

Addressing magnetic tape on the Atlas 1 computing machine

By D. B. G. Edwards*, D. Aspinall* and R. Lane†

A consequence of the pre-addressed fixed-length data block format which the Atlas magnetic tape system uses is that tapes with this layout have to be provided from 'new' tapes before the system can even operate. To avoid the necessity of building expensive and elaborate off-line equipment, small modifications were included in the design of the basic Atlas tape system to provide this necessary facility. The process is performed using Atlas itself, but because of the computer's time-sharing techniques only a small fraction of the computer's capability is used up. The paper provides a factual description of the tape testing and addressing facilities, explaining the reasons for the major decisions involved in the design.

Introduction

In any magnetic-tape digital recording system means must be provided for the recovery of previously recorded data, and typically the tape is subdivided into regions called *blocks*. Location of any item of data can be achieved using a simple counting process, provided that a previous record has been made of its count. The process can be further facilitated by the inclusion of a co-ordinate or *block address* which uniquely defines the absolute position of the block of data with which it is associated.

The ordinary computer user would prefer a system which allows blocks to be of variable size without restriction, but in practice there are certain limitations. If the block is very short then, because of the inter-block gap which is a function of the start/stop performance of the tape mechanism, the efficiency of tape usage is poor, e.g. at packing densities of 400 digits per inch 128 words occupy less space than the inter-block gap. The upper limit to block length, though basically defined by the tape length, is normally limited by the size of the computer's main store into or from which data transfers take place. To achieve system compatibility within a range of computers the block size is limited by the smallest main store and is unlikely to be greater than 4096 words.

These restrictions are already quite severe, and advantages can be gained by adopting a fixed block length; on Atlas a 512-word block is always used. Tapes can now be addressed at the same time as they are tested. Faulty areas of magnetic tape are detected and left blank. This saves repeated attempts during the life of the tape to record information in these areas, and therefore reduces the wastage of machine time. Individual blocks in a record can be changed without copying the entire record, and the system has the capability of being used as a backing store. The decision to adopt a fixed block length was taken for engineering and economic reasons in that the system performs efficiently even with imperfect tape. However, the user has been isolated from any bad effects of this decision by providing software, namely certain magnetic-tape

extracodes, which permits him to specify variable-length blocks. Such blocks are always packaged by the program into 512-word units.

Testing of tapes cannot anticipate the faulty regions which develop while the tape is in use or being stored. Such regions have to be identified in use and, since it is desirable to have all blocks on tape addressed sequentially, it is necessary to re-address the tape after these faulty regions have been erased.

In Atlas several operations can be controlled by time sharing the computer's activities (see Sumner, Haley and Chen, 1962; Kilburn, Edwards, Lanigan and Sumner, 1962; Edwards, Aspinall and Kilburn, 1964; Kilburn, Howarth, Payne and Sumner, 1961). One of these operations is the addressing or re-addressing of magnetic tape.

Summary of the Atlas tape system

The minimum requirements of a magnetic-tape system are that it be able to search for and transfer information. In the Atlas system these processes can take place on 8 tapes simultaneously. The high data rate involved (one word every 11 μ sec) under such a condition means that an automatic system of data transfer to the computer main store has to be provided. Control and monitoring activities are performed by programs. An 'interrupt technique' (Sumner, Haley and Chen, 1962) is used to inform the computer that action is required to control the tape system. When an interrupt signal is received the computer rapidly determines its identity, and takes the necessary action to deal with the contingency. This action is taken by programs called the *interrupt routines*.

The layout on tape (see Edwards *et al.*, 1964)

The dimensions of a block on tape are shown in Fig. 1. The reference marks define the block and inter-block lengths. The accurate detection of these marks is ensured by writing them as four reversals of magnetization which will produce four pulses on playback. These are integrated by an analogue circuit which produces an output pulse if any three are present. This allows for partial drop-out, and guards against spurious signals

* Department of Computer Science, The University, Manchester 13.

† I.C.T. Ltd., West Gorton, Manchester.

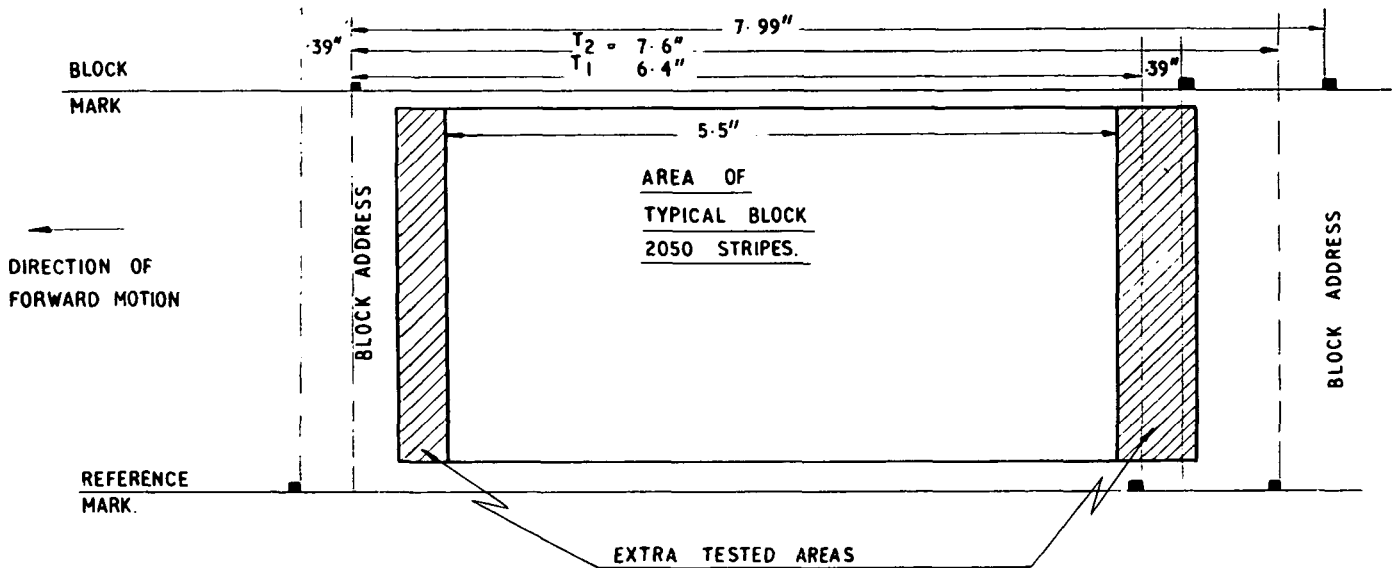


Fig. 1. Dimensions of block on tape

being mistaken for the mark. The system uses a double head, one for reading and one for writing, separated by 0.39 in. As the reference mark is read during Pass 1 it causes the block mark and address to be written to tape. Thus the distance along the tape between the reference and block marks is the read-write head separation of 0.39 inches. The block mark consists of 13 reversals of magnetization; the five corresponding to the clock pulses of the block address are preceded and followed by four which permit detection of the mark before the address is reached whilst the tape is travelling in either the forward or the reverse direction.

The distance between the leading and trailing block marks of 6.9 inches allows sufficient room for a block of data 2050 stripes long written under adverse speed tolerance conditions.

The start-stop characteristics of the tape deck specify the inter-block gap to be 0.8 inches. The total block length of 7.99 inches allows approximately 5400 blocks to be fitted onto a reel 3600 feet in length.

Programs are permitted to use up to 5000 blocks. If this figure is exceeded monitoring takes place. As much of the tape as possible is addressed, the blocks beyond 5000 being required for subsequent re-addressing. The layout of a reel of tape is shown in Fig. 2. A margin of tape 24 feet in length is left unmarked between the buckle and the title block, to allow for possible damage during handling. To permit easy identification of the title block, block 5000, and the last block on tape, they are given special address numbers 8189, 8190 and 8191 respectively. A margin of tape is left between the last block and the metal-backed leader to prevent this leader from reaching the sensing post in normal use.

The tape addressing process

The addressing or re-addressing of magnetic tapes is a fundamental part of the Atlas magnetic-tape system

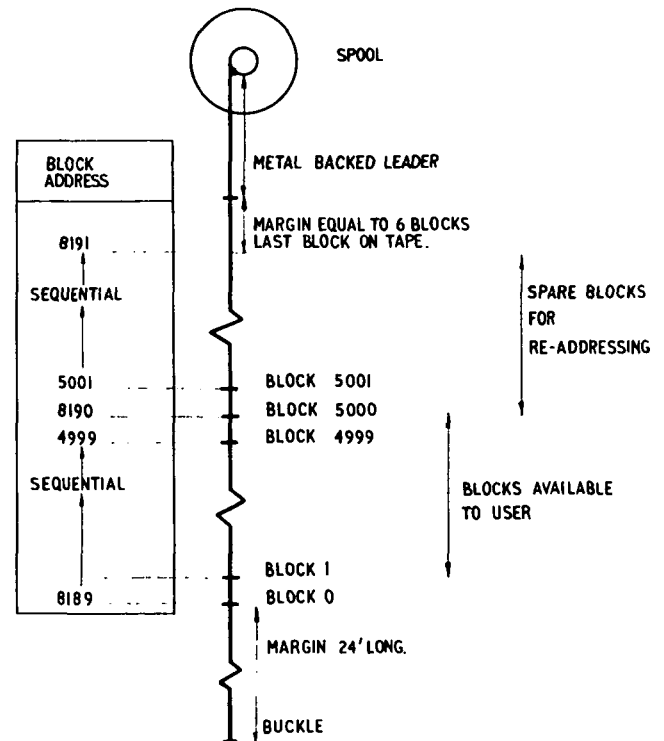


Fig. 2. Layout on 3600 ft of tape

and is not within the province of the object program. Thus the operation is not made available as an extra-code function. Instead, it is built into the supervisor itself, stored on magnetic tape and called into action, when required, by an input document on punched paper tape or cards. A file of usable tapes is kept, and it is the job of the operators to ensure that new tapes, or

those deemed faulty by the system, are addressed or re-addressed as appropriate.

The program is subject to three restraints.

1. The amount of effort required, on the part of the operator, should be a minimum.
2. The system should, as far as possible, be protected against operator's errors.
3. The demands on the central computer should be minimized in respect of
 - (a) computing time
 - (b) the amount of fast-access store reserved for the tape addressing and re-addressing routine.

It is essential that most of the interrupt routines are available, when required, in the fast-access store, and so these have been segregated from the other routines. Since the program is not in frequent use, the allocation of space in the fixed store for its interrupt routines is not justified. Hence they were written to occupy, along with their working space, one page of main core store, this being the minimum amount which can be locked out by means of the page-address registers. (For flow charts see Atlas Routine Specification, Vol. 1.) However, in the case of the end-of-tape interrupts, where the deck has stopped and is therefore not in urgent need of attention, the interrupt routine merely resets the interrupt and queues a request for action on extracode control. The extracode routine is not included with the locked-down interrupt routines since the time taken by the supervisor to locate it, if necessary, and to call it down from the magnetic drums is of little consequence. The "end of tape" interrupt and other routines amount to two further *pages* of store.

The process is started off by the operator who supplies the computer, via the appropriate peripheral, with the input document calling for the tape addressing and re-addressing program. When the channel is available the supervisor reads the complete addressing program of three pages from magnetic tape and stores them in the main core store from which, if their activity is low, they may be relegated to the magnetic drums by a supervisor routine known as the *drum learning program* (Kilburn *et al.*, 1962). The program announces its readiness to proceed by printing, on the operator's teleprinter, a command to engage the tape deck.

Before the deck can be used for addressing tape a switch must be operated to modify the deck. This allows power to be applied to the write amplifiers on the block and reference marker tracks, and also provides a control signal to the Tape Co-ordinator which switches one of the eight channels into the Tape Addressing mode. Whilst in this mode the control and counting circuits, which exist to facilitate conventional transfers, are altered slightly to control the addressing process.

When the operator engages the deck as instructed by teleprinter the event is detected by the supervisor, and arrangements are made to re-enter the addressing program. On re-entry checks are made that writing is permitted and that the "deck modified" switch has been

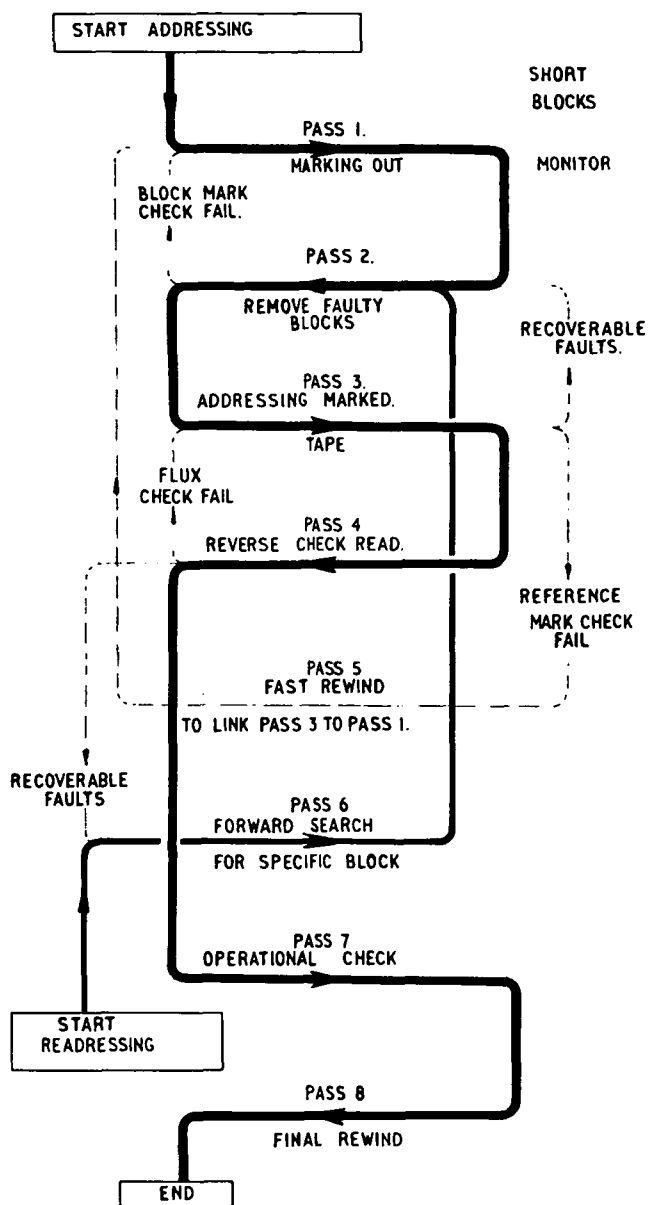


Fig. 3. Enumeration of the passes

appropriately set. If these checks are not satisfied, the deck is disengaged and the operator is instructed to take action. When the deck is set up correctly the program instructs the supervisor to lock down the interrupt routines in the main core store, and the deck is then started. At the conclusion of the addressing or re-addressing process, the interrupt routines are unlocked, the deck is released to the system, and the program is terminated.

Pass 1

The tape addressing process requires the tape to pass over the recording heads several times as summarized in Fig. 3. During the first pass the tape is marked out.

Lengths along the tape are measured by taking the tape speeds to be 120 i.p.s. $\pm 1\%$ and counting discrete time intervals.

The 24 ft. leader between the buckle and the first block on tape is measured by counting time intervals defined by a simple monostable timer which produces a simulated block-address interrupt approximately 40 milliseconds after being triggered by program. The resulting interrupt routine increases and checks the count, and either retriggers the timer or, at the end of the required time, commands the tape system to write the first reference mark to tape.

The distance between reference marks must be accurately defined in order to obtain the most efficient use of the tape. The necessary accuracy is achieved by using timing pulses from a crystal oscillator, which exists to control normal write-to-tape transfers. These pulses are counted in the normal way by the transfer control circuits, but their circuit range is extended for this purpose.

In due time the reference mark is read by the playback head which trails 0.39 inches behind the write head. Detection of this mark causes writing of the leading block mark and address. It also initiates the counting sequence to measure distances T_1 and T_2 (see Fig. 1) until the block's trailing reference mark and the next block's leading reference mark are written. Detection of the trailing reference mark causes writing of the trailing block mark, whilst detection of the next leading reference mark causes re-entry to the above sequence, which repeats itself automatically until failure to read a leading reference mark or the end of tape is reached.

The reading of a reference mark provides the sequencing signal. Loss of one reference mark can cause the sequence to get out of step. Though there is redundancy in the reference-mark signal, it is on an edge track and liable to total loss by physical damage. Since it is so important, a circuit has been provided to produce a signal simulating the reading of a reference mark. If the simulation has been necessary then a flip flop is set which is monitored to allow the associated block to be erased on a subsequent pass.

Whilst the reference marks control the sequence of writing marks to tape, the reading of the block mark provides a signal to the program to permit checking of the sequence and provision of control signals. A similar circuit is provided to simulate reading of a block mark and if simulation is necessary a flip flop is set for monitoring purposes.

The tape is tested for drop outs during Pass 1 by writing an all 1s pattern to each leading block address and to the area between reading a leading block mark and writing the trailing block mark. This area, although greater than that required for 2050 stripes, must be tested since blocks may be written subsequently under varying speed tolerance conditions and may be positioned at one or the other extreme of the area. The pattern is also checked during Pass 1 by using the

signals from the trailing read head. The amplitude of this signal is of the order of 10 to 20% greater than would be obtained after partial demagnetization of the tape, by subsequent passes over the heads, and so may mask partial drop-outs. It is therefore necessary to alter the bias voltage applied to the peak sense amplifiers, to set a more critical discrimination during the first pass than is applied during conventional read transfers.

Both the information and clock tracks are tested, and a single drop-out is sufficient to condemn the block, the fault being stored on a flip flop which is inspected by the block-address interrupt routines. These routines keep a running total of the block-address interrupts until a fault is detected. At this point the total is stored in a list and a new total is started. The totals are stored whenever the tape moves from a good region to a faulty region or vice versa. Thus the entries in the list constitute a measurement of the lengths of the good and faulty regions along the tape.

During Pass 2 a block is erased by turning on the write current at a block-address interrupt, and since, in reverse, the write head follows the read head, the absence of a trailing block mark renders the complete erasure of such a block impossible. To overcome this, the Pass 1 interrupt routines arrange to condemn the block following such a fault, along with the block itself.

At the end of Pass 1, a further check is possible. The tape is now stationary, on the leader, and so such checks are executed on extracode control. When the first block mark was written, the time was noted, and it was noted again on the last block-address interrupt. From the elapsed time an assessment is made of the maximum number of block marks permitted. If this value has been exceeded the program monitors, printing the message "short blocks" on the operator's teleprinter.

If the total number of block marks is satisfactory, the list is adjusted to condemn the last six blocks in order to create a blank leader at the high-numbered end of tape. Also, from the list, the number of blocks remaining after the faulty ones have been removed is calculated and stored. Finally the tape is started, in reverse, with the write reference mark flip-flop set.

Pass 2

The write reference mark flip-flop is set and reset as the list dictates, thus removing the reference marks from all the faulty regions of tape. When the tape reaches the metal backing, the extracode routine is entered. This checks that the total number of interrupts counted in Pass 2 is correct, and if not, the process is restarted at the beginning of Pass 1.

At this point in the process the tape is rewound, and the only data required for use in Pass 3 is the expected number of blocks on tape. This number is stored so that, in the event of a machine breakdown during Passes 3 or 4, a restart may be made at the start of Pass 3. The tape is then started in a forward direction.

Pass 3

In a similar manner to that in which the 24 ft leader is measured in Pass 1, the direction of magnetization is set in the correct state 12 ft from the metal backing (Appendix). The program operates as in Pass 1 with the exception that the write current is inhibited on the reference mark track, the existing reference marks are used to define the position of the blocks, and an isolated digit, which resets the direction of magnetization, is written to each of the clock tracks in the inter-block gap. Faults, where detected, are listed in a form identical to that of Pass 1 so that they may be removed by a re-run of Pass 2 with little administration being necessary. The blocks are sequentially addressed and, since the last block is now known, the three special blocks can be correctly addressed.

At the end of tape, the extracode routine checks that the correct number of interrupts occurred. If this check fails the indication is that, somewhere, a reference mark has appeared or disappeared. Hence reversion to Pass 1, via Pass 5, is necessary. Otherwise the routine examines the list. If errors were detected during Pass 3, an adjustment is made to the expected number of blocks and Pass 2 is re-entered.

If all checks succeed, the tape is started in the reverse direction.

Pass 4

Block addresses are checked in this pass and, where faults occur, the expected address of the offending block is stored in a list. This list is, of necessity, different in nature from that prepared in Passes 1 and 3 since the tape is now running in the reverse direction. The absence of an inter-block field reversal is noted but no immediate action is taken.

On reaching the metal backing, a check is made that the correct number of interrupts was obtained and that no inter-block field reversal was missing. If these checks fail, the program reverts to the start of Pass 3, since there is no evidence to indicate any disparity in the reference marks. Otherwise the program examines the list of faulty blocks. If this list is not empty, it is translated into the form acceptable to Pass 2, and Pass 2 is re-entered by way of Pass 6.

If, however, these checks succeed, the operator is instructed to unmodify the deck. When the operator has obeyed this request, the tape is tested, under user's conditions, by use of the normal magnetic-tape extracodes in Passes 7 and 8.

Re-addressing tape

The situation where it is required to remove specified blocks from an existing tape is identical to that obtaining at the end of Pass 4 where faults have been listed.

Therefore, the tape re-addressing routine achieves its objective by assembling the list of faulty blocks, given in the input document, in the form required by Pass 6, as for a Pass 4 failure, to search for the highest bad block,

and the remainder of the process is exactly as for addressing a magnetic tape.

Other checks

It is possible, though unlikely, that the number of faults detected in any one pass is large enough to preclude a complete listing. Because of the different techniques used to enumerate the faults, it is impossible to assign a fixed upper limit, but storage space exists for about 400 entries. In the event of this space being exceeded, the tape is rewound, the message "too many faults" is printed, and the program abolished.

On the successful completion of the addressing or re-addressing process, the program writes system administrative information (name, serial number, number of blocks, date and time addressed) to block 0. Before re-addressing a tape, the program checks the input document, and any non-standard data causes monitor action. If block 0 is not listed as faulty, its contents are checked against the input document, monitor action ensuing in the event of a check failure. Monitor action in the above cases consists of the message "error in data" followed by the abolition of the program.

Performance

After a preliminary test of the system, to remove program and circuitry errors, a magnetic tape was prepared for a more stringent test. By the use of acetone, patches of magnetic oxide were removed from a 500 ft length of tape. The tape was then manhandled to introduce creases and indentations, and respooled. Addressing was accomplished at the first attempt without the program finding it necessary to revert to an earlier pass.

During Passes 1 to 6 the channel used for addressing tape makes no demands on the main core store. Thus any slowing down of the central computer arises solely from the time taken up by the interrupt routines. The

Table 1
Times taken by interrupt routines

PASS NO.	ESTIMATED TIME PER BLOCK (MICROSECONDS)		
	MIN.	MAX.	NORMAL
1	192	272	236
2	96	144	96
3	192	276	256
4	156	192	164
5	Fast Rewind:— No Block Address Interrupts		
6	88	88	88
7	30	132	76
8	Fast Rewind:— No Block Address Interrupts		

amount of work done by these routines varies, in general, from leading to trailing block address, from pass to pass, and with the nature of the detected errors. The approximate mean times, per block, taken up by the interrupt routines are listed in **Table 1**.

Experience shows that, during tape addressing, the number of blocks condemned by Pass 1 is somewhat less than 1%. Furthermore, it is rare for the pass system to depart from the sequence 1, 2, 3, and 4. Hence, the average processing time per interrupt is approximately 94 microseconds, slowing down the central machine by about 0.3%.

Excluding any delays introduced by the operator, the total time required to address a tape of length 3,600 ft is about 36 minutes. One deck and one channel are occupied for the whole of this period, but the time taken up on the central computer is about 5.3 seconds.

Further refinements

Some additions have been made to the program whereby the detection of faults under user's conditions, in Pass 7, causes the computer to punch, on paper tape, the input document required for re-addressing. The decision to re-address is left to the operator, since such faults could result from machine malfunctioning.

It is also possible, in the input document, to specify an upper limit to the permitted number of blocks. By this means the leading portion of a tape may be addressed. This facility is used where a tape is required which contains substantially less than 5000 blocks. The time occupied on the tape addressing channel and the deck is reduced in direct proportion.

Conclusions

The tape addressing facility was the first part of the tape system to be commissioned, and has functioned satisfactorily since late in 1962. The linking of tape addressing into the supervisor was completed in

References

- EDWARDS, D. B. G., ASPINALL, D., and KILBURN, T. (1964). "Design Principles of the Magnetic Tape System for the Atlas Computer," *The Radio and Electronic Engineer*, Vol. 27, No. 1, p. 65.
- KILBURN, T., EDWARDS, D. B. G., LANIGAN, M. J., and SUMNER, F. H. (1962). "One Level Storage System," *Trans. I.E.E.E.*, EC-11, April 1962, p. 223.
- KILBURN, T., HOWARTH, D. J., PAYNE, R. B., and SUMNER, F. H. (1961). "The Manchester University Atlas Operating System, Part 1," *The Computer Journal*, Vol. 4, p. 222.
- SUMNER, F. H., HALEY, G., and CHEN, E. C. Y. (1962). "The Central Control Unit of the Atlas Computer," *Information Processing 62* (Proc. I.F.I.P. Congress 1962), p. 657.
- Atlas Supervisor and Fixed Store Routines*, Routine Specifications, Vol. 1, R450 and R451.

December 1963, and re-addressing in March 1964. It is more economical than an off-line addressing system since it is able to share a tape deck and much control electronics with the conventional system.

Its effect upon machine efficiency is difficult to estimate. The slowing down of the central computer by the interrupts is some 0.3%. However, since one page of core store, containing the tape-addressing routine, is locked down throughout tape addressing, the other programs in the machine use a core store of reduced size resulting in more drum transfers within the one-level store. The actual slowing down due to this is a function of the core store size, job type, and compiler.

A fair estimate of the slowing down of the jobs on the Manchester University Atlas with 16,000 words of core store is some 3%, a factor of ten worse than indicated by the slowing of the central computer. On-line time-shared tape addressing is economical only if carried out on a large computing machine.

The supervisor routine uses tape as a backing store. For this purpose it requires only the first few hundred blocks on a tape, and tends to wear this region severely. If a fault develops on the tape the offending region is cut off and the shortened remainder is addressed.

The re-addressing process in which the offending block is erased is of little value in this case since the heavily worn neighbours will soon show faults. This is an important but isolated use of the tape system, and is a task which would be better carried out by a disc store. The more usual tape jobs tend to wear a tape evenly over its entire length. Such tapes should benefit from the re-addressing facility, but there has been little experience of its use and it is difficult as yet to reach a valid conclusion as to its efficacy.

Acknowledgements

The Authors wish to acknowledge the support of all members of the Atlas 1 project both at Manchester University and I.C.T. Ltd., West Gorton.

Appendix

Direction of magnetization on tape

The method of writing a block to tape is to start the tape forward with the write amplifiers turned off. The address passes safely over the write head and is read by the trailing read head to be checked by the interrupt

routine, which then turns on write current. The write current erases previous data before stripes are written to tape. It is essential that turning on erase current on a clock track does not of itself produce a reversal of

magnetization which may be mistaken for a clock pulse on playback. This is guaranteed during Pass 3 by magnetizing the clock track, after the last clock reversal of the leading address, in the erase direction and ensuring that an even number of reversals takes place up to and including the last clock reversal of the next leading address.

The block address itself consists of five reversals of magnetization on each clock track. Thus when the address of the first block is written its last reversal will leave the tape magnetized contrary to erase unless an odd number of reversals precedes the first address. For this reason an isolated reversal of magnetization occurs before the first block address, causing there to be six reversals of magnetization between the write current

being first turned on at the beginning of Pass 3 whilst the tape is still on leader, and the region of tape between the leading address of the first block and the data.

It is also important to check for the presence of this isolated reversal, and facilities are provided for this during playback in Pass 4. Having provided a system for writing and checking an isolated reversal it is used in every interblock gap during Pass 3, and the write current is kept in the erase direction at all times except whilst writing the isolated reversal and the leading block address.

A final check of the direction of magnetization of every block is carried out in Pass 7 as data are written to each block in the conventional way.

Correspondence

An impossible program**

To the Editor,
The Computer Journal.
Sir,

I must admit that my previous acquaintance with *reductio ad absurdum* arguments has been confined to those which state the hypothesis which it is their intention to disprove; in the light of Mr. Lunnon and Mr. Outred's letter,* however, I now understand that Mr. Strachey intended his proof† to commence:

Suppose that there exists a Boolean function, which we will call $T[R]$, taking a routine (or program) R with no formal or free variables as its argument, such that, for all R , $T[R] = \text{True}$ if R terminates if run and $T[R] = \text{False}$ if R does not terminate.

I was obtuse in taking Mr. Strachey's proof to consist of definition and counter-example; however, I feel that my refutations‡ require but little amendment to meet the *reductio ad absurdum* argument, and I hope that you will allow me to rephrase them in that context.

In all three of these arguments I equate "program" with "program capable of being run". My understanding is that Mr. Strachey would concur. If, however, he does not—if he considers that a set of instructions not capable of being run still qualifies for the title "program"—then I will immediately agree that $T[R]$ does not exist (in the sense of "is not defined") for all programs R ; the proof is then trivial.

(i) I asserted that the function T exists by definition. Let us define it now.

Let R be a program (with no formal or free variables). Either R terminates if run or it does not. Define $T[R] = \text{True}$ if R terminates if run; $T[R] = \text{False}$ if R does not terminate if run. $T[R]$ is, then, defined for all R .

It should be noted that for certain R the value of $T[R]$ is unknown—I agree with Mr. Higman* in lamenting the fact that we do not know $T[A]$, where A is the example program he gives. For other R the value of $T[R]$ may be unknowable—in the light of Gödel's Theorem it should be possible to prove this. But the fact that the value of a function is in some cases unknown or unknowable does not affect the question of the existence of the function.

(ii) I asserted that P is not a program. My original

argument is certainly invalid in the light of the fact that Mr. Strachey's argument is a *reductio ad absurdum* one. May I replace it by another?

(iii) Consider the set of instructions P' :
Set of instructions P'
§L: if Y go to L
Return §

In working through this set of instructions we arrive at the question: what is the value of Y ? Is Y true or is it false? If the value of Y is known, then we can insert this value and, so, proceed with the execution of the set of instructions. But if the value of Y can not be determined until after the execution of P' has been proceeded with beyond this point (and clearly P' can not be proceeded with until after the value of Y has been determined) then we are at an impasse: P' can not be executed, and, so, is not a program. A particular case occurs when Y is $T[P']$; in this case P' is P . P is not, then, a program.

(iii) I asserted that Mr. Strachey had drawn too restricted a conclusion from his argument. In any *reductio ad absurdum* argument one has not just one hypothesis but several (or, if you like, one compound hypothesis). Arrival at a contradiction by a valid argument demonstrates that (at least) one of the original hypotheses is false. But it does not indicate which of the original hypotheses is false.

In the case of Mr. Strachey's argument a second hypothesis is (implicitly) introduced: that P is a program. The contradiction at which he arrives by valid argument demonstrates that one of the hypotheses is false. I maintain that his conclusion can be no more than "either T does not exist or P is not a program".

In conclusion, may I apologize to you and Mr. Strachey for the style of my previous letter. At the time of writing it I could not entirely dismiss the suspicion that Mr. Strachey had his tongue in his cheek, and I was attempting to guard against being taken in by a hoax; but that does not excuse its somewhat impertinent tone.

Yours faithfully,

27 August 1965

H. G. AP-SIMON

* This Journal, July 1965, p. 175.

† *Ibid.*, January 1965, p. 313.

‡ *Ibid.*, April 1965, p. 72.

** The first three of the letters that follow were received before the October issue went to press, but unfortunately they had to be held over for lack of space. Ed.