stream numbers; therefore the title on the document must agree exactly, character for character, with that given in the Job Description. No two documents should have the same title, and a title must not be more than 80 characters long.

2.2.3 Magnetic Tape and Disc

As explained in paragraph 2.5, there are two magnetic tape systems, using Ampex and IBM tapes respectively, and a magnetic disc file. The Ampex system and the disc form the main backing store and are in many ways equivalent, whilst the IBM system is for various specialised uses.

2.2.3.1 Main System

There are 16 Ampex TM.2 decks, which use 1-inch tape laid out in pre-addressed blocks of 512 words each; the first block — Block 0 — contains the serial number of the tape and the title, and the information starts in Block 1. There are about 5,000 blocks on each tape, totalling about 2½ million words. Each block is 5.46 inches long with an inter-block gap of 2.3 inches, giving a total length of 3,600 feet. The normal tape speed is about 10 feet per second with fast wind and re-wind speeds of 15 feet per second for long searches. The instantaneous transfer rate is 90 Kcs, equivalent to about 89 μs per word. Allowing for the inter-block gaps the average transfer rate is about 64 Kcs.

The *Disc File* is the Data Products Model 5045, holding 16.78 million Atlas words on 32 physical discs. There are two independent sets of read/write arms, each with its own control. The information is distributed over 256 tracks on each surface of each disc and the 50 bits (48 + 2 parities) of each word are read from 4 different tracks. During a single revolution, taking 52 milliseconds, 4096 words are accessed, so the mean transfer rate is 12.7 microseconds per word. The mean access time is 195 milliseconds, including positioning, verification and latency, with a maximum of 300 milliseconds. The mean access time to an adjacent track is 135 milliseconds.

An Ampex tape or a disc area is referred to in a program by a logical number, which must be between 1 and 126 inclusive, though it is advisable to use up to 99 only. If the tape or disc area is required for future use a previous application must have been made for a title, as described in paragraphs 2.5 and 2.6 (unless one of the standard systems on tape or disc is being used). There must be a declaration in the Job Description consisting of the word

TAPE or DISC followed on the same or a new line by the logical number and
the title. As explained in paragraph 2.5.1, the title should not contain more
than 14 characters.
Thus for example:

> TAPE 4/100 N0539 UNSORTED DATA* INHIBIT
>
> TAPE 9 N0772 SORTED DATA* PERMIT
>
> DISC 4 R012 OLD DATA
>
> DISC 9 W345 NEW DATA
>
> DISC 99 R500 FORTRANLIB

would be valid titles. In the case of the tapes, the operators would be instruc-
ted to mount them so that N0539 could not be written on, and N0772 would be
write-permitted. In the case of the disc, a built-in check would ensure that
R012 OLD DATA could not be written to, while W345 NEW DATA could.
The system disc area R500 FORTRANLIB is always accessible to the user,
but he is unable to write to it. Tape 4 would be loaded so that it is positioned
at block number 100, which will reduce waiting time if the user knows that
his information for a particular run starts there.

If temporary storage is required only during the execution of the current pro-
gram, it is preferable to declare in the Job Description

> DISC COMMON 6/25
>
> DISC COMMON 5/50

This reserves common areas of the disc for the duration of the program,
assigning logical numbers 5 and 6, of lengths 50 and 25 blocks respectively.
No one program may use more than 2000 blocks in total. Alternatively, it is
possible to use

> TAPE COMMON 5
>
> TAPE COMMON 6

but this is less desirable, as it increases the amount of tape handling, and
also may be wasteful in machine time, as access to the disc is much more
efficient in the avoidance of 'tape waiting' time.

Note that in all these cases, the logical number *follows* COMMON.

A program may use up to 12 tapes in all, including COMMON and IBM, but the
number should be kept as low as possible. Programs requiring large numbers
of tapes present considerable operational difficulties and are likely to suffer
delays.

2.2.3.2 IBM System

This is used when it is necessary to transfer information between Atlas and any equipment which itself uses IBM tape, including IBM computers and the SC 4020. The tape is ½-inch wide and a reel can be made to hold up to about 1.8 million Atlas words. It does not have a fixed block structure, nor is there a title on the tape itself. Great care must be taken to check that the same title is written in the Job Description, the Operation Request and on the label on the tape reel, because the computer cannot check that the correct tape has been used.

Only one tape may be used by a program. This must be declared by the words TAPE IBM, followed by a logical number in the range 0 to 126 and a title.

e.g.   TAPE IBM

100 SMITH NEW DATA

OPERATION REQUEST

Job Number:

Name:

Telephone:
Title:

	Time			Store
Computing		Compile		
Execution		Execution		

Compiler:

INPUT/OUTPUT	Input	Output Qty.	BRK.O/P
Line Printer			
Cards			
Paper Tape			
8 Track			
7 Track			
5 Track			
Graphical			

COMMON	Magnetic Tape		Disc	

Magnetic Tape Titles

I.B.M. Tape Titles

Private Disc Areas

Destination:	Program Restarts
Return	Manual
Retain at A.C.L.	Automatic
Support Group	Comments: **P.T.O.**

AL.7(9/68)

2.3   OPERATION REQUEST

To have a program run on the machine it must be sent to the Laboratory with a completed Operation Request Card. This is reproduced on the facing page, and explanatory notes are given below.

Job Number: See Section 2.1.

Title: This is the title given in the Job Description (see Section 2.2.2.1).

Computing Time: This can be given in hours, minutes or seconds, e.g. 7.5 seconds.

Execution Time: Specify time if magnetic tapes/disc areas are used.

Compile Store: Specify number of blocks required (see Section 2.2.2.4).

Execution Store: Specify number of blocks required (see Section 2.2.2.4). This important quantity must be given as accurately as possible.

Compiler: Indicate which compiler is to be used.

Input: Tick the relevant boxes to show the form of input used.

Output: Indicate in the appropriate boxes the forms and quantities of output expected. If "BREAK OUTPUT" is included, please tick the boxes for the output media that are affected (see Section 4.3).

Magnetic Tape/Disc Common: Indicate the number of common tapes and common disc areas used. The use of common tapes is now only required if the total common area exceeds 2,000 blocks (see Section 2.2.3.1).

Magnetic Tapes: Give the tape numbers of tapes to be used. See Section 2.5 *Magnetic Tapes* for information about tape numbers and how to obtain them.

IBM Tape Titles: Give IBM tape titles for both input and output. Note that tapes to be "file protected" should be labelled accordingly (see Section 2.5).

Private Disc Areas: Give the key numbers of all disc areas to be used, e.g. W012, R234 (see Section 2.6).

Destination: Tick the appropriate box for the return of the job.

Program Restarts: Tick appropriate box if MANUAL or AUTOMATIC RESTARTS are incorporated.

Comments (to be written on reverse side of card): Any other remarks which may be useful to the Operating Staff.

The information given on this card should agree with that given in the Job Description. The Computer Operators use the information from this card to schedule work for the machine and incorrect information can lead to waste of machine time.

2.4   DATA PREPARATION SERVICES

2.4.1   The Laboratory offers a range of facilities for punching cards or paper tape (5 and 7 channel) from manuscript. Users may apply for punching to be done by the Laboratory's Data Preparation staff, by submitting a Card and Paper Tape Preparation Request Form (see specimen below).

<div style="border:1px solid">

Science Research Council
ATLAS COMPUTER LABORATORY
Chilton, Didcot, Berks.

Tel: Abingdon 1900
Extn. 6284   **CARD AND PAPER TAPE PREPARATION REQUEST**

DATE   _____

NAME   _____

TEL. NO. _____

ADDRESS _____

_____

_____

_____

TYPE OF WORK	
PAPER TAPE 5 CHANNEL	
PAPER TAPE 7 CHANNEL	
PUNCH AND LIST	
EDIT	
COPY TAPE	
PRINT AND COPY TAPE	
CARDS	
PUNCH	
VERIFY	
LIST	
REPRODUCE	
INTERPRET	
SORT	

TYPE OF CODING	
FORTRAN	
ASP.	
ABL.	
ALGOL.	
MERCURY AUTOCODE	
EXT. MERCURY AUTOCODE	
ATLAS AUTOCODE	
ANY OTHER CODE (PLEASE SPECIFY LAYOUT REQUIRED)	

Please tick the appropriate boxes for both type of work and type of coding.

OSL   AL 13(11/65)

</div>

To: Data Preparation Section,
Atlas Computer Laboratory,
Science Research Council,
Chilton, Didcot, Berkshire.

Telephone enquiries should be made to: Abingdon 1900, Extension 6284.

The demand for this service is considerable but irregular and no specified turn-round can be guaranteed. In general, punching of programs will take precedence over data. Users preparing large projects can help by sending the work in instalments. The Laboratory has arrangements with several punching bureaux and may off-load work on to these at peak periods (see also paragraph 1.5.3).

The Laboratory will not punch a program from a computer print-out unless the applicant can give an assurance that he has tried and failed to get a machine-produced copy.

Users with any unusual requirements, such as the punching of cards from survey questionnaires, are strongly recommended to contact the Operations Group before submitting work, preferably even before designing the question-naires, as this will save a great deal of trouble at later dates, and often drastically reduce the amount of punching effort required.

2.4.2 Rules and Conventions in Coding

A. Submit a clear and legible coding sheet, or a typed or printed sheet. Illegible or ambiguous work will be returned.

B. Distinguish carefully between certain pairs of characters which are easily confused, as follows:

   1 (one) from I (letter)   :   convention is I (letter)

   1 (one) from / (oblique, slash)

   2 (two) from Z (letter)   :   convention is Z (letter)

   6 (six) from G (letter)

   0 (zero) from O (letter)   :   convention is $\overline{\mathsf{O}}$ (letter)

C. Indicate spaces (blanks) by a number within a circle:

   e.g. A ④ .. B ② .. C   for AbbbbBbbC   (b = blank)

   The number, e.g. 4 takes up one of the 4 spaces required and must be followed by the remaining 3 spaces. This simplifies the punch operator's task.

D. For Mercury Autocode use

                   Ø rather than x

                   ⁻ rather than μ

                   ' rather than ⁓

E. Provide a note of any departures from conventional coding prac-tice.

F. Specify lay-out of data by means of an example, not a format statement.

G. Give precise instructions for any insertion of blank cards or leaving of lengths of blank tape.

2.4.3 Users' Punching Equipment

A number of card punches and punched paper tape equipment in the Data Preparation Room is set aside for users of the Laboratory. They are intended to enable users to do small amounts of punching, for example to make corrections to a program, and should not be monopolised by one user for a long period. It is an accepted rule that a user does not occupy a card punch for more than 5 minutes or a Flexowriter for more than 15 minutes if another user is waiting. A machine belonging to the punch-room service may, if vacant, be used, but only with the permission of the Head of the Data Preparation Section or her deputy.

A new visitor to the Laboratory must make himself known to the Head of the Data Preparation Section, or her deputy, before using any of this equipment; and all users are asked to report any faults they find in a machine, so that the maintenance engineer can be told.

Finally, a rule of great importance: leave the machine clear of cards, tape and paper and *throw away, at once, all used cards.*

2.4.4 Other Equipment

The following machines are located in the Data Preparation area and are available for general card handling:

2 card reproducers	(I.C.T. model 203)
3 card interpreters	(I.C.T. model 424)
1 card sorter	(I.C.T. model 313)

The staff of the Data Preparation Section will do work of this kind for users, but anyone with a large amount to be done is expected to do the work himself. A user wishing to do such work, or to have it done for him, must first ask the Head of the Data Preparation Section, or her deputy.

2.4.5 The Handling of Cards and Paper Tape

Care is needed at all times. Anything sent through the post should be firmly packed and protected from being crushed. There are a number of commercial products available to ensure the safe transit of cards and paper tape: the Laboratory will advise if requested.

The following is a general guide to good practice:

(i) **Paper Tape**

a. Keep all paper tape free from contact with dirty surfaces.

b. Ensure that tapes are punched on equipment which has been correctly set. The requirements are given in the I.C.T. document "Specification of Dimensions for Punched Tapes" — CS 364.

c. Use the paper tape specified for I.C.T. computers and made by Waterlow.
See I.C.T. document "Specification of Paper for Punched Tapes" — CS 363.

d. Do not use oiled tape.

e. If splices are necessary, they must be carefully made. The Lab-
oratory will if requested advise on equipment.

f. Leave a 6" run-out at the beginning of the tape. On this run-out
write a title which will enable the operators to identify the owner
of the tape and the job to which it belongs.

(ii) **Punched Cards**

a. Store cards away from heat and damp.

b. Ensure that the edges of a deck are not damaged.

c. Always work on a clean surface.

d. If possible use I.C.T. cards. Cards made by IBM or other manu-
facturers will be accepted, but since the Atlas readers are adjust-
ed to I.C.T. dimensions, this may lead to difficulties and delay
the running of the work. Heavy weight quality.

2.5   MAGNETIC TAPES

2.5.1   The main system is based on *Ampex 1-inch tape,* on which the infor-
mation is written in pre-addressed blocks of 512 words each, about 5,000
blocks to a tape. Every reel is labelled with a 4-digit number preceded by
the letter N (that is, of the form Nxxxx) which is given by the Laboratory;
and must carry a title, which is given by the user.

A user wishing to have tapes assigned to him must complete a Magnetic
Tape Request Card (see specimen below).   The University representatives
hold stocks of these cards.

AL. 10(3/66)

Science Research Council
ATLAS COMPUTER LABORATORY
Tel: Abingdon 1900 Ext. 6640

MAGNETIC TAPE REQUEST CARD

NAME:
ADDRESS: TELEPHONE NO.

JOB NO: DATE:

Please supply _____ magnetic tapes titled:

TITLE

It should be sent to:

> The Magnetic Tape Librarian,
> Atlas Computer Laboratory,
> Science Research Council,
> Chilton, Didcot, Berkshire.

The Librarian will allocate and title the tapes requested and advise the applicant on an Allocation Card (see specimen below).

AL.11(1/68)

SCIENCE RESEARCH COUNCIL
ATLAS COMPUTER LABORATORY
Tel: Abingdon 1900 Ext. 6284

MAGNETIC TAPE ALLOCATION CARD

The following magnetic tapes have been allocated to you as requested:

TAPE NUMBER   TITLE

As with Job Numbers, requests for tapes may be made by telephone: the number is Abingdon 1900, Extension 6640. A request card must be sent in support of a telephone request as soon as possible.

All requests should be made as far in advance as possible, to give the Laboratory staff time to label and address the reels.

When a tape is loaded the Supervisor compares the title, which is written in Block 0, with what is called for in the Job Description. The slightest discrepancy will cause the tape to be rejected and the run to be terminated. A title, excluding the tape number, should not exceed fourteen characters.

Comments may be added to a title by terminating it with an asterisk — thus

N0987 JONES DATA* NEW STYLE

The * must follow *immediately* after the last character of the title, or chaos will ensue. The characters after the * will be reproduced on the operators' teleprinter, but not checked against the title on the tape. The complete string must not exceed 30 characters; space, or multiple spaces, counting as one character. The main use of this facility is to tell the operators how the tape is to be mounted on the deck — with or without a write-permit ring. If information is to be written on to the tape during the running of the program, one puts:

<div align="center">N0987 JONES DATA* WRITE PERMIT</div>

To tell the operators to mount a tape so that nothing can be written on it, one puts:

<div align="center">N0987 JONES DATA* FILE PROTECT</div>

The instructions can be shortened to

<div align="center">*WP</div>

or

<div align="center">*FP</div>

which are recognised by the operators.

It is good practice to keep tape titles as short as possible.

Users are asked to inform the Tape Librarian *in writing* when they have finished with any tapes allocated to them, so that these can be released for re-allocation. Periodically the Librarian will send to every user a list of all the tapes he is holding, with a request to release any which are no longer needed.

2.5.2 IBM Tapes

In addition to the Ampex system, the Chilton Atlas has an IBM 729 Mark IV deck which uses IBM ½-inch, 7 channel tape written at 556 or 200 bits per inch.

This is the medium used to drive the SC 4020 microfilm recorder on which all graphical output is produced; for use for this purpose, see the Laboratory's publication 'SC 4020 Plotter Manual' Ref. 5.5.l. It can be used also to transfer data or other information between Atlas and other computers, for example IBM machines. The identifying numbers are assigned by the Laboratory as with the Ampex tapes, but all begin with 9. However, the use of IBM tape on Atlas can present problems and prospective users are asked to seek the advice of the Support Group before applying for these tapes, or submitting tapes from other installations to be read on Atlas.

Tape reels may not normally be taken out of the Laboratory. Anyone wishing to record information from Atlas and physically transfer it to a machine elsewhere will be asked to provide the reels required.

2.6   THE MAGNETIC DISC

The Chilton Atlas is provided with a large disc file. This is used for several purposes, and in particular it may be used in exactly the same manner as I"

Ampex tapes. A user who wishes to obtain his own titled area on the disc must apply to the Head of the Operations Group, who will require to be convinced that the user has a genuine need for this area, and that the overall performance of the machine will be improved by the allocation, as space is in great demand.

The operation of the disc is similar to that of the tape system. A title will be allocated to the user of the form

<div align="center">012 SMITH I</div>

The numerical part (012 in the example) called the key number *must* be preceded by either W or R. The user should specify in his disc application whether he wishes to write to the disc: if he does not wish to do so, a special marker will be set in the disc directory; if this is the case, it will be impossible to write to the disc area, and the key number must be preceded by R, e.g. R012; this may be useful when using a developed program. If, however, writing is allowed, the user has the option of preceding his key number by either R or W, e.g. R012 or W012. In the former case, any attempt to write to his area will be stopped, whereas in the latter case both reading and writing are permitted and he may do as he wishes.

2.7   SUMMARY OF ACTIONS REQUIRED

   (i)   Obtain a *Job Number* (paragraph 2.1) by completing a *Job Application Form* (paragraph 2.1.2).

  (ii)   Request any *Magnetic Tapes* (paragraph 2.2) or *Disc Areas* (paragraph 2.2.3, 2.6) required and make sure that you receive the tape numbers together with the given titles correctly written.

 (iii)   If you wish to have either program or data punched, complete a *Card and Paper Tape Preparation Request* (paragraph 2.4).   This form should also be used if cards are to be reproduced or paper tape edited.   Write your instructions clearly in ink and make any special comments you think will assist the operators to carry out your instructions correctly.

  (iv)   Write a *Job Description* (paragraph 2.2). and see that all documents are correctly titled and terminated.

3. COMPILERS

3.1   The Atlas operating system makes it possible for the machine to run programs written in any of a large number of languages, with very little trouble to either the programmer or the operators. Compilers for the most important languages are kept permanently on the Supervisor area of the disc and can be called with a single command. Others need a few extra commands.

3.1.1   Standard Compilers

The following are available directly from the system:

1.	IIF	10.	ALGOL
2.	IIC	11.	ABL
3.	SERVICE	12.	EMA
4.	TAD	13.	MAC
5.	SPECIAL	14.	EDIT
6.	AA	15.	COED
7.	FORTRAN4	16.	CHLF3 (London)
8.	HARTRAN	17.	CC
9.	HAR2	18.	LOAD (London)
		19.	FORTRAN (London)

As explained in paragraph 2.2.2.5 any of these is called by writing, for example:

                    COMPILER HARTRAN

or   USE ALGOL

in the Job Description.

The following brief notes give some guide to these compilers:-

1.   IIF and   2. IIC (Intermediate Input F and C): are rudimentary assembly languages, little used except by the engineers.

3.   SERVICE: is a collection of routines for listing, editing and similar operations with private tapes (see paragraph 4.5).

4.   TAD (Tape Addressing): is used only by the operators.

5.   SPECIAL: calls in any of the non-standard compilers, see paragraph 3.1.2 and Ref. 5.7.19. Also used for calling in a private compiler from a user's tape.

6.   AA (Atlas Autocode): is an ALGOL-like language developed by Brooker and Rohl of the Department of Computer Science at Manchester University. Features of the language include block structure, dynamic storage allocation and facilities for defining, constructing, and analysing tree

structures. Developments of AA, known as compilers AB and ABC, are also available at the Atlas Laboratory using Extra Compilers (3.1.2); see references 5.4.1, 5.4.2, 5.4.3.

7. FORTRAN4: the Manchester compiler for this form of FORTRAN. See Ref. 5.2.9.

8. HARTRAN: the FORTRAN system used as standard at Chilton. See paragraph 3.2.1 and Refs. 5.2.

9. HAR2: is equivalent to HARTRAN.

10. ALGOL: see paragraph 3.3 and Refs. 5.3.

11. ABL (Atlas Basic Language): assembly language. See Ref. 5.6.1.

12. EMA (Extended Mercury Autocode): has been developed from the language designed for use on the Mercury computer by R.A. Brooker, when at Manchester University. The changes needed for a Mercury Autocode program to run on Atlas are minimal and occur mainly at the beginning and end of the program. See Ref. 5.4.4.

13. MAC: equivalent to EMA.

14. EDIT and COED: editing programs written at Chilton. See Ref.5.7.9.
& 15.

16. CHLF3: a version of Mercury Autocode.

17. CC: the Brooker-Morris Compiler Compiler. See Ref. 5.4.5.

18. LOAD: a London program, for writing user programs to private tapes. See Ref. 5.7.20.

19. FORTRAN: Fortran V, the London version of Fortran. See Ref. 5.2.11.

3.1.2 Non-Standard Compilers: "Extra Compilers"

A number of compilers have been collected together from the London, Manchester and Chilton (S.R.C.) installations and put on to a single tape called EXTRA COMPILERS. This has been copied to the disc. Users wishing to use one of these compilers should write in their Job Description:

DISC n R501 EXTRA COMPILERS

and call the compiler through COMPILER SPECIAL.

Where two versions of the same compiler exist, as for example in the case of EMA, the addition of -M indicates the Manchester Compilers version, and of -L the London Compilers version. The correct name in this case is therefore EMA-M or EMA-L, which must appear after the COMPILER SPECIAL (or *KEEP-TAPE n directive if present).

A typical example would be:

                    JOB 10123 SMITH

                    . . .

                    . . .
                    DISC I R501 EXTRA COMPILERS

                    . . .

                    . . .
                    USE SPECIAL
                    EMA-M

Programs using Fortran will require the *KEEP-TAPE n directive, but users are requested to 'unload' the area as soon as they enter their programs so that others may use the compilers as soon as possible.

The compilers available, and their lay-out on the disc is as follows:

Compiler	Starting Block	Originating Tape	Date of Version
ACL	77	London	16. 2.68.
UPDATE	167	London	17.11.67.
CC-L	179	London	28.12.67.
SICTRAN	208	London	9. 2.68.
ABL	246	London	8.11.66.
EMA-L	270	London	2. 2.66.
SERVICE-L	310	London	10.11.67.
LOG-L	320	London	16. 2.68.
DEMI	330	London	7. 9.67.
IPL5	332	Chilton	24. 8.65.
LISP	353	Chilton	30.10.67.
SOL	368	Chilton	28. 3.68.
COCOA	390	Chilton	15. 2.68.
ABC)	397	Manchester	
AB)	398	Manchester	
IIC	448	Manchester	22. 4.64.
SERVICE-M	451	Manchester	1. 6.66.
IID	459	Manchester	22. 6.65.
LOG-M	461	Manchester	2. 2.66.
FOR2	490	Manchester	20. 9.66.
CC-M	522	Manchester	20. 6.66.
EMA-M	550	Manchester	26. 7.67.

Block I is a directory block.

3.2   FORTRAN

Because of the importance of Fortran — which accounts for about half of the time on Atlas — the use of this language is discussed at greater length.

A Fortran compiler and operating system was defined by A.E.R.E. Harwell and the Atlas Laboratory in collaboration and written by A.E.R.E. and the Laboratory with some assistance from I.C.L. It is called HARTRAN, and is basically Fortran II with extensions. This is the system used as standard at Chilton and is described in Refs. 5.2. Two other Fortran compilers are available in the group of standard compilers, the I.C.L. compiler for A.S.A. Fortran, which is effectively Fortran IV; and a compiler written by the University of London Atlas Computer Service called Fortran V.

References to Fortran literature are given in Section 5, and a note on the differences between Fortran IV and Hartran in paragraph 3.2.2.

3.2.1 Hartran

A Hartran program can be made up of any selection from three major sections:

    (i) source language routines in Fortran

    (ii) source language routines in ASP (Atlas Symbolic Programming), a mnemonic form of the machine assembly language

    (iii) routines in BAS (Binary and Arbitrarily Symbolic), which are binary forms of routines which have been previously compiled or assembled from Fortran and ASP routines

Details of these are given in Ref. 5.2.

Each Fortran routine must be preceded by a directive *FORTRAN, punched in columns I to 8 of a card, each ASP routine by a directive *ASP in columns I to 4, and each binary routine by *BAS, which is automatically produced by the compilation process.

The cards *FORTRAN, *ASP may contain further information, starting in column 9, for example, NOCARDS or LIST. The former will suppress the production of cards for the routine in question; the latter will produce a listing of the compiled program in a form similar to ASP, but should be used only if there is a real need for this as it is expensive in machine time.

These sets of cards are terminated by *ENTER, punched in columns I to 6, which causes the program, after the compilation and loading process is completed, to be entered at the beginning of the main program. *ENTER is followed by any data to be read by the program on stream 0, then an end-of-document marker, which is a card with 7 and 8 in column I and a terminator in column 80. The terminators are given in paragraph 4.6. The whole must be preceded by a suitable Job Description, and any data on any stream other than 0 supplied as a separate DATA document.

When this deck of cards is presented to the machine the Fortran compiler will process any Fortran routines, producing, on stream 0, a listing of the source program, information about the various identifiers used within the program, and also a list of the relative locations at which lines of the source program are situated. If the compilation is successful it will also produce, on stream

15, a set of cards, the first being a card containing

<div align="center">*BAS routine name</div>

and this is followed by several binary cards: these can be recognised by the presence of +79 punched in column 1, the last card being distinguished by the absence of the + punch. The user should check the serial numbering on columns 79 and 80 of all these cards, and if these are not sequential, for example 17 18 19 18 19 20, he should remove the *earlier* of the duplicated numbers, in this case the first 18 and 19, as these indicate a malfunction of the card punch. He should also check that the last card does not have the + punched in column 1. (If his own program produces a punched card output stream serial numbering of these cards also is recommended, so that any spurious cards may be removed). When the cards have been checked the original Fortran decks should be replaced by the BAS decks, if no changes in the source program are required. Similarly, the ASP assembler will process any ASP routines producing, on stream 0, a listing of the source program and, on stream 15, if the assembly is successful, binary cards preceded by *BAS. If a compilation or assembly is unsuccessful two or more cards will be produced, the last one bearing a visible FAILED punching. These cards should be discarded. The results of the compilation and assembly, together with the BAS routines in the deck, are automatically loaded into the Atlas store, any library routines being supplied by the system. A map of the layout of the store is produced and if no errors have been found the program is entered at the beginning of the main routine.

It will be seen from the above that output streams 0 and 15 must normally be specified in the job description, as both are used for the systems output. If stream 15 is not specified then the standard option for all subroutines is set to NOCARDS, and the compilations and execution, if any, will continue without any attempt to produce card output from the compilation.

There are several forms of the statements to obtain input and output in the execution phase, all of which are acceptable. The full set is, for input:

<div align="center">

READ(N,FMT) list   (1)
READ INPUT TAPE N, FMT, list   (2)
READ FMT, list   (3)

</div>

and for output:

<div align="center">

WRITE(N,FMT)   (4)
WRITE OUTPUT TAPE N, FMT,list (5)
PRINT FMT, list   (6)
PUNCH FMT, list   (7)

</div>

Forms (1) and (2) are treated identically, as also are forms (4) and (5). In these statements, N is either an explicit unsigned integer in the range 0 to 15 or an integer scalar variable whose value is in the same interval. The format specification is indicated by FMT, which may be the statement number

assigned to a FORMAT statement in the routine, or which may be the number of an array holding the text of the format specification at execution time.

The input statement (3) will always refer to data on input stream 0. The forms (I) and (2) will normally refer to input stream N, but if that is not defined the statements may refer to magnetic tape N of the job description.

Similarly, for output the PRINT statement refers to output stream 0, and the PUNCH statement to output stream 15. The other statements normally refer to output stream N, but again if this is not defined the statements may refer to magnetic tape N of the Job Description, which may be either Ampex 1" or IBM compatible ½" tape.

The following example shows the detailed lay-out. The spacing is important.

```
        JOB VAI23 J SMITH SIMPLE EXAMPLE
        COMPUTING 30 SECONDS
        STORE 10 BLOCKS
        OUTPUT 0 LINEPRINTER 200 LINES
        2 LINEPRINTER 10 LINES
        15 CARDS 50 LINES
        INPUT 1 VAI23 DATAI
        COMPILER HARTRAN
        *FORTRAN
                READ 100, I
                PRINT 101, I
                READ (1,100)J
                WRITE (2,101)J
                PUNCH 101, J
                CALL XXX(I,J)
                GO TO EXIT
        100     FORMAT (II)
        101     FORMAT (IX,II)
                END
        *BAS    XXX   ..... )      produced from a
        binary cards         )      previous compilation
        *ENTER
        2
        file card (7 and 8 in column 1, C in column 80)
        DATA
        VAI23 DATAI
        3
        File card (7 and 8 in column 1, Z in column 80)
```

The output on stream 0 to be expected from this job, apart from the effect of the routine XXX, would consist of a listing of the Fortran source program, followed by a map of the layout of the store, and a heading.

EXECUTION STARTED ON date AT time WITH THE INSTRUCTION COUNTER READING n

This would be followed by:

    (a) the number 2, printed in column I

    (b) on stream 2, the number 3 printed in column I

    (c) on stream 15, the binary cards from the compilation and a card
        with 3 punched in column 2.

3.2.1.1 Store and Time Requirements

These depend very much on the nature of the program, so only a rough approx-
imation can be given to start with. The End-of-Job Statistics (see paragraph
4.1) given at the end of a run will enable this to be improved and should
always be studied.

The following is a *rough* guide to the quantities involved in compilation:

COMPUTING = 1,000 source cards per minute

OUTPUT 0   = $1\frac{1}{2}$ x number of source cards

OUTPUT 15 = $\frac{3}{4}$ x number of source cards

STORE   = the standard allowance of 120 blocks is usually enough for
                a routine of up to 150 statements. If only AS^P or BAS
                cards are used (i.e. no Fortran source language) the amount
                of store is approximately 40 blocks more than the loaded
                program.

If we were to repeat the above job, replacing the Fortran routine by binary
cards, we would replace the relevant items in the job description by some-
thing similar to:

        COMPUTING 10 SECONDS
        STORE 10/50 BLOCKS
        OUTPUT 0 LINEPRINTER 50 LINES
        15 CARDS 3 LINES

The replacement of the 'STORE' description in particular is very advanta-
geous, to both the efficiency of Atlas and to the progress of the user's job
through the machine.

3.2.1.2 Errors

The Hartran system gives very extensive error diagnostics, described in
Refs. 5.2.2 and 5.2.3. Some errors are so common as to warrant noting here.

    (i) OUTPUT NOT DEFINED

         EXCESS BLOCKS

         OUTPUT EXCEEDED

         C TIME EXCEEDED

These usually indicate an inadequate Job Description.

      (ii) ILLEGAL FUNCTION

           SV OPERAND

If the dump of the program shows that there is an empty register (40 ... 0), check to see if a binary card is missing.

If the control address is near location 0 check to see if a computed GO TO is out of control, or again to see if a binary card is missing.

If the store appears to contain random data, check to see if an array is being subscripted outside its bounds and is overwriting the program.

If one of the lower-numbered B-registers contains a value with leading digit 6, check to see if one of the subroutine calls has the wrong number of arguments. This may cause an illegal access, or a print statement to terminate prematurely, followed by a jump to location −1.

      (iii) V TAPE ERROR

This can occur as a result of attempting to read more records from a binary tape than have been written. Or it can be caused by a faulty editing procedure. For example, if records 1 to 4 have been written and we wish to change record 4, we must backspace to the beginning of record 3, read a record, then write the new record 4. Failure to do this will cause sections of the information to be lost, when a subsequent read operation will give V TAPE ERROR.

      (iv) END TAPE

This diagnostic is often caused by trying to backspace a tape which is already positioned at its beginning.

3.2.1.3 Fortran Library

An extensive set of Fortran routines is held on the disc in the area R500 FORTRANL4B. A list of these routines, with instructions on how to access them, is given in Ref. 5.2.12.

3.2.2 Fortran IV

A user wishing to run a Fortran IV program·at Chilton can either consider whether the I.C.L. compiler for ASA Fortran, described in Ref. 5.2.9, will accept his program − as it will in many cases − or change to make it acceptable to Hartran. A note on the changes to be made is given in Ref. 5.2.7. The most serious are:

     (i) removal of any DATA and BLOCK DATA statements

     (ii) change of any logical IF statements.

         e.g. IF (A.GT.B) X = Y

            CONTINUE

of Fortran IV must be replaced by

IF (A.GT.B) 1,2   (or any labels in place
                          of 1 and 2 which are
                          not used elsewhere in
                          the routine)

1 X = Y

2 CONTINUE

and IF (C.EQ.D) GO TO 3

must be replaced by

IF (C.EQ.D) 3,

where the final comma is essential.

A conditional expression may need extra parentheses: thus the Fortran IV statement

IF (A.GT.B AND C.EQ.D) GO TO 123

must be changed to

IF ((A.GT.B) AND (C.EQ.D)) 123,

and again the final comma is essential.

3.3   ALGOL

The Algol system available on the Chilton Atlas consists of an Algol com-
piler coupled to a flexible input pre-processor. The Algol compiler accepts
Algol 60 with a few minor restrictions. Full details of the basic compiler
with the original I/O package are given in Ref. 5.3.1.

The philosophy behind the pre-processor developed at the Atlas Laboratory
is given in Ref. 5.3.2. The aim has been to improve the interface between
the user and the compiler. The most important part of the pre-processor, as
far as the user is concerned, is that it is capable of accepting Algol programs
produced originally for other British computers. This is achieved by speci-
fying three different I/O packages which can be automatically inserted into
an Algol program, allowing many different input forms.

The three I/O packages are:

    (i) ICT: the standard I/O package originally available with the
                compiler and described in Ref. 5.3.1.

   (ii) KDF9:   the set of I/O procedures developed by English Electric
                for the KDF9 and now available on the System 4 compu-
                ters as well.

  (iii) ELLIOTT: the I/O package available on the Elliott 803, 503
                  and 4100 range.

The Algol dialects currently accepted are:

a. ICT 7-HOLE TAPE   The original input form provided by ICT

b. ELLIOTT 5-HOLE TAPE The standard input for the Elliott 803

c. ELLIOTT 8-HOLE TAPE The standard input for the Elliott 503

d. KDF9 8-HOLE TAPE   The standard input for the KDF9

e. ATLAS CARDS   A space dependent card dialect used at the Laboratory

f. INTERNAL ICT   An improved version of a.

g. ELLIOTT CARDS   A card dialect defined for the Elliott 503

h. ICT 8-HOLE TAPE   The standard ICT 1900 tape input form

i. ICT 1900 CARDS   The standard ICT 1900 card input form

j. ICT 7-HOLE UC   An ICT 7-hole tape dialect using upper case to denote reserved words

k. ELLIOTT 4100 TAPE   The standard 8-hole tape input for the Elliott 4100 range.

l. SYSTEM 4 TAPE   The standard 8-hole tape input for the System 4 range.

This list is continually updated and the user should consult the most recent Algol papers for additional dialects. The Laboratory is willing to include any dialect in this list as long as sufficient use will be made of it to warrant spending the programming effort required to define it.

The particular input form and I/O package required is defined by a processor command at the head of the program. For example:

INPUT SYSTEM 4 TAPE WITH KDF9 I/O PROCEDURES;

Complete details of the Algol system are given in Refs. 5.3.3, 5.3.4 and 5.3.5. As well as accepting a variety of Algol dialects, the system also provides the following facilities:

(i) Optional listing of any part of the program on the lineprinter.

(ii) Conversion of a program to either the ELLIOTT CARDS or ICT 7-HOLE TAPE DIALECTS.

(iii) A filing system for storage of packed-up Algol text.

(iv) Editing of Algol programs or filed text by replacement, deletion and insertion of lines.

(v) A library of about 300 procedures defined as files is available and can be included individually in a user's program at his request.

A new graphic output package called GROATS (Graphic Output on Atlas for the SC4020) is also available. This provides a set of input procedures for using the SC4020 microfilm and hardcopy plotter installed at the Laboratory. These can be used with any of the existing I/O packages. A separate manual for the GROATS package is available (see Ref. 5.3.6).

4. GENERAL OPERATIONAL INFORMATION

4.1 END-OF-JOB STATISTICS

After the output from a program, the Supervisor will print out on Stream 0 information about the actual use of the machine. This provides a valuable guide to any changes which should be made to the Job Description for later runs of the program.

Consider the following example:

	REQUESTED	USED	COMPILE
INSTRUCTION INTERRUPTS	4800	2630	1638
COMPILE STORE	160	154	
EXECUTION STORE	70	45	
	DECKS	BLOCKS	WAITING
MAGNETIC TAPES	2	114	5
	STORE TIME	DRUM TIME	DECK TIME
	2841	8	42
COMPILER NUMBER	9		

INPUT	0	5	BLOCKS	READER 0
INPUT	5	3	BLOCKS	READER 0
OUTPUT	0	180	RECORDS	LINEPRINTER
OUTPUT	6	132	RECORDS	LINEPRINTER
OUTPUT	15	20	RECORDS	CARDS

This indicates that, of the 4800 instruction interrupts requested, i.e. 30 seconds, a total of 2630 were used of which 1638 were taken up in compiling.

Of the 160 blocks requested for compiling, 154 were used and similarly 45 of the 70 for execution.

Two magnetic tape decks were used with 114 blocks transferred and 5 seconds waiting time.

The total input amounted to 8 blocks.

The total output amounted to 312 lines and either 20 decimal cards or 17

binary cards, as in this case I record is equivalent to about 7/8 of a card.

"Store Time" is the integrated occupation of the store in block-seconds; similarly for Drum Time in sector-seconds and Deck Time in deck-seconds.

4.2 RE-STARTS

Some means of re-starting or of rescuing information, in case of a machine failure, must be included in all long-running programs, so that never more than 10 minutes machine time is lost. This is the maximum amount the Laboratory will credit the user with if there is a breakdown.

There is no standard recipe for re-starts. "Break Output", described in paragraph 4.3, is useful in this connection and various dumping facilities are available but must be used with caution. The general strategy and the details must be decided by the programmer, to suit each particular program. In all cases the re-start procedures should be planned along with the program, and not added as an afterthought.

4.3 "BREAK OUTPUT"

The standard output procedure is for the Supervisor to assemble all the output from a program on disc or tape and print it all after the run has ended. But if the programmer inserts a BREAK OUTPUT statement at one or more points in the program, the output then assembled will be printed or punched whenever the statement is encountered. This is operationally convenient, and therefore recommended if there is a large amount of output, say between 4,000 and 10,000 lines on the printer: for more than 10,000 lines, the output should first be written to a private tape (see paragraph 4.4). This facility can obviously provide a simple rescue procedure in certain cases if there is a machine failure.

The required statements, in the most commonly-used languages, are as follows. The Operation Request Card should state if BREAK OUTPUT is used.

    (i) *Hartran*

The following statement is inserted, usually prior to a PRINT instruction:

        CALL OUTBRK(N) where N is the output stream number.

    (ii) *Algol*

The following procedure should be inserted at the head of the program:

```
procedure  break output (n);
value n;  integer n;
begin  code
1300,  92,  —,  n;
1071,   0, 92,  0;
end;
```

The call would be, for example, break output (3).

(iii) *AA*

The following routine should be included in the program:

                   routine break output (*integer* n)
                   1300, 92, –, n
                   1071,   0, 92, 0
                   end

(iv) *EMA*

The following instruction should be included at a relevant point in the program:

                   1071,   0, 0, n

where n is the output stream number.

4.4   LARGE INPUT/OUTPUT JOBS: Private Tapes

If a program requires a large volume of input or will produce a large amount of output, the reading or printing operations should be done separately from the main run by means of private tapes. In general, all that is necessary is for the Operation Request Card to state the amount of input or output required; for example, 10,000 cards input or 40,000 lines output: the machine room staff will arrange the intermediate tape transfers.

However, a user may for one reason or another wish to organise these operations himself: for example, he may wish to build up an input file gradually as he gathers data, or to have a large amount of output from which to study parts from time to time. The way this is done is described in the following paragraphs, which explain how a user can:

    (1) copy information from any input medium to a magnetic tape, writing it to any part of the tape he chooses.

    (2) read information from a tape, written in this way, into a program

    (3) write output from a program on to a tape, again choosing where on the tape this shall go.

The object is to improve the efficiency of handling large volumes of input and output. The essential further step, of printing from a private output tape, is provided by COMPILER SERVICE as described in paragraph 4.5.04-Off-Line Output.

4.4.1   Data to Magnetic Tape

The document to be transferred can be read in from any peripheral. It must be headed thus:

      COPY TAPE *b* (*b* = block number at which writing is to start)

      title of tape

      one or more new lines or dummy record

or   PRIVATE INPUT *b*

 title of tape

 one or more new lines or dummy record.

Nothing else is required; the Supervisor transfers the material to the named tape starting at the stated block. The "new lines" or dummy record is effectively the title of the document and is not copied to the tape.

When reading has been completed the tape is automatically re-wound and disengaged and information showing that the operation is complete is put out on the operators' teleprinter. The user will be informed of the next available block on the tape, so that he can plan further writings to the same tape, and so if he wishes build up a large file in stages.

4.4.2 Data Input from a Private Magnetic Tape

Data recorded on magnetic tape by the above process can be called by a program by means of the following statements in the Job Description

<div align="center">

INPUT TAPE *a/b*/o

n with any title

TAPE *a* title of tape

Here *a* is the logical number of the tape

b is the address of the starting block

n is the number of the input stream
</div>

The title following *n* is any identifier the user finds convenient and is not checked with anything written on the tape.

The title following *a* is, as usual, the title in Block 0 of the tape and is checked.

4.4.3 Program Output to Magnetic Tape

This is achieved by declaring an output stream in the Job Description as follows:

 OUTPUT *n*   TAPE *b* /   (final destination)(amount)
 (title of tape)

The output will then be written to the named tape, starting at block number *b*

  e.g. OUTPUT 3 TAPE 100/LINEPRINTER 10000 LINES

   N0123 JONES RESULTS

The Supervisor will write as the first two records:

  OUTPUT *n*

  (title of the job which produced the output)

The record can be printed from the tape, or put out on whatever other device

has been specified, by the procedure of paragraph 4.5.4.

4.4.4   Further Use of Private Output Tapes

A private output tape can be used also in three other ways:

(i) *as input to the same program*

It must be re-named as an input tape, using Extracode 1070. It may then be used simply as a normal input stream, but it must be remembered that there are the two extra records at the beginning.

(ii) *as input to a different program*

It is declared in the Job Description of the subsequent program as in paragraph 4.4.2. It need not be re-named as input. In using an output tape as input, one has to remember the presence of the two heading records.

(iii) *as a working tape in the same program*

If the tape is declared as output stream n it can be used in this way by referring to it as tape $110 + n$. For example, private output tape 4 becomes tape 114. But great care must be taken over the positioning of the tape. If it is used as a working tape before the output is written, it must be re-positioned (by means of Extracode 1001) to the starting block given in the Job Description. If it is used in this way as an interlude in the output, the block to which the output has taken it must be found by the "where-am-I" Extracode (No. 1024) before anything else is done, and the tape re-positioned to this before any more output is written.

To use a tape safely as a working tape after using it for private output, the stream should be broken and thus killed, and also the "where-am-I" extracode (1024) must be used, to ensure that the tape queue is emptied. This ensures that the remnant of the stream is written to the tape, and that no further transfers will then take place. This is particularly useful if the beginning of a tape is used for variable data, say, and the top is used for off-line output.

4.5   COMPILER SERVICE: listing, copying operations etc. of general use.

4.5.01   Introduction

There is a set of standard programs for the following operations:

CARD LIST *	4.5.02
PAPER TAPE LIST *	4.5.03
OFF LINE OUTPUT	4.5.04
EDIT TAPE	4.5.05
COPY TAPE	4.5.06
MAG TAPE LIST	4.5.07
COMPARE MAG TAPE	4.5.08

CHECK VARIABLE TAPE *	4.5.09
RENUMBER	4.5.10
SELECT INPUT	4.5.11
SELECT OUTPUT	4.5.12
BREAK OUTPUT	4.5.13
COPY DOCUMENT	4.5.14
FIND DOCUMENT	4.5.15
WHAT DOCUMENT	4.5.16
COMPILER *	4.5.17
END *	4.5.18
PLEASE *	4.5.19

Other services provided are included in this list for completeness but should not normally be used except by systems programmers or the operating staff.

CHANGE SUPERVISOR

COPY OFF SUPERVISOR

COPY ONTO SUPERVISOR

COPY SUPERVISOR

EXIT

FAST COPY

IDENTIFY SUPERVISOR

OPERATOR

PM *

SPRINGCLEAN *

SUP TO DISC

Most of the titles are self explanatory and more details follow below.

It will be seen by the example given below, that after obeying one service, a further service may be entered. Those services marked * must appear last in the string of commands as they terminate the job.

In general, a command (e.g. MAG TAPE LIST) where the first 8 letters are significant and spaces are ignored is followed by a parameter list which consists of a string of numbers separated by spaces or tabs. Backspace may not appear anywhere.

Any of these services can be called by a Job Description which specifies

COMPILER SERVICE

or USE SERVICE

The amount of COMPUTING TIME required is negligible and 30 SECONDS should be more than adequate for most purposes. EXECUTION TIME may be significant in those services using magnetic tapes. The STORE request of 12 BLOCKS should be sufficient for most services and it is suggested that users consult the job statistics which give the actual number of execution blocks used.

We may have for example

                JOB 10000 BRYANT COPY AND LIST TAPE
                OUTPUT 0 LINEPRINTER 800 LINES
                STORE 12 BLOCKS
                TAPE 1   N0000 MASTER* INHIBIT
                TAPE 2 N0001 COPY *   PERMIT
                COMPUTING 30 SECONDS
                USE SERVICE
                EDIT TAPE
                2 20 1 30 5
                MAG TAPE LIST
                0   2 20   22
                1   2 23   24
                END
                78   Z

This will copy 5 blocks from tape 1, blocks 30 to 34 inclusive, onto tape 2 blocks 20 to 24, then print blocks 20 to 22 of tape 2 in octal and print blocks 23 to 24 of tape 2 as instructions.

4.5.02 Card List

This service is used for listing punched cards. The control statement
                CARD LIST
is followed by the data to be listed, and terminated by the control statement
                END LIST
A new page may be obtained by inserting
                NEW PAGE

The END LIST suppresses the INPUT ENDED monitor and gives a page-throw on the lineprinter.

HARTRAN binary cards are decoded to some extent according to the BAS format.

This service is also useful for duplicating cards (not binary) or paper tapes, by defining output stream 0 for card or paper tape output instead of lineprinter output.

4.5.03 Paper Tape List

This service will list paper tape punched in any standard code (not binary). It treats all TAB directives correctly and omits erases, there should be no back-space symbols on the tape. The control statement
>PAPER TAPE LIST

is followed by the tape to be listed which must be terminated by ***Z.

This service should not be used for duplicating tapes as tabs will be replaced by spaces. The service CARD LIST should be used for this purpose with paper tape as output 0, as this is a straight copying operation.

4.5.04 Off Line Output

This service enables any part of a private output or input tape to be output on any specific peripheral. The control statement OFF LINE OUTPUT must be followed by four parameters.

>OFF LINE OUTPUT
>A   B   C   D

will cause block C to block D inclusive on logical tape B to be output by the device defined by parameter A. If D = 0 the output will cease at the end of the document.

A = 1 lineprinter 0	A = 6 lineprinter 1
A = 3 5-hole tape punch	A = 7 card punch 0
A = 5 7-hole tape punch	A = 8 sigma 2 link
	A = 9 card punch 1

Only one document may be output from a tape at one time. If an attempt is made to output a second document from the tape, then the diagnostic message TAPE NOT DEFINED will occur. However, a further document may be output from a second tape using *another* device number. The following example will output documents from tapes 4 and 5 on the lineprinter.

>OFF LINE OUTPUT
>1   4   1   0
>6   5   1   0

4.5.05 Edit Tape

This service does a straight copy, block by block, from one magnetic tape to another, without subsequent checking. The control statement EDIT TAPE is followed by one or more lines each of five parameters.

                        EDIT TAPE
                        A B C D E

will cause E blocks to be read from logical tape C starting at block D and then written to logical tape A starting at block B.

4.5.06 Copy Tape

This service is for copying an ordinary user tape efficiently and with vigorous checks. The user's tape should be specified as TAPE 126 in the Job Description. The control statement COPY TAPE is followed by one or more lines of two parameters.

                    JOB ......

                    . . .
                    TAPE 126 N0123 SOURCE* INHIBIT
                    TAPE   I N0456 COPY
                    TAPE   2 N0457 COPY

                    . . .
                    USE SERVICE
                    COPY TAPE
                    AI   BI
                    A2   B2

                    . . .
                    END
                    78   Z

This will copy blocks AI to BI inclusive, A2 to B2 inclusive, etc., from TAPE 126 to up to seven tapes listed in the Job Description, with numbers in the range I to 7. After copying the tapes are read backwards and checked. Successful copying will be concluded by the message TAPES OK. If tape failure occurs, copying will continue without the faulty tape where possible.

4.5.07 Mag Tape List

This service enables the information on a magnetic tape to be printed in one of four forms; as octal numbers, machine instructions, characters or as hexadecimal numbers. The control statement MAG TAPE LIST is followed by one or more lines of four parameters.

                    MAG TAPE LIST
                    A   B   C   D

will cause blocks C to D inclusive to be printed from logical tape B in the mode defined by A;

A = 0 for octal listing
A = 1 for machine instructions
A = 2 for alphanumeric characters
A = 3 for hexadecimal listing

4.5.08 Compare Mag Tape

This service will compare one magnetic tape with another and print any differences. The control statement COMPARE MAG TAPE is followed by one or more lines of six parameters.

```
            COMPARE MAG TAPE
            A   B   C   D   E   F
```

will compare logical tape A starting at block B, with logical tape C starting at block D, continuing for E blocks. Only the first F differences in any set of differences in a block are printed. The count of differences is reset to zero by finding 2 words which compare successfully, or by the end of a block. To obtain no printing suppression, F should be specified as 512 or greater.

4.5.09 Check Variable Tape

This service will check that a tape allegedly written in variable tape mode is correctly written. Any error will cause a dump of the blocks around the faulty one for subsequent analysis, and the check will terminate. The tape to be checked must be logical tape 1 in the Job Description. The control statement CHECK VARIABLE TAPE will check tape 1 commencing at block 1.

4.5.10 Renumber

This control statement RENUMBER followed by two parameters A and B, re-numbers logical tape A as logical tape B (where A, B S 110). This is useful for services such as COPY DOCUMENT which refers to absolute logical tape numbers.

For example

```
            . . . .
            TAPE 1  N0001  A TAPE
            TAPE 2  N0002  B TAPE
            . . . .
            USE SERVICE
            . . . .
            RENUMBER
            1   3
            2   1
            . . . .
            END
            78                    Z
```

will renumber N0002 B TAPE as logical tape 1.

4.5.11   Select Input

This control statement is followed by a single parameter A, which defines the stream number for all succeeding input operations.

4.5.12   Select Output

This control statement is followed by a single parameter A, which defines the stream number for all succeeding output operations.

4.5.13   Break Output

This control statement is followed by a single parameter A. This has the effect of 'breaking output' on stream A, so that in the event of a breakdown, all the output is not lost.

4.5.14   Copy Document

This service will copy documents from one magnetic tape to another and may be used to link documents together. The control statement COPY DOCUMENT is followed by one or more lines of three parameters A, B, C.

```
          COPY DOCUMENT
          A      B      C
```

will copy a document from logical tape 1 starting at block A, onto logical tape 2 starting at block B. The end of the document is linked to C. If C = 0 this is marked as the end of the document; if C = 6000 it is linked to the next available block on tape 2.

For repeated use of this service, the parameters A, B can be used as follows.

If A = 6000 continue reading from tape 1 according to the last link read. This is useful for concatenating an output document with several breaks into one document.

If A = 0 do not read from tape 1, instead set up a dummy block containing only an end marker.

If B = 0 continue on tape 2 after the end of the previous document.

The following example copies three consecutive documents (output breaks) from tape 1 starting at block 1, making them into one document on tape 2 starting at block 10, with an end marker to terminate.

```
          JOB 1000 BRYANT
          COMPUTING 30 SECONDS
          OUTPUT 0 LINEPRINTER 1000 LINES
          STORE 12 BLOCKS
          TAPE 1 N0001 A TAPE
          TAPE 2 N0002 B TAPE
          USE SERVICE
```

```
              COPY  DOCUMENT
              I    IO    6000
              6000  0    6000
              6000  0       0
              78                              Z
```

4.5.15   Find Document

This service is intended for finding out what documents (if any) are on the
tape. The control statement FIND DOCUMENT is followed by two parameters
A, B.

              FIND DOCUMENT
              A   B

prints the first 18 words of block A to block B inclusive, in octal, on logical
tape 1, and also the first record (approx.) in character form of each block.

4.5.16   What Document

This control statement WHAT DOCUMENT may be followed by one or more
single parameters A. It lists the first block of the documentson logical tape 1
at block A, finds the last block and prints its block number. It also prints
the title of tape 1.

4.5.17   Compiler

The purpose of this service is to allow the user to enter a compiler of his
choice (probably after performing other services). We may have

              USE SERVICE
              . . .
              COMPILER
              A   B
              . . .

A and B are the n address and the contents of Ba for the 1147 (call compiler)
instruction. This will call compiler A and enter at B, having unset the process
switch. If B = 0 the standard entry point to the compiler is used. The ABL
manual should be consulted for further details.

The complete list of compilers on the Supervisor follows.

Compiler Name	Number
ON	13
IIF	5
IIC	I
SERVICE	19
TAD	7
SPECIAL	29

AA	4	EDIT	14
FORTRAN 4	15	COED	16
HAR 2	10	CHLF 3	31
HARTRAN	9	CC	6
ALGOL	12	LOAD	36
ABL	2	FORTRAN	34
EMA/MAC	3		

4.5.18 End

This service terminates execution and is normally the last control card in any service job (apart from those commands which terminate the job).

4.5.19 Please

This simply ends the program with a polite rejoinder HAPPY TO OBLIGE.

4.6 WARNING SEQUENCES

These are used either to define different parts of a document or to mark its end. On paper tape they consist of three asterisks followed by a letter or a decimal digit and on cards by a 7 and 8 multi-punched in column I and the letter or digit in column 80.

Sequences with letter or digit indicators are given below with the appropriate command or action to be taken.

***A   ignore the documents up to this point

***B   read following information in binary up to the end of the document

***C   end current document and read next one

***E   read following information in binary up to the end of the document marker

***F   disengage the reader and, on re-engaging, read in binary up to the end of the next document

***I   (discontinue reading in binary and) continue to read document in internal code

***P   suppress parity checking up to next end of document marker

***T   disengage the reader and, on re-engaging, continue reading in the same mode

***Z   end document and disengage reader (blank in column 80 is equivalent to Z)

***n   (where n can have a value between 0 and 9)
disengage the reader and, on re-engaging, read in the code given in the following table:

10/69

Code No.	2	3	4	9
Cards	1900	360 EBCDIC	6600	Standard Atlas
5 Hole	Elliott 803	Telex	Standard Atlas	Standard Atlas
7 Hole	1900	1500 Creedomat	Friden	Standard Atlas
8 Hole	1900	Elliott 503	KDF9	1900

4.7 CODE OF PRACTICE

The following notes are offered in the way of advice and caution to the user. Many of them only repeat what has been said in other parts of this Handbook, but observations of the difficulties in which users can find themselves suggests that this repetition may serve a useful purpose.

General

(a) Fill in the operation request card completely and correctly — it has not been uncommon for this to disagree radically with the Job Description.

(b) Give the following information, which can be of great help to the Laboratory staff:

   (i) the amount of output if it is large;

   (ii) if break output and restart facilities have been incorporated;

   (iii) if any off-line processing is required; and

   (iv) the address to which results should be sent.

(c) Do not, unless really necessary, request multiple copies of output.

(d) Before departure, leave information about any jobs still in the computer in a book held by External Reception.

(e) Do not use the same Job Description automatically for every run of the same program; such details as time, store and amount of output can usually be reduced.

(f) Keep a duplicate copy of each job to be used in the event of loss or of damage beyond repair.

(g) Do not submit identical copies of the same program; e.g. compilation of *same* routine in several runs.

(h)   Do not define output streams in the Job Description which are not in fact required.

(i)   Take care over carriage control on output of results so that disasters such as one line per page do not occur.

Cards

(a)   Punch the Job Description on special white cards.

(b)   Interpret the Job Description at least.

(c)   Remove extra cards produced by the system at the beginning and end of BAS decks before re-input.

(d)   On a job of more than ten minutes' duration, a Fortran program should consist only of BAS cards (i.e. it should already have been de-bugged).

Magnetic Tape

(a)   Restrict magnetic tape titles, including the Atlas Laboratory number, to fourteen characters.

(b)   Remember that identification of a tape is made by this title alone, character for character including spaces.

(c)   Inform the Magnetic Tape Librarian at the Laboratory as soon as a tape is no longer required.

(d)   Take particular care over the identification of IBM tapes as these have no titles recorded at the beginning of the tape and hence there are no means of checking the correctness of the tape used.

Paper Tape

(a)   Before use, check for quality and alignment of sprocket holes.

(b)   Leave a blank header of about twelve inches length.

(c)   If tapes have to be loaded in a particular order, mark them clearly with numbers and put a note on the Operation Request Card.

(d)   Make a note of the terminator on every tape.

(e)   Fix a true printed copy of the Job Description to the tape box.

(f)   If all eight holes are to be read on eight-hole tape, make a note on the Operation Request Card.

(g)   If output tapes are to be put in again, check the beginning and end for diagnostic information produced by the System.

5. LITERATURE

A key to the availability of these documents (A, B ...) is given at the end of this section.

5.1   GENERAL USE OF ATLAS

 5.1.1 Users' Handbook   A

 5.1.2 Preparing a Complete Program for Atlas I. March 1966   A

These two documents should both be read before any attempt is made to use Atlas.

5.2   HARTRAN AND OTHER FORTRANS

 5.2.1 Atlas Fortran Manual (Part I) E.J. York.   AERE/R4599
     1964   5/6d.   (There is no Part II)   E

 5.2.2 Hartran System Note 4.   March 1966   A

 5.2.3 Revised Appendix III to Hartran System Note 4. Sept 1966 A

 5.2.4 INTERASP, an Intermediate ATLAS Symbolic Programming
     Language.   AERE/R4285   1963   5/ —   E

 5.2.5 A Primer of Fortran Programming for use on the Atlas and
     Orion Computers.   CS 390   Oct 1963   18/ —   B

 5.2.6 Making a Fortran 2 Program Suitable for use with the
     Atlas Fortran Compiler.   CS 318B   F

 5.2.7 A Guide to the Conversion of a Fortran IV Program to
     Atlas Fortran   F

 5.2.8 Usage of Magnetic Tape in the Hartran System   F

 5.2.9 Users' Guide to Fortran 4 on Atlas I   F

 5.2.10 Hartran System Note I: Routine Linkage Conventions
     within Hartran   A

 5.2.11 Fortran V.   C.F. Schofield. 32/6d. +   C
          3/ — postage

 5.2.12 Atlas Hartran Subroutine Library   A

The potential user of Hartran should be familiar with 5.2.1., 5.2.2., and 5.2.3. 5.2.1 describes the Fortran language in use, 5.2.2 and 5.2.3 describe the use of Hartran as a system on Atlas, and in particular give lists of errors. 5.2.3 gives advice on how to interpret execution phase errors. The beginner is

strongly recommended to read 5.2.5 before attempting to understand 5.2.1, 5.2.2 and 5.2.3. 5.2.4 describes the machine language which may be used within the Hartran System. It must be used in conjunction with the I.C.T. Atlas I Programming Manual (5.6.1). 5.2.12 gives details of the subroutine library available under Hartran.

5.3 ALGOL

5.3.1	I.C.T. Atlas I Computer Algol Reference Manual. February 1966, reprinted January 1968. CS 378B 14/ —	B
5.3.2	The Atlas Algol Preprocessor for Non-standard Dialects F.R.A. Hopgood and A.G. Bell, The Computer Journal, February 1967	
5.3.3	The Atlas Algol System	A
5.3.4	The Listing, Editing and Library Facilities for the Atlas Algol Compiler (Algol Paper 11)	A
5.3.5	The Standard Algol Library Tape, ALGOLIB (Algol Paper 12)	A
5.3.6	GROATS: a Graphic Output Package for the Atlas using the SC 4020	A
5.3.7	ACM Algorithms	G
5.3.8	Revised Report on Algol 60. CS 377 6/ —	B
5.3.9	Primer of Algol 60. CS 379A (5068) 16/-	B

5.4 OTHER HIGH-LEVEL LANGUAGES

5.4.1	Atlas Autocode Reference Manual. March 1965. 7/6d.	D
5.4.2	Atlas Autocode Mini-Manual. July 1965 Supplied free of charge with 5.4.1	D
5.4.3	Various small papers on Compilers AA and AB. Free	D
5.4.4	Extended Mercury Autocode for Atlas CS 402/5052. 60/ —	B
5.4.5	The Compiler Compiler R76. Reprinted from Annual Review in Automatic Programming Vol. III. 1963. 3/ — (Stocks not available at present: will reprint on request)	B

5.5 PLOTTERS

5.5.1	SC 4020 Plotter Manual. September 1967	A
5.5.2	The Graph Plotter (2nd Edition). September 1966	A

(Limited supplies are kept for reference only)

5.8 MISCELLANEOUS

5.8.1 Some Aspects of Current Operation and Research. 1966   A

5.8.2 Some Research Applications of the Computer, 1968.   A

5.8.3 Atlas Computer Laboratory Brochure   A

The above documents may be obtained according to the following key:

A From the Atlas Laboratory Librarian (University users *must* apply through their local university representative).

B From: Technical Publications Department,
                  International Computers Limited,
                  30/31 Friar Street,
                  Reading, Berks.

on payment of the appropriate charge.

C From: Information Officer,
                  University of London Atlas Computing Service,
                  39 Gordon Square,
                  London W.C. 1

on payment of the appropriate charge.

D From: Computing Service,
                  Department of Computer Science,
                  The University,
                  Manchester 13.

E From: H.M.S.O.
                  49 High Holborn
                  London W.C.1

on payment of the appropriate charge.

F These documents, if required, may be obtained from the Support Group, Atlas Computer Laboratory, in writing. *Only limited supplies are kept.*

G This large document is kept at the Laboratory, and particular algorithms will be photocopied on request. A list of the algorithms appears in 5.3.5.   Application should be made to the Support Group in writing.

H From: The Library
                  Building 465
                  AERE Harwell
                  Didcot
                  Berkshire

J From: Programming Advisory Service,
                  University of London Institute of Computer Science,
                  44 Gordon Square,
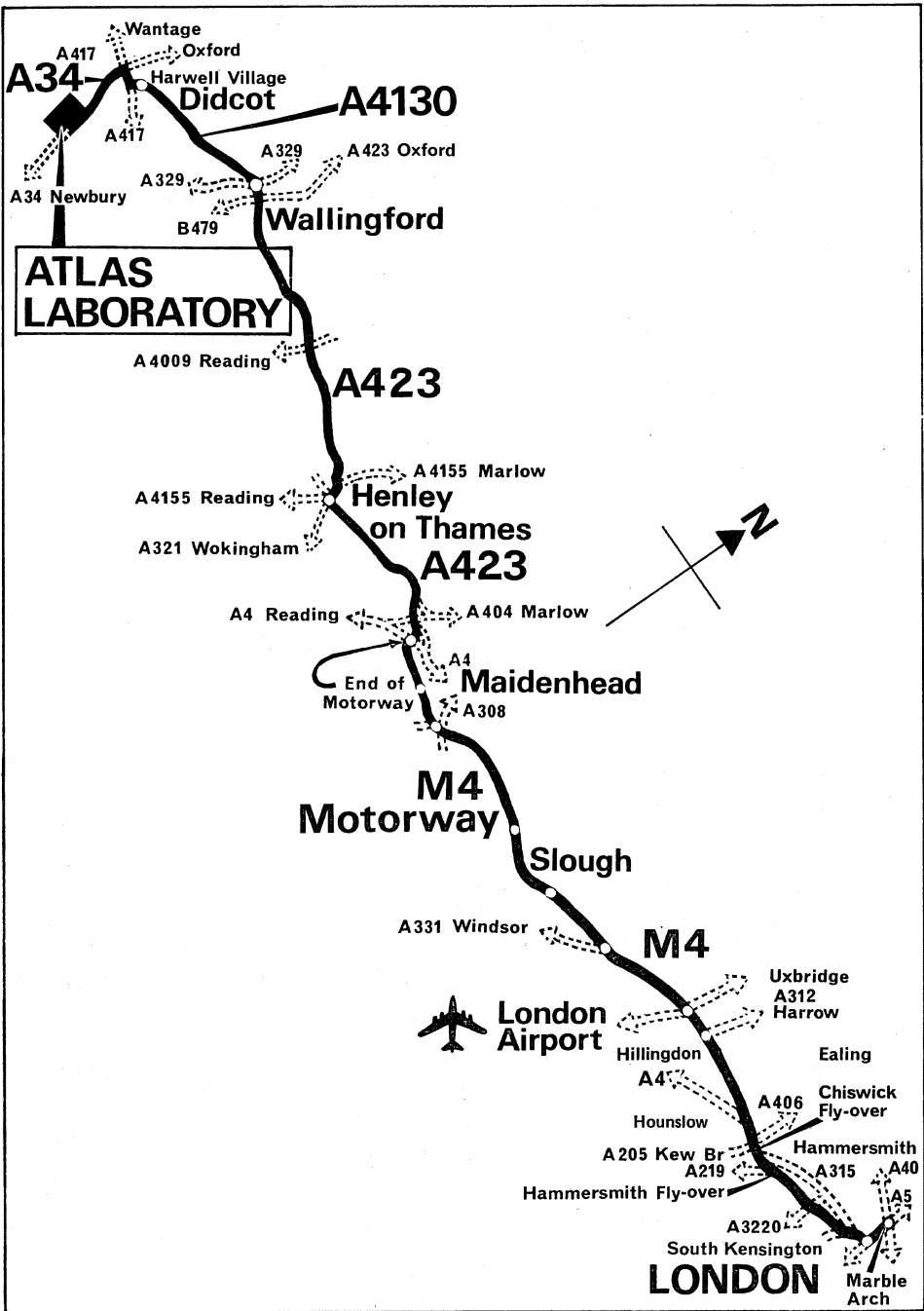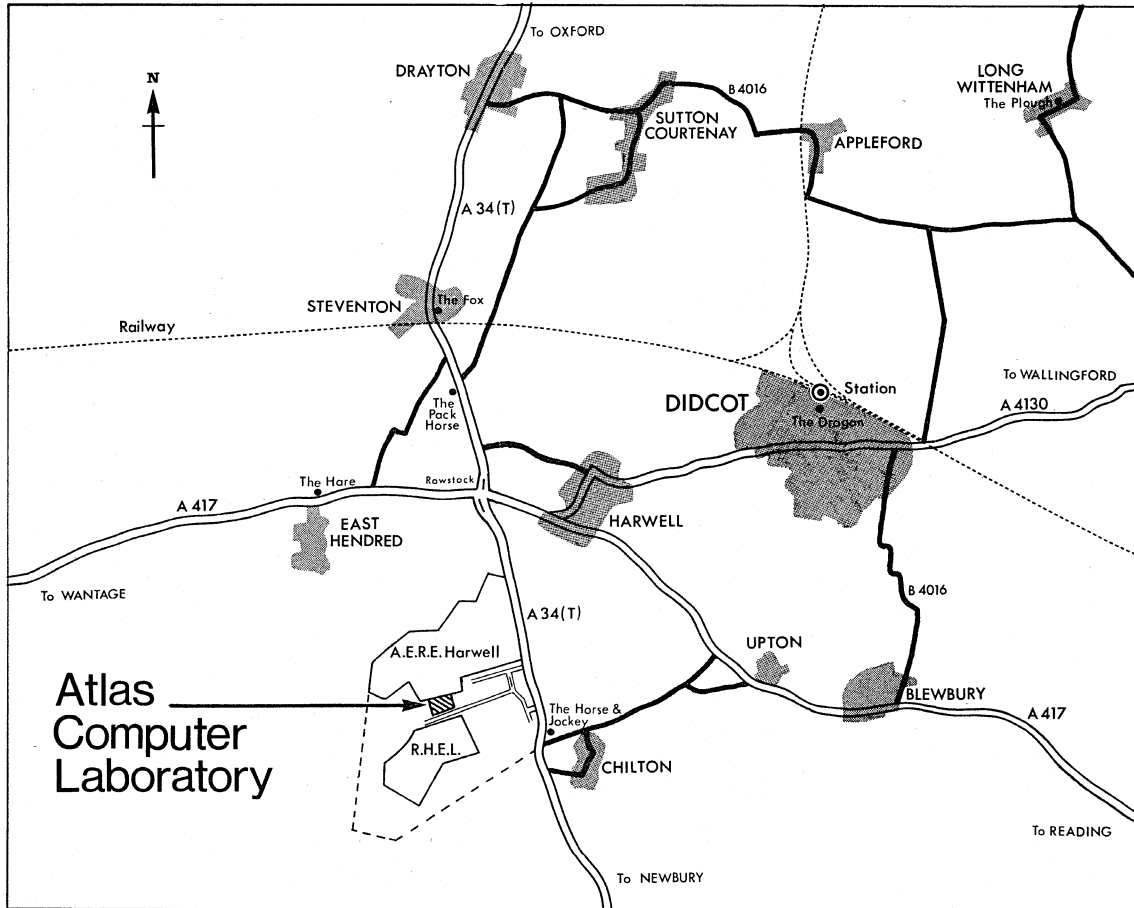   London W.C.1.

6. TELEPHONE NUMBERS, MAPS
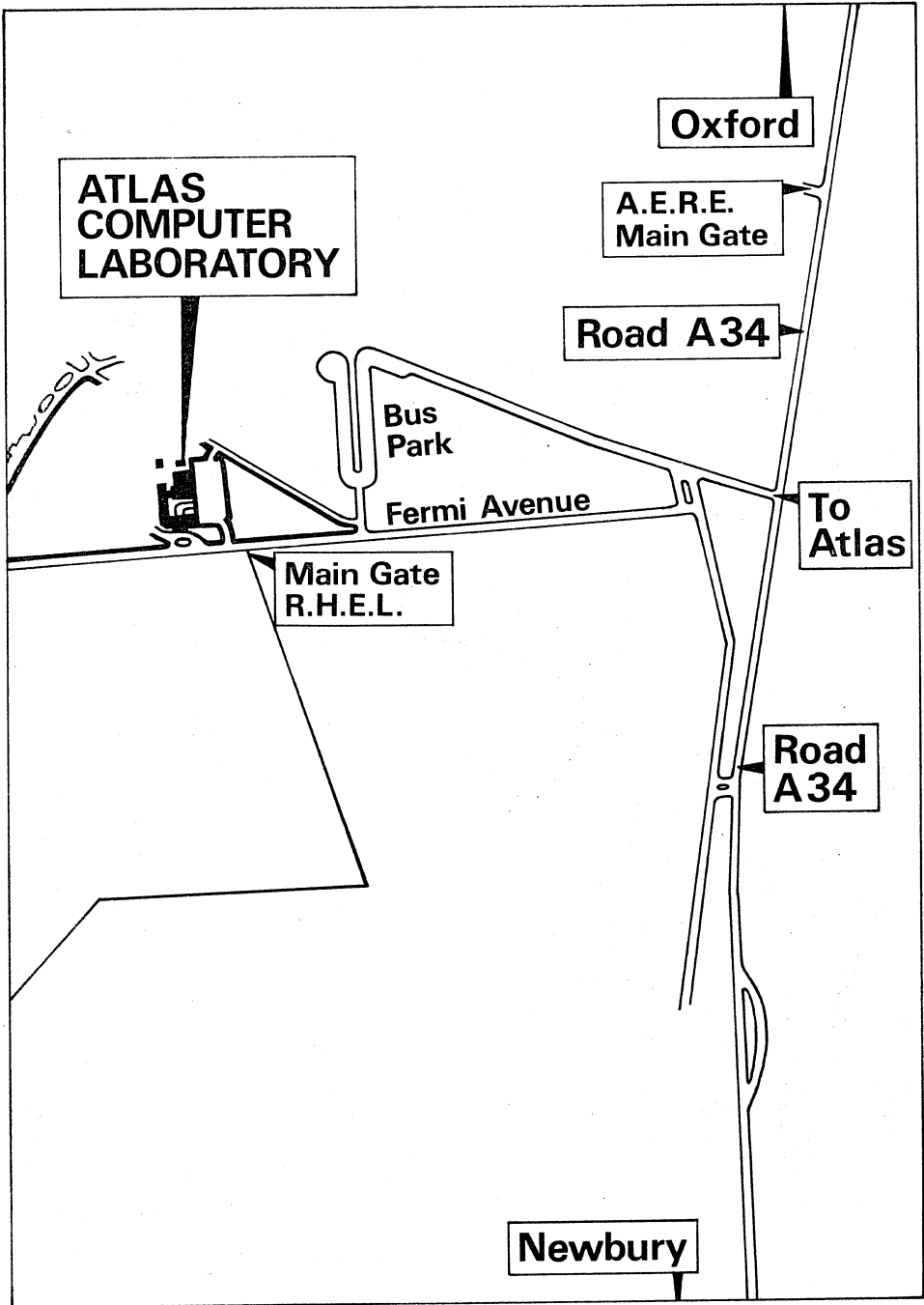
SOME LABORATORY TELEPHONE EXTENSIONS
WHICH MAY HELP USERS

Exchange Number .. Abingdon 1900

Director .. .. .. Extension 547
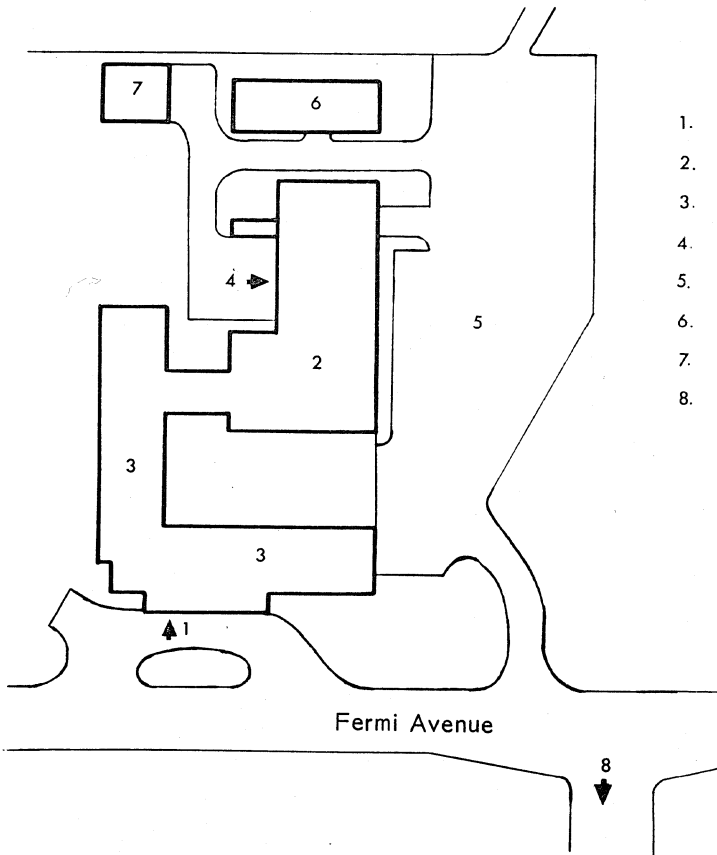
		Extension
(1) *Technical Enquiries*		
Head of Programming Group		6104
Head of Operations Group		6103
Head of Support Group		6121
	and Rowstock	471
General University and External User Program Enquiries		6193
Librarian		6226
(2) *Operational Services*		
Head of Data Preparation Section Application for University Job Numbers (urgent only)		6284
External Reception		429
Receipt and Despatch of Work Requests for Operational Stationery (can also be obtained on direct line:	Rowstock	321)
Magnetic Tape Librarian		6640
(3) *Administration*		
Laboratory Administrative Officer		6106
Front Hall Reception Users are asked to call this number for accommodation and transport bookings		6296

N

To OXFORD

DRAYTON

B 4016

LONG
WITTENHAM
The Plough

SUTTON
COURTENAY

APPLEFORD

A 34 (T)

STEVENTON   The Fox

Railway

The
Pack
Horse

DIDCOT   Station   To WALLINGFORD

The Dragon   A 4130

A 417   The Hare   Rowstock

EAST
HENDRED   HARWELL

To WANTAGE

A 34 (T)   B 4016

UPTON

A.E.R.E. Harwell

Atlas
Computer
Laboratory

The Horse &
Jockey   BLEWBURY   A 417

R.H.E.L.   CHILTON

To READING

To NEWBURY

ATLAS
COMPUTER
LABORATORY

Oxford

A.E.R.E.
Main Gate

Road A34

Bus
Park

Fermi Avenue

To
Atlas

Main Gate
R.H.E.L.

Road
A34

Newbury
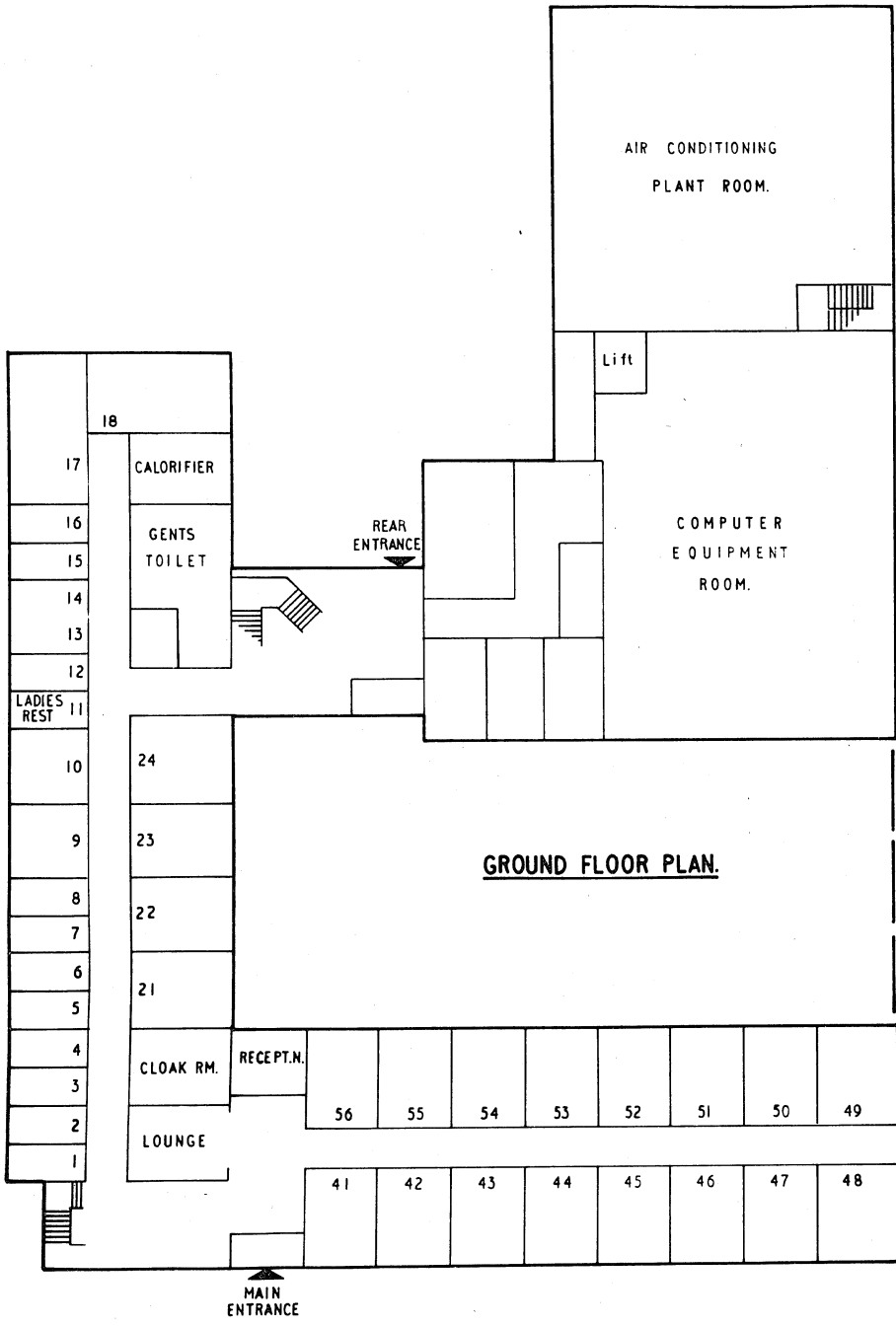
1. Main Entrance
2. Computer Block
3. Office Block
4. Lift Entrance (Goods)
5. Car Park
6. Alternator House
7. Garages
8. Main Entrance R.H.E.L.

Fermi Avenue

PLAN OF THE BUILDING

AIR CONDITIONING PLANT ROOM.

Lift

REAR ENTRANCE

COMPUTER EQUIPMENT ROOM.

18

17

CALORIFIER

16

GENTS TOILET

15

14

13

12

LADIES REST 11

10

24

9

23

8

22

7

6

21

5

4

CLOAK RM.

RECEPT.N.

3

2

LOUNGE

1

GROUND FLOOR PLAN.

56 55 54 53 52 51 50 49

41 42 43 44 45 46 47 48

SIDE ENTRANCE

MAIN ENTRANCE

C A R   P A R K.

DARK ROOM

ANCILLARY EQUIPMENT ROOM

COMMON ROOM

Lift

COMPUTER ROOM

KITCHEN.

LADIES TOILET

DATA PREP.

I. C. L.

ENG. R.

35

34

'THINK' ROOM

33

40

32
31

39

30
29

38

FIRST FLOOR PLAN.

28
27

37

26

ARCHIVES

SMALL CONF.CE.

25

64

63

62

61

60

CONFERENCE ROOM

57

58

59

LIBRARY

102640701