

ELLIOTT 900 SERIES SIMULATOR

ELLIOTT 900 SERIES COMPUTERS

Background.

"Elliotts" originally began as the "Elliott Brothers" a company of instrument makers. Over time the company moved into electronics and from thence to digital computing, by which time the company had changed its name to "Elliott Automation". In 1967 this business merged with English Electric. The mainframe data processing side of the combined companies merged into what eventually became International Computers Ltd (ICL), now part of Fujitsu. The process control and real-time systems side was taken over by GEC in 1968 and initially named "Marconi Elliott Computer Systems" then "GEC Computers" in 1971. In 1991 most of GEC except for the defence related businesses were renamed "Marconi plc", which collapsed in 2006. British Aerospace purchased the defence related parts of GEC, at which point the Elliott name vanished.

Elliotts produced several series of popular digital computers including the Elliott 803 and 503. The 900 series followed on from these and found widespread use in embedded digital control systems. Being much smaller than their predecessors they were suitable for deployment in mobile as well as fixed settings, and as the range developed, progress in miniaturization found them shrinking in size from the initial desk sized units to smaller boxes that could readily fit and operate in vehicles, ships and aircraft.

The first machine in the range was the 920A. It was quickly succeeded by the 920B and used for example in the Nimrod Mk I aircraft. Further developments were the smaller and faster 920M used in RAF Jaguar fighters, the 920C that featured an extended instruction set and was faster still and finally the 920ATC.

Alongside the military versions of the machines, Elliott Automation produced civilianized versions for applications such as process control, for running laboratory equipment in hospitals and elsewhere and also for teaching programming in schools and colleges. The civil versions were the 903 and 905, or when used in a process control setting, the ARCH 9000, and ARCH9050 respectively.

ELLIOTT 900 SERIES SIMULATOR

The best known of the civil machines due to its widespread use as a teaching system was the Elliott 903, based on the 920B, manufactured from about 1965 onwards. It had an 18-bit word ferrite core store with a 6 microsecond cycle time, paper tape I/O and a Teletype. Up to 64K words of store could be fitted in units of 8K. As an economy measure, both the sequence control (i.e., program counter) and index registers were held in the store rather than provided in the hardware. This led to a typical instruction time of 25 to 30 microseconds, but this was a bonus when using the machine for real-time work since at least these two registers did not need to be preserved between the four interrupt levels, as they were already in four separate pairs of storage locations. The 905 was a later faster machine, based on the 920C which could have up to 128K words of store, and held the sequence control register in inside the processor.

The 903 was constructed using transistors on plug-in packages. Peripherals could include a plotter, a line printer, a card reader, magnetic tapes, industrial interfaces, and displays for plant monitoring. There was no disk to act as a focus for an operating system, although an operating system based on magnetic tape was written for providing batch processing of student jobs and several for real-time applications.

Counting the military versions probably about 1000 machines were sold.

In summary the various 900 series models were, in historical order:

1. 920A / 901
2. 920B / 903 / ARCH9000
3. 920M
4. 920C / 905 / ARCH9050
5. 920ATC.

Languages available for the 900 series included ALGOL 60, BASIC, Coral, FORTRAN II and an assembler called "SIR" (Symbolic Input Routine) assembler. A version of FORTRAN IV was offered for the 905.

Library packages for ALGOL included ALMAT, a matrix arithmetic library, which originated on the earlier Elliott 803 and 503 computers, a plotting package based on routines developed for the Elliott 4100 series computers and ESP (Elliott Simulation Package) for running event-based models of physical systems. There was also a PERT package for project planning and various

ELLIOTT 900 SERIES SIMULATOR

utilities for program editing, tape copying and naturally test programs for diagnosing hardware faults.

The Elliott 900 series has been documented by Terry Froggatt on his personal web site: <http://www.tjfroggatt.plus.com/>
Some of the same information can also be found on the "Our Computer Heritage" web site:
<http://www.ourcomputerheritage.org>.

A more general history of Elliott Automation is in Simon Lavington's book: "Moving Targets - Elliott-Automation and the Dawn of the Computer Age in Britain, 1947 - 67", published by Springer, March 2011. ISBN 978-1-84882-932-9.

The author has collected and scanned many Elliott manuals. These can be found at:

http://homepage.ntlworld.com/andrew.herbert1/andrew_herbert/elliott.html.

ELLIOTT 900 SERIES SIMULATOR

ELLIOTT 900 SERIES COMPUTERS - HARDWARE.

This section summarizes the 900 series hardware from the perspective of a programmer and operator to give some context to understanding the simulator facilities. The information is mostly derived from Volume 1 of the Elliott 900 Technical Manual.

903 Operator's Control Panel

There were several kinds of control panel for fitting to different models of the 900 series and for different operating environments, some with more facilities than those described below, others with a subset, especially when the machine was part of an embedded system.

From the point of view of ordinary users the most important features of the operators console for a 903 series computer are

1. On and OFF switches for power
2. A RESET button to reset the machine to a standard state
3. A STOP button to halt execution
4. A RESTART button to resume execution after a STOP
5. A set of 18 word generator keys that can be used to set up a bit pattern to load into memory, or an address at which program execution is to begin
6. A JUMP button that cause program execution to start at the address set on the word generator
7. An ENTER button to load the word generator to the accumulator
8. An OBEY button to execute the instruction set up on the word generator keys.

These effects of various buttons are provided by simulator commands. There were several other buttons and switches to allow engineers to carry out diagnostic checks, but these are not relevant to the simulator.

ELLIOTT 900 SERIES SIMULATOR

900 order code summary.

The basic unit of addressing is the 18 bit word.

The most significant bit in an instruction word is the modifier bit, the next four are function bits and the lowest 13 are address bits.

18	17	14	13	1
/	F			N

Four registers are available to the programmer: A, Q, B and S. A is the accumulator and Q is the right half of the double length (A, Q) register. Q is sometimes called the "auxiliary register". S is the sequence control register, which in modern terminology would be called the "program counter". It is incremented after each instruction execution, except when a jump is taken, when the destination of the jump becomes the new content of the sequence control register. For the 920A, B and M models, this register actually a location in store determined by the current interrupt priority level. For the 920C model, S is held in a register and only written back to store and read from store upon a change in interrupt priority level. B is an index register. The register is held in a store location determined by the current interrupt priority level. If the modifier bit is set, the contents of B are added to the 13 address bits of the instruction prior to executing the instruction. (N.B., this does not affect the instruction in store).

In the tables following, m is the content of location M, with M being either N or N+ B if the instruction is modified.

/	B-modification	In the first stage of B-modification the Q register is affected in an undefined manner, consequently the effect of
		a modified 3 instruction is undefined.
0	Set B register and Q register	B := Q := m
1	Add to A	A := A + m
2	Negate and add	A := m - A
		Q := m

ELLIOTT 900 SERIES SIMULATOR

3	Store Q	m[17-1] := Q[18-2]; m[18] := 0
4	Load A	A := m
5	Store A	m := A
6	Collate with A	A := A & m
7	Jump if A = 0	S := m if A = 0
8	Jump	S := m
9	Jump if A < 0	S := m if A < 0
10	Count in store	m := m + 1
11	Store S	m := S; Q[17-14] := S[17-14]
12	Multiply	(A, Q) := A * m
13	Divide	A := (A, Q) / m rounded
14	n	Shift operations
	0 < n < 49	Shift (A, Q) left n places
	8164 < n < 8192	Shift (A, Q) right (8192 - n) places

Other 14 instructions specify block transfers to and from i/o devices identified by N with A containing a start address and Q a count of words to transfer (Q ≤ 4096). Details are given below in the sections on individual devices.

15	z	IO operations
	z = 2048	Shift A left 7 places; OR in character from paper tape
reader		to lowest 8 bits
	z = 2052	Same but for Teletype
	z = 6144	Punch character from lowest 8 bits of accumulator
	z = 6148	Same but for Teletype
	z = 7168	Program terminate to lower interrupt level. It starts in level 1 and can descend to level 4

Additional instructions supported by 920C/905/ARCH9050

15	z = 7169	Skip next instruction if A ≥ 0.5 or A < -0.5 or A = 0 (i.e., test for A standardized)
	z = 7170	B := B + 1; skip next instruction
	z = 7171	if B[13..1] = 0 A := the settings of word generator keys on the operator's control panel
	z = 7172	Q[18-2] := A[17-1]; Q[0] := 0
	z = 7173	A[17-1] := Q[18-2]; A[18] := 0
	z = 7174	A := B

ELLIOTT 900 SERIES SIMULATOR

```

z = 7175          B := A
z = 7176          Set relative addressing
z = 7177          Set absolute addressing

```

Other 15 instructions perform single word transfers to and from i/o devices with the device being specified by N. The data to be transferred is either taken from, or delivered to A as appropriate. Details are given below in the sections for each individual device.

In addition to the primary effects described above, several instructions had secondary effects, differing from machine type to machine type and these have been implemented in the simulator as appropriate. (The generic "900" simulation only includes the primary effects, matching the behaviour of the Ada simulators).

The following details of how "undefined" secondary effects worked is taken from a document "Programming Compatibility of 920 Series Computers", P.J. Lawrence, 1965, supplied by Terry Froggatt, and summarized below.

Function	920A	920B	920M	920C
B modify	Q := I*	Q[16-1] := N[16-1]	Affected	
	The number thus placed in Q will be over-written if the instruction itself has any effect on Q.			
6 collate	Q := A + n			
7 jump if zero	Q := I*	Q := N	Q := N	
8 jump	Q := I*			
9 jump if negative	Q := I*	Q := N		
11 store SCR	Q := S	Q := N	Q := N	
	Q[1] := 1 if A < 0 otherwise Q[1] := 0			
14 shift	N=0-4095 shift left N N=4096-8191 Shift right N	N=0-2047 shift left N N=6144-8191 shift right N	N=0-48 Shift left N N=49-64 Shift left (N-16) N=8128-8143 Shift right (N-16)	???

ELLIOTT 900 SERIES SIMULATOR

			N=8144-8191 Shift right N	
15 i/o	N=7169-8191 terminate	N=7169-7183 terminate N = 7184-8191 Computer stops	N=7169-8191 terminate	N=7168 terminate

Note (1): I* means all 18 bits of the instruction (920A only).

Note (2): N is interpreted modulo 8192 for 14 and 15 instructions.

The simulator estimates the execution time of programs based on the instruction timings listed on the "Our Computer Heritage" web site. The time for each instruction depends on the specific processor model and the speed of the memory, which differed between processor models, and some models came with different store speed options. The simulator also estimates elapsed time: this is the execution time plus an allowance for the time required to complete input-output operations, e.g., the time to advance paper tape through the reader, punch or teletype as required. These times could also vary depending on the specific units used. No allowance is made for the further time taken by the operator to wind tapes and label them etc. These factors significantly limited the productivity of 900 series machines used as general purpose computers with the machine standing idle for significant periods while the user dealt with paper tape and throughput limited by the relatively slow data rate of paper tape readers and punches.

Addressing Store.

920A

The 920A had either 4K or 8K of store. Thus the entire store could be addressed directly by the 13 bit instruction address (N).

ELLIOTT 900 SERIES SIMULATOR

920B/903/ARCH9000

These machines had a basic configuration of 8K of store that could be extended in units of 8K to a maximum of 64K. Each 8K unit was referred to as a module and numbered from 0 to 7.

Within each module, program and data are addressed relative to the first location (location 0) of that module. B-modification is used to access program and data outside the current module. The address actually addressed is $(S[16-14]+N+B)[16-1]$ therefore the address used for modification must be relative to the current module. Attempts to address store modules not fitted cause the processor to hold up. Because even the modified jumps used for subroutine exit are interpreted relative to the start of the module, Function 11, Store SCR, stores the module-relative address of the next instruction, namely just $S[13-1]$, into the specified store location. The remaining bits of S are placed in Q, (to facilitate return from "far calls" to subroutines in other modules).

920M

This machine was essentially the same as 920B, but with a maximum memory configuration of 32K.

920C/905/ARCH9050

These machines could have up to 8 modules of 8K store to a maximum of 64K, or up to 8 double modules of 16K store to a maximum of 128K. In addition, there were two addressing modes, absolute and relative: the N bits of an instruction are interpreted according to the function being obeyed and the state of the Address Mode Indicator (H) which is held as the top bit of the S register.

If $H = 0$ and/or on functions 7, 8 and 9 the N bits specify a location of the block in which the instruction itself is located; i.e., the absolute address of the location is formed by adding the N bits to bits 14-17 of the S register, the latter being those which specify the block in which the instruction is located. (Note the bottom 17 bits of the S register are used to allow addresses to extend to 128K, compared to just the bottom 16 bits being used in the earlier machines).

ELLIOTT 900 SERIES SIMULATOR

If H = 1 but not on functions 7, 8 or 9 the N bits specify absolutely a location in the first block of store (0 to 8191).

Thus, if H = 0, the 920C/905 behaves like a 920B/903/ARCH9000, where all addresses are interpreted relative to the start of the current 8192-word store unit. If H = 1, jumps are still interpreted relative to the current store unit, but data addresses are absolute.

For backward compatibility, programs entered by the JUMP button commence (on level 1) in Relative Address mode.

Instruction Times.

Instruction	Effect	Time in usec					
		920A	920B or 903	920M	920M	920C or 905	920C or 905
	for a Store Cycle time of	6-7½?	6	5	2	2	1
/	Add for B Modification	8	6.5	6	3.2	2.2*	1.2*
0 N	Set B register	33	28.5	22	10.8	6.6	3.6
1 N	Add	27	24	19	10.6	4.4	2.4
2 N	Negate and Add	33	27	21	12.6	5.3	3.3
3 N	Store Q	27	23.5	20	11.6	5.3	3.3
4 N	Read	27	24	19	10.6	4.4	2.4
5 N	Write	27	23.5	20	11.6	5.3	3.3
6 N	Collate	30	23.5	19	10.6	4.4	2.4
7 N	Jump if zero						
	A < 0	25	20	16	10.4	2.2*	1.2*
	A > 0	28	21.5	17	11.4	2.2*	1.2*
	A = 0	36	26.5	21	12.6	2.2	1.2
8 N	Jump	27	24	19	10.6	2.2	1.2
9 N	Jump if negative						
	A ≥ 0	33	20	15	9.4	2.2*	1.2*
	A < 0	33	26	19	10.6	2.2	1.2
10 N	Count in Store	30	24	20	11.6	5.6	3.6
11 N	Store SCR	38	31.6	25	17.8	5.3	3.3
12 N	Multiply	186	76.5	38	29.6	12.2	10.2
13 N	Divide	192	79.5	39	30.6	21.3	19.3
14 0-36 and 14 8156- 8191	Left Shift or Right Shift n places	28 +9n	22 +3n	16 +n	10.4 +n	3.9 +.9n	2.9 +.9n

ELLIOTT 900 SERIES SIMULATOR

14	2048-6143	Block Input or Output n words	n/a	$\geq 23.5 + 9.5n$	$\geq 18 + 10n$	$\geq 12.4 + 7.2n$	$\geq 4.8 + 6.5n$	$\geq 3.8 + 5.5n$
15	0-6144	Single Input or Output	≥ 28	≥ 20.5	≥ 20	≥ 14.4	≥ 6.5	≥ 5.5
	n/a	Program Interrupt	0	0	0	0	5.6?	3.6
15	7168	Program Terminate	28?	20.5	20	14.4	11.4	7.4
15	7169	Test standard no skip skip	n/a n/a	n/a n/a	n/a n/a	n/a n/a	4.9 5.8	2.9 3.8
15	7170	Count and test no skip skip	n/a n/a	n/a n/a	n/a n/a	n/a n/a	8.3 9.2	5.3 6.2
15	7171	Key Input	n/a	n/a	n/a	n/a	5.7	3.7
15	7172	A to Aux	n/a	n/a	n/a	n/a	4.7	3.7
15	7173	Aux to A	n/a	n/a	n/a	n/a	4.7	3.7
15	7174	A to B	n/a	n/a	n/a	n/a	6.1	4.1
15	7175	B to A	n/a	n/a	n/a	n/a	6.1	4.1
15	7176	Set relative	n/a	n/a	n/a	n/a	5.7	3.7
15	7177	Set absolute	n/a	n/a	n/a	n/a	5.7	3.7

* on the 920C/905/ARCH9050 modifying a conditional jump adds no time if the jump is not taken.

Interrupt Priority levels.

To enable interrupt driven operation, a 900 series processor can execute programs in any of four different priority levels.

Each priority level has its own sequence control register and index register. These registers are locations in store and can be referred to by programs in the normal way. (Note that the 920M and 920C/905/ARCH9050 hold the current sequence control register in a hardware register and only write it back to store when changing level, whereas in the earlier machines each access to the sequence control register requires a memory cycle.)

Priority Level	B. Reg. Location	S.C.R. Location
(Highest) 1	1	0
2	3	2
3	5	4
(Lowest) 4	7	6

ELLIOTT 900 SERIES SIMULATOR

The accumulator and the auxiliary register are shared between all four levels, so they must be safeguarded by program every time an interrupt occurs. It will also usually be necessary to reset the sequence control register on terminating a level so that the program, when next demanded, will start again at the same location.

All these conditions are fulfilled by the following control instructions. They are applicable to any program on levels 1, 2 or 3 that starts at location N.

Location	Instruction		Remarks
	Function	Address	
(N-6)	—	—	Store for lower level Q.
(N-5)	—	—	Store for lower level A.
(N-4)	0	N-6	Reset lower level Q.
(N-3)	14	1	Shift left 1 place.
(N-2)	4	N-5	Reset lower level A.
(N-1)	15	7168	Terminate, Note S reset to N.
Entry	(N)	5	Store lower level A.
	(N+1)	3	Store lower level Q.
	(N+2)		Required program (x locations).
	(N+2+x)	8	Jump to reset for lower level.

If the contents of Q on any of the lower level are not required then instructions N-4, N-3 and N+1 can be omitted and store location N-6 is not required.

Note that this program does not preserve bit 1 of Q, (the least significant bit of the AR). Any interrupts that occur during a function 0 instruction are deferred. This make it safe (in this control code or elsewhere) to load the most significant 17 bits of Q using function 0 and a one-place left shift, even though this briefly uses the least significant bit.

Programs entered by the JUMP button are obeyed from level 1. The initial program should set up locations 2, 4 and 6 to appropriate addresses, and then terminates to level 4, which is regarded as the base level.

ELLIOTT 900 SERIES SIMULATOR

Note that for the 920C/905/ARCH9050 there is a processor S register rather than S being a location in store. On a change in priority level, either by interrupt or 15 7168 instruction the content of S is written to the store location associated with the level being left and S is reloaded with the content of the store location associated with the target level.

Initial Instructions.

All 900 series machines load an "initial instructions" program into the top locations of the first 8K of store when powered on or when the operator pushed the RESET button to reset the machine or pushed the JUMP button to instruct the machine to commence execution at a location specified by the word generator hand keys on the operator's console.

	Address	Instruction	Effect
	8180	/15 8189	constant -3
ENTRY	8181	0 8180	set B to -3
	8182	4 8189	clear low end of A
	8183	15 2048	shift A and input paper tape character
	8184	9 8196	jump if bit 18 of A is set
	8185	8 8183	jump if A >= 0
	8186	15 2048	shift A and input a further character
	8187	5 8180	store word just read
	8189	4 1	read B into A
	8190	9 8182	jump if A < 0
	8191	8 8177	jump if A >= 0

(For the 920A the 15 2048 instructions are replaced by 15 4094 instructions which have the same effect.)

In simple terms this programs skips over blank tape and then reads 3 21 bit words which are stored in locations 8177, 8178 and 8179 respectively. The first non-blank character must therefore have bit 4 set and then be followed by 9 further characters containing the bootstrap instructions. If these instructions set B to -n and transfer control to 8181, n further words will be read in. Control is again transferred to 8177 that should therefore now either contain a dynamic

ELLIOTT 900 SERIES SIMULATOR

stop (8 8177) or a jump to the start of the program just read in.

Locations 8180-8191 as initial instructions are "enabled" whenever the JUMP button is pushed on the operator's console. For machines with more than 8K of store, the initial instructions are disabled after execution of a 15 7168 order. For machines with just 8K of store, the initial instructions are permanently enabled.

When initial instructions are disabled the locations can be used as normal store.

Peripherals.

The 900 series machines used paper tape as the primary means of input and output. The early machines had paper tape reader and punch only. Machines like the 903 were generally equipped with a ASR33 teletype in addition, allowing interactive input-output and therefore also command style interfaces to control executing programs. (Elliott documentation generally refers to them as 'teleprinters', rather than teletypes). There were additionally optional peripheral facilities such a graph plotter, line printer, card reader and magnetic tapes. Some magnetic tape machines therefore ran a magnetic taped-based batch operating system. Some 905 machines were equipped with disk drives and disk operating systems were produced for these. Generally these further peripherals were attached via a multiplexor that allowed up to 8 devices to be multiplexed over the simple i/o channel to the processor. (The paper tape devices connected via a separate paper tape station channel, and therefore on paper tape only machines there was no need to fit a multiplexor.

The F# simulator supports the basic paper tape and teletype facilities, the multiplexor, digital graph plotter, card reader, line printer and magnetic tape devices.

ELLIOTT 900 SERIES SIMULATOR

PAPER TAPE FACILITIES.

The minimum paper tape configuration was a reader and a punch. Readers were available in 250, 500 and 1,000 character per second versions. The paper tape punch operated at 100 characters per second. 5, 7 or 8 track tape could be handled.

Reading paper tape - Modes 1, 2 & 3.

Depending on the hardware fitted, paper tape could be read in one of three modes, as follows:

Mode 1: read tracks 8,7,6,4,3,2,1 as a seven bit code (i.e., track 5 is ignored)

Mode 2: read tracks 7,6,5,4,3,2,1 as a seven bit code (i.e., track 8 is ignored)

Mode 3: read all 8 tracks as an eight bit code.

Mode 1 is intended for use with 920 telecode and was the only mode provided for the 920A.

Paper tape stations that included a teletype generally only supported Mode 3, unless an in-line mode switch was fitted in the cable connecting the tape reader to the station.

SELECT switches.

Many 903s in particular were shipped in the '903C' configuration with an attached teleprinter. In this configuration a paper tape station control panel was fitted to the front of the system with an input selector switch and an output selector switch. Input could be set to 'READER', 'AUTO' or 'TELEPRINTER'. If READER was selected all paper tape input instructions (i.e., 15 2048, 15 2052) would input from the reader, with TELEPRINTER selected all input was taken from the teletype, with AUTO selected input came from the reader for 15 2048, and the teletype for 15 2052. The output

ELLIOTT 900 SERIES SIMULATOR

selector performed a similar function for paper tape output instructions (15 6144, 15 6148).

Generally the instructions to read or punch paper tape hold up the processor until the device was able to transfer data. However, machines could be fitted with an 'online adaptor' that allowed programs to poll paper tape station devices to determine when they were busy, so other work could be done rather than holding up on the device.

When an online adaptor was provided additional instructions were available. The adaptor added a 15 6145 instruction to set a control word, which had a bit for each device (bit 1: reader, bit 2: punch and if fitted, bit 3: teletype input, bit 4: teletype output). Setting the bit to 0 in the control word enabled online operation. Setting the bit to 1 obtained the "hold up" behaviour. In addition the adaptor added a 15 2049 instruction to read device status with the corresponding bits to the control word set to 1 if the device was ready to transfer data, 0 otherwise.

ELLIOTT 900 SERIES SIMULATOR

903 MULTIPLEXOR.

The 903 multiplexor allowed the single i/o channel to the processor to be shared by up to 8 devices. The presence of the multiplexor was transparent to both the processor and the connected devices.

The eight channels were divided into two sets called A and B. Instructions were provided to enable and disable each group as follows:

15 6016	Disable A and B
15 6017	Enable A and disable B
15 6018	Disable A and enable B
15 6019	Enable A and B.

ELLIOTT 900 SERIES SIMULATOR

4100 PERIPHERALS.

An Elliott 903 could be equipped with peripherals from the larger Elliott 4100 range of mainframe computers. In particular a "scientific line printer", a card reader and a teleprinter multiplexor. All these devices were connected through a special '4100 Interface matching Unit' (IMU) that converted between the 4100 computer interface and the 903 computer interface. If the 903 only had 4100 peripherals, the 4100 IMU would be connected directly to the processor, if there were other devices such as digital graph plotter or magnetic tape, the IMU would be attached via a multiplexor.

No details have been found relating to the teleprinter multiplexor. For the card reader and line printer, the details of these devices has been reverse engineering from the 903 subroutine library which contains the routine QLPOUT, for arranging output to the line printer and the routine QCARDIN for reading from the card reader.

4100 IMU.

The 4100 IMU can have up to eight channels in steps of two, each channel connecting a different 4100 device to the 903. Generally a channel 1 was used for a card reader and channel 2 for a line printer.

There are several different modes of data transfer provided:

SINGLE WORD TRANSFER (8 bits of data are transferred to or from l.s. bits of accumulator)

15	1536+n	Input data on channel (n+1)
15	1568+n	Input status word of channel (n+1)
15	5632+n	Output data to channel (n+1)
15	5664+n	Output control word to channel (n+1)

BLOCK TRANSFER

14	3648+n	Input packed data repetitive
14	3584+n	Input data unpacked repetitive
14	3680+n	Input status word packed repetitive
14	3616+n	Input status word unpacked repetitive
14	5696+n	Output data packed repetitive
14	5632+n	Output data unpacked repetitive
14	5728+n	Output control word packed repetitive

ELLIOTT 900 SERIES SIMULATOR

14 5664+n Output control unpacked repetitive.

Generally status words, as the name implies carry information about the device status, including busy indications and error conditions. Control words are used to instruct a device to take some action, in particular to initiate data transfers. The readiness of a device to receive further control words and/or transfer further data is signalled by interrupts.

Block transfer instructions apply to channel (n+1). As with other devices the address of the block to be used should be in A and the number of words in the block to transfer in Q.

Unpacked transfers write 8 bit characters one per word, packed transfers write words containing three 6 bit characters in the order most to least significant bit of each word.

The 15 6100 instruction (input status word) causes the 4100 IMU status to be read in to the accumulator. The significant bits of the word are as follows:

n+1=0	interrupt on channel n true
n+1=1	interrupt on channel n false
n+9=0	attention on channel n true
n+9=1	attention on channel n false
17=0	VALID line true during data transfer
17=1	VALID line false during data transfer (e.g., non-existent peripheral addressed)
18=0	COMPLETE signal not detected during data transfer, VALID line true for selected channel
18=1	COMPLETE signal detected before transfer of last character in block transfer, or VALD line false for selected channel.

Bits 17 and 18 are set to their appropriate values by each single or block transfer instruction. They will each be set to a 1 if a non-existent peripheral is addressed, and in the case of an input instruction any characters that are transferred are undefined. If VALID becomes false or COMPLETE becomes true, dummy transfers are performed and the status word indications set.

Bit 18 is also set to a 1 during a packed block transfer instruction, if the peripheral can only transfer a fixed number of characters, the total of which is not a multiple of three (e.g., 80 or 128). In an input instruction, characters transferred after the COMPLETE signal is detected are undefined.

ELLIOTT 900 SERIES SIMULATOR

The INTERRUPT and ATTENTION lines for all eight channels of the IMU are routed to the processor level 2 interrupt line. Interrupts signal readiness to transfer data, attentions signal error conditions.

Interrupts and attentions when signalled interrupt on level 2 and persist as indications in the IMU status word until cleared by an i/o operation to the interrupting device. Attentions persist until the associated condition is cleared.

When the processor is reset, bits 17 and 18 of the status word are reset to 0 and all operations immediately terminated, with a RESET pulse sent to all 4100 peripherals.

Card Reader.

The 4100 card reader operated at 300 * 80 column cards per minute.

Conventionally the reader is connected to channel 1 of the IMU.

The status of the reader is interrogated using the 15 1568 instruction (read status word). This loads a status word into the accumulator encoded as follows:

Bit 1	= 1 if reader is busy
Bit 2	= 1 if the reader is in "manual" (i.e., offline)
Bit 3	= 1 if interrupts are inhibited
Bit 4	= 1 missed transfer - i.e., the computer has failed to keep up with the reader
Bit 5	= 1 if a recoverable error has occurred (i.e., input hopper empty, receiver is full or open, card in transit not fully read)
Bit 6	= 1 if a non-recoverable error has occurred, (i.e., damaged card, card jam, machine fault).

Card reader interrupts are controlled using the 15 5664 instruction (set control word) with the control word in the accumulator, as follows:

Bit 1	= 1 to inhibit interrupts
Bit 2	= 1 to enable interrupts
Bit 7	= 1 to feed the next card

ELLIOTT 900 SERIES SIMULATOR

The reader interrupts on priority level 2 when the next column (or first column of the next card, as required), is ready to be input. Data is transferred from the reader using two 15 1536 instructions in succession, the first of which reads the first 6 bits of the 12 bit column code and the second reads the remaining 6 bits. (Note that the bits are read inverted: if the card column has the pattern 100100010000 reading from top to bottom, the pattern 000010001001 will be received by the computer. The reader interrupts as each successive column on the current card becomes available.

Line Printer.

The 4100 series scientific printer operated at 315 lines per minute with a maximum of 132 characters per line.

The status of the printer is interrogated using the 15 1569 instruction (read status word). The status is encoded as follows:

Bit 1	= 1 if printer is busy
Bit 2	= 1 if printer is in "manual" (i.e., offline)
Bit 3	= 1 if interrupts and attentions are inhibited
Bit 4	unused
Bit 5	= 1 if printer is in "paper low" state
Bit 6	= 1 if a non-recoverable error has occurred, (e.g., paper broken or finished, paper run away, hammer fuse blown, yoke is open, hammers are disabled).

The code &77777 signals that the line printer channel is invalid (e.g., printer not connected and/or switched off).

Printer interrupts are controlled using the 15 5665 instruction (set control word) with the control word in the accumulator, as follows:

Bit 1	= 1 to inhibit interrupts
Bit 2	= 1 to enable interrupts.

The printer interrupts on priority level 2 when it is ready for further output, either following an interrupt enable control signal or completion of a previously transferred line of output.

Data is transferred to the printer using a 14 5697 instruction (packed data repetitive). The accumulator should point to a

ELLIOTT 900 SERIES SIMULATOR

buffer containing packed SIR internal symbols, three symbols per word from most to least significant bit and the Q-register a count of the number of words to be transferred. The first six bit character of the buffer is interpreted as a count of the number of lines to throw before printing the current line. If the number of characters to be printed does not exactly fill the final word of the buffer, the unused positions should be filled with 0 (space).

When using the simulator, the line printer will appear to be in manual until the ATTACH LPR command is given. Line printer output is directed to a new window that pops up when the device is first used. The line printer can be taken back to manual using the DETACH LPR command.

Note the simulator reports an error if an attempt is made to initiate a transfer when the printer is busy - the effect on a real 903 + 4100 line printer is not described in any of the available documentation.

ELLIOTT 900 SERIES SIMULATOR

GRAPH PLOTTING.

A range of incremental graph plotters was available for the 900 series machines and software provided for producing graphs from SIR and ALGOL programs.

The plotter is controlled by the 15 4864 instruction which transfers a control code from the accumulator to the plotter. (There is also a 14 4864 instruction for block transfer to the plotter but this is not used by the Elliott standard software).

The significance of each bit in the control word is as follows:

1	step W
2	step E
4	step S
8	step N
16	pen up
32	pen down.

The plotter can move in any of 8 directions: N, NE E, SE, S, SW, W, NW, NE by code 10 for example. The effect of commanding opposing movements (e.g., code 3) is undefined, and the simulator reports an error if this is attempted.

ELLIOTT 900 SERIES SIMULATOR

MAGNETIC TAPE.

A 9k ch/s magnetic tape system was available for the 900 series.

The simulator models the 900 series hardware at the instruction level, but generally programmers accessed tape using utilities and libraries provided by Elliotts.

The tape system was connected through a microprogrammed controller. It operated upon ½ inch wide tape, using 7 tracks at a speed of 45 inches per second reading and writing, at a packing density of 200 characters per inch, giving a data rate of 9 Kc/s, or 3 kilo-words per second, i.e., a word every 333 usecs.

Up to four tape handlers could be connected to the controller. A handler could be in one of two states: 'manual', i.e., under operator control or 'remote' i.e., under computer control.

Tapes had physical markers near their beginning (approximately 4.3-4.9m in) (approximately 7.6-9.1m in) and their end that stops the hardware fully unwinding a tape from its spool in either direction. When first mounted a tape would automatically position at the 'load point' just past the start of tape marker leaving sufficient space (just over 3 inches) to enable backspacing over the first block written to the tape.

Instructions could be issued via the controller to any handler. Each handler has a fixed predetermined number in the range 0 to 3. Instructions issued to a handler in the manual state would be treated as instructions to 'do nothing'.

Data is transferred to the controller as 18 bit words that are reassembled into 6 bit characters by the controller. Data may be checked with odd or even parity and logic is included to detect long or short block errors. The system is self-clocking and therefore it is impermissible to write a character of seven zeros. (This can't arise if odd parity is selected; if using odd parity the user must avoid writing zeros). Note during a write operation a zero character will only be detected when the data is checked by the read circuitry about 8 msec later.) An apparent zero character may appear at any time if one or more bits in a character are dropped, or if there are imperfections in the tape. Whenever

ELLIOTT 900 SERIES SIMULATOR

a zero character is detected the tape is automatically stopped.

During backspacing the tape will only stop after a gap of more than 8 characters, making it possible to backspace correctly over blocks containing zero characters.

Data may only be written while the tape is moving in the forward direction, however it is possible to retreat over a block, i.e., backspace.

Data is normally written as a series of consecutive blocks with the start of the first block defined by the load point. A new block may be written if:

1. It is the first block defined by the marker
2. The preceding block has just been written
3. The preceding block has just been read
4. A block to be replaced has just been backspaced over.

If these rules are followed a block will be readable if it is either the first block or immediately follows a readable block other than the latest block.

Only one Read, Write, Erase or Backspace may occur in the system at a time. A Rewind however is offline to the controller so that any number of handlers may rewind simultaneously.

During reading and writing a gap of 1.5 characters or more is taken to indicate that a block has terminated. If a however a character is detected within 10 msec after detection of the longitudinal parity check character, then bit 10 'zero character' of the status word is set to indicate that the end of block might not be genuine.

There are two modes of operation - interrupts may be used or inhibited as required. The standard libraries inhibit interrupts, whereas some of the standard test programs enabled them. Data may be transferred in blocks (14 instruction) or words (15 instruction) or by a sequence of blocks and/or words.

The tape stops at the completion of each instruction.

Tape operations are controlled by three kinds of instruction: a Control Word instruction that commands the action to be taken, a Status Word instruction that interrogates the status

ELLIOTT 900 SERIES SIMULATOR

of the tape system and various data transfer instructions for reading and writing data.

The Status Word instruction interrogates an internal status word in the controller for the currently selected handler in which individual bits are set to indicate the presence or absence of different operating conditions, such as manual versus remote, parity error, various kinds of data transfer error and so forth.

Writing.

To write data on the selected handler the following sequence of instructions is used:

15 5121	-	Write Control Word (Prepare to write)
15 5120	-	Write a word, or
14 5120	-	Write a block
...	-	Further data writes
15 5122	-	Close Block

If the selected handler is in manual the sequence becomes a 'do nothing' - the sequence is executed as described below, but no data is transferred and bit 11 of the status word is set to 1, similarly if the tape mounted on the handler is not fitted with a write permit ring.

Prepare to write starts the tape moving. When at the nominal recording speed is reached (at least 9 msec) an interrupt is generated which must be answered within 250 usec by issuing the first data transfer instruction. This first transfer is completed and a further interrupt is generated 250 usec before the next data transfer is required. Similarly subsequent interrupts must be answered with 250 usec so that the computer supplies data fast enough to satisfy the data transfer rate to the tape. Characters are written to tape from most significant to least significant, word-by-word in order. After the last word has been transmitted, the corresponding interrupt must be answered with a Close Block instruction to indicate no further data is to be written. A final interrupt is given when the tape has stopped (at least 20msec later). A Status Word instruction may then be given, but there is no restriction on the response time to this final interrupt.

ELLIOTT 900 SERIES SIMULATOR

If interrupts are inhibited it is the programmer's responsibility to ensure instructions are issued at the correct rate, as if they were being given in response to interrupts. If an instruction is given before the controller is ready to transfer data, the computer will be held up until the controller is ready.

If a Status Word instruction is issued at any time during a write operation, other than following a Close Block instruction, then receipt of the status word will terminate the transfer in the same way as a Close Block instruction, but the status word will not be available until the tape has stopped.

If an interrupt is not answered in time by executing an appropriate instruction (Write a word, Write block or Close Block), bit 3 (Missed Data Transfer) will be set in the status word and the transfer terminated forthwith. Interrupts will still be generated at the normal rate until the Close Block instruction is issued by the data transferred will be discarded.

If any magnetic tape instruction other than Write a word, Write a block, Close Block or Status Word is issued while a write is in progress, the instruction will be held up indefinitely (and the simulator reports an error).

At the end of a block a check character is written after a gap of approximately 4 characters that contains a longitudinal even parity checksum.

If writing has taken the tape beyond the physical end of tape marker bit 9 'end of tape' will be set in the status word.

If the transfer has been treated as a 'do nothing' and the tape is stationary, the final Close Block instruction will issue its interrupt without any delay.

Writing a tapemark.

To write a tapemark on the selected handler the following sequence of instructions is used:

15 5121	-	Write Control Word (Prepare to write a tape mark)
---------	---	---

ELLIOTT 900 SERIES SIMULATOR

15 5120	-	Write a word
15 5122	-	Close Block

In all other respects this sequence behave similarly to writing a block, except that only bits 1-6 of the accumulator presented to the Write a word instruction is written to tape as a tape mark character.

Reading.

Reading follows a similar model to writing, using the following sequence:

15 5121	-	Write Control Word (Prepare to read)
15 1024	-	Read a word, or
14 1024	-	Read a block
...	-	Further data reads
15 5122	-	Close Block

The amount of data read off the tape is determined solely by the length of the block of data on the tape. If the block on tape contains more words than that expected by the computer, bit 6 ('long record') is set in the status word. If the block on tape contains fewer words than expected by the computer, zeros will be transferred at the normal rate until a Close Block is issued. Bit 5 ('short record') will be set in the status word. The time taken for the tape to accelerate to reading speed following the Prepare instruction is at least 14 msec and the time taken to stop the tape after a Close Block following a read sequence is at least 11.8 msec.

If the number of characters read off the tape is not an integral number of words, the least significant bits of the last word transmitted to the computer are filled with zero characters, but the short record bit will not be set.

Note the long record bit will also be set if a transfer is terminated by a Status Word instruction.

Bit 4 of the status word is set if a parity error is detected.

If an attempt is made to read from blank tape, the effect is undefined. (The Simulator produces a 'zero character' status indication).

ELLIOTT 900 SERIES SIMULATOR

A tapemark can be read as if it were a single word block. Note that on reading the tapemark character will be in the most significant 6 bits of the accumulator, whereas it was written from the least significant.

Control Word.

15 5121 - Output control word from accumulator to the controller.

Only bits 1-10 are significant as follows:

BITS	EFFECT
1, 2	Handler number (0-3). The remaining bits only apply to the selected handler.
3, 4	Unused.
5	=1 interrupts permitted, =0 interrupts inhibited.
6	=1 select odd parity.
7	=1 select even parity.
6,7	=0,0 leave parity unchanged from previous setting.
6,7	=1,1 undefined (the simulator reports an error).
8,9,10	=0 do nothing
	=1 Erase
	=2 Backspace
	=3 Rewind
	=4 Prepare to write a tapemark
	=5 Prepare to read
	=6 Prepare to write
	=7 Rewind in manual.

Erase writes zeros to approximately 4 inches of tape destroying any data previously present. If the handler is in manual, or the tape is not fitted with a write permit ring the instruction is treated as 'do nothing'. If interrupts are permitted, an interrupt will be generated when the operation is complete and the tape has stopped. If a Status Word instruction is issued prior to the interrupt occurring the computer will be held up until the controller is ready. If the erase passed the end of tape marker, bit 9 will be set in the status word.

Backspace retreats one block on the selected handler, leaving the tape positioned to enable block just traversed to be read or over-written. If the handler is in manual the operation is treated as a 'do nothing'. If the beginning of tape marker is encountered, the operation is stopped and bit 11 set in the status word. If the tape is at the load point, the

ELLIOTT 900 SERIES SIMULATOR

instruction will be treated as a 'do nothing' and bit 11 will be set in the status word. The controller will be busy until the tape comes to rest at the end of the operation, at which time an interrupt is generated. If a Status Word instruction is issued prior to the interrupt occurring the computer will be held up until the controller is ready.

Rewind and Rewind in manual both rewind the tape on the selected handler. Rewind leaves the handler online whereas Rewind in manual sets it offline as soon as the rewind is initiated.

Both instructions move the tape at high speed (180 inches/sec approximately) in the reverse direction until the load point is detected. The tape then stops and re-positions to come to rest at the load point.

If a Rewind instruction is issued when the tape is at the load point it is treated as 'do nothing' instruction. The handler will remain busy from the time the Rewind instruction is received until the tape has come to rest and is ready for further operations. Bit 1 'handler busy' of the status word will be set during this interval. No interrupt is generated by a rewind instruction. A Status Word instruction may be issued and answered at any time while a rewind operation is in progress.

A Prepare to write instruction is used to start the tape moving forwards. It is impermissible in isolation and must be followed by a Write word or Write block instruction in response to the interrupt generated when the tape is running at nominal recording speed. Prepare to write makes the controller busy until the following write operation is complete. If any other instruction is issued other than those specified above. The computer will be held up indefinitely (the simulator reports an error). Refer back to the section above on writing for the error conditions associated with this command.

A Prepare to read instruction is used to start the tape moving forwards. It is impermissible in isolation and must be followed by a Read word or Read block instruction in response to the interrupt generated when the tape is running at nominal recording speed. Prepare to read makes the controller busy until the following read operation is complete. If any other instruction is issued other than those specified above. The computer will be held up indefinitely (the simulator reports

ELLIOTT 900 SERIES SIMULATOR

an error). Refer back to the section above on reading for the error conditions associated with this command.

A Prepare to write a tapemark instruction is used to start the tape moving forwards. It is impermissible in isolation and must be followed by a Write word instruction in response to the interrupt generated when the tape is running at nominal recording speed. Prepare to write makes the controller busy until the following write operation is complete. If any other instruction is issued other than those specified above. The computer will be held up indefinitely (the simulator reports an error).

Status Word.

15 1025 Input status word of currently selected handler to the accumulator.

The significance of the bits in the status word are as follows:

BITS	SET BY	CLEARED BY
1 Handler busy	Accepting a Rewind or or Rewind in manual Instruction.	Completing the rewind.
2 Handler in manual	Handler in manual state, or unavailable.	Handler in remote State.
3 Missed data transfer	Any interrupt other than a final interrupt not answered within 250 usec	Accepting a Control Word instruction.
4 Parity error	Parity check failure during reading or writing.	Accepting a Control Word instruction.
5 Short record	Reading a short block.	Accepting a Control Word Instruction.
6 Long record	Reading a long block.	Accepting a Control Word instruction.

ELLIOTT 900 SERIES SIMULATOR

7 Write permit	Write permit ring fitted.	Write permit ring removed.
8 Load point	Detecting beginning of tape marker.	Not detecting marker.
9 E.O.T.	Detecting end of tape tape marker while moving forward.	Accepting a Control Word instruction.
10 Zero character	Detecting false end of block.	Accepting a Control Word instruction.
11 Done nothing	An instruction has been treated as a 'do nothing'.	Accepting a Control Word instruction.

If bit 2 is set to 1, bits 1,7,8 and 9 have no significance.
Bits 12-18 are undefined.

All status word bits are reset by a Control Word that selects
a different handler, or pressing the Reset switch on the
controller.

Interrupts and timing.

The magnetic tape controller interrupts on level 2. The
inhibiting of interrupts merely prevents the interrupts being
transmitted to the computer and in no way affects the logic
within the controller.

If interrupts are inhibited, the missed data transfer status
is set to 1 if a Write instruction is not followed by a
further Write or Close Block instruction within 250 usecs.
Similarly for Read instructions. The first Write instruction
must be made available within 9 msec of issuing the Prepare
instruction and the first Read instruction within 14 msec.

ELLIOTT 900 SERIES SIMULATOR

The Elliott documentation gives some timings for magnetic tape operations and the specification of test program XMT71 gives further information, although the two are neither complete nor consistent. The simulator adopts the following timings:

Time to read/write a word	333 usecs
Prepare to Read (do nothing)	9000 usecs
Prepare to Read	9333 usecs
Prepare to Write (do nothing)	14000 usecs
Prepare to Write	14333 usecs
Read/Write (per word)	333 usecs
Close Block (after reading)	11800 usecs
Close Block (after writing) +	20000 usecs
Erase	111000 usecs
Back Space ++	20800 usecs

- + 25 ms if final write was a Write Word operation (required to make XMT71 succeed).
- ++ plus time taken to read characters of previous block, if any.