The Development of a Satellite On-Board Microcomputer*

Deryck W. Eardley
British Aircraft Corporation
Bristol, England.

Summary The paper describes a miniaturized computer for space use that is being developed under a contract from Intelsat. An all CMOS design is used with the corresponding low power and high speed associated with CMOS. Extensive use is made of LSI and the computer is based on the Intersil or Harris 6100 microprocessor. This has the same architecture as the PDP8-E and the extensive range of DEC PDP8-E software can be used. Provision has been made for both serial and parallel input/output ports. A development prototype is described that has serial and parallel ports. The associated memory is expansible up to 32K words in 2K blocks. RAM or PROM can be used or interchanged in blocks of 256 words. A programmable timer is included for real time applications.

A bench model of the computer is also described which is made for program development. It is not to the same environmental standards as the development prototype but has the same facilities together with a control panel and provision for a teletype. The use of the Bench Model for software development is described.

Introduction BAC are in the process of developing a microcomputer under a Intelsat contract. This microcomputer is intended for Attitude and Orbit Control on board a satellite. However it is a general purpose machine that has many applications.

Two models are being produced a Bench Model and an engineering (‘E’) model. The Bench Model will be used for testing and for program development and will incorporate a control panel and interface for a teletype. The ‘E’ model is more like a flight model and requires the same considerations from a programming and operating point of view.

The design philosophy is to produce an all CMOS design and to make as much use of large scale integration (LSI) as possible. The all CMOS design gives low power consumption and high speed. The use of LSI produces a physically small unit with the same capabilities as larger units and increases reliability due to the use of less components. The microprocessor used is the Intersil IM6100 or Harris HM6100 together with their associated LSI control elements. This has the same architecture as the PDP8/E and will

This work was supported by Intelsat under Contract No. TS 837.

take the extensive range of Digital Equipment Corp. (DEC) software. The microcomputer is designed for 10V operation but can be run off a lower voltage for lower power operation; this however, reduces operating speed.

Serial and parallel input/output (I/O) ports together with a direct memory access (DMA) channel are provided. The serial port uses only four lines giving a clock, busy signal, priority chain and a data line. Both the DMA and the parallel port have 12 data lines, to give and receive data as 12 parallel bits. They have two control lines which may not always be used in the case of the parallel port. The parallel port has separate input and output data lines each with two separate sets of control lines (one flag and one sense line each). The DMA data lines are bidirectional requiring one set of data lines only together with data ready and data received lines for control purposes.

'E' Model The 'E' model of the microcomputer operates from a 35 volt supply of ± 30% total variation and over a temperature range of -20°C to +100°C. The facilities provided and method of control are the same as that of a flight model. It is made up from the following printed circuit boards:-

- Microprocessor board - contains the 6100 microprocessor, DMA port, programmable timer, watch dog timer, power-on reset and memory control logic.
- 2K memory board - PROM or RAM can be accommodated or mixed in blocks of 256 words. Up to 16 such boards can be used together providing a maximum of 32K words of memory.
- Serial I/O port board - up to 26 such ports can be used in the unit.
- Parallel I/O port board - 2 ports are contained on each board up to 52 ports can be used in the unit.

A block diagram of the system is given below:-

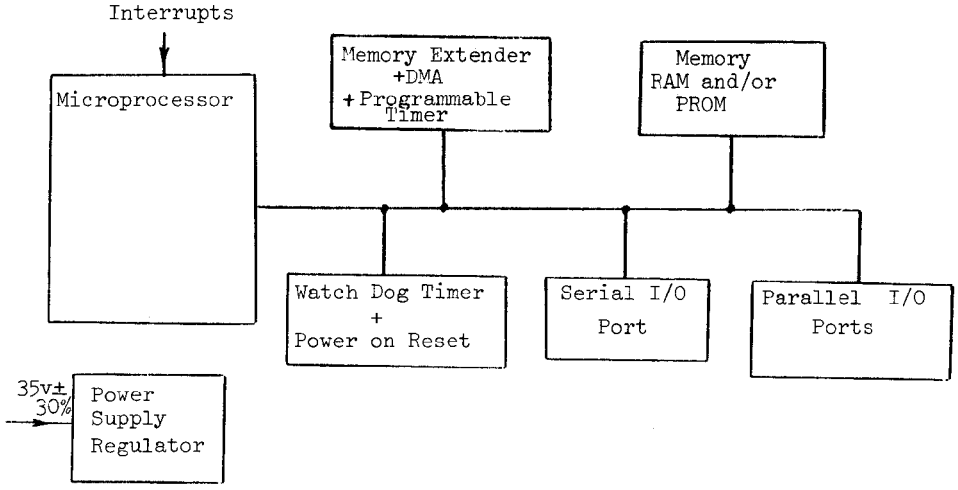

Fig. 1. Block diagram of system.

The limits on speed of operation of the I/O ports as regards processor timing and software considerations are 160K words/sec. (2 Mbits/sec) for the Serial and Parallel ports and 660K words/sec. (8 Mbits/sec) for the DMA port. In practice the Interface drivers and lines together with power requirements will limit the speed of operation. Increasing the speed of operation, the line length or the number of devices on the line increases the power requirements of the interface circuits. Hence a compromise must be made between power and interfacing requirements. If a limit of 25K bits/line is assumed, the speed of operation of the Serial port is then 1,562.5 words/sec. (25K bits/sec.) and for the Parallel and DMA ports is 25K words/sec. (300K bits/sec). Open drain drivers are used in the I/O interface with pull up resistors at the receiving end.

The system time for a 12 bit addition is 3 μS. This is limited by the processor timing and delays in the memory driving circuits.

Processor Board A number of functions are contained on this board which are shown in fig. 2.

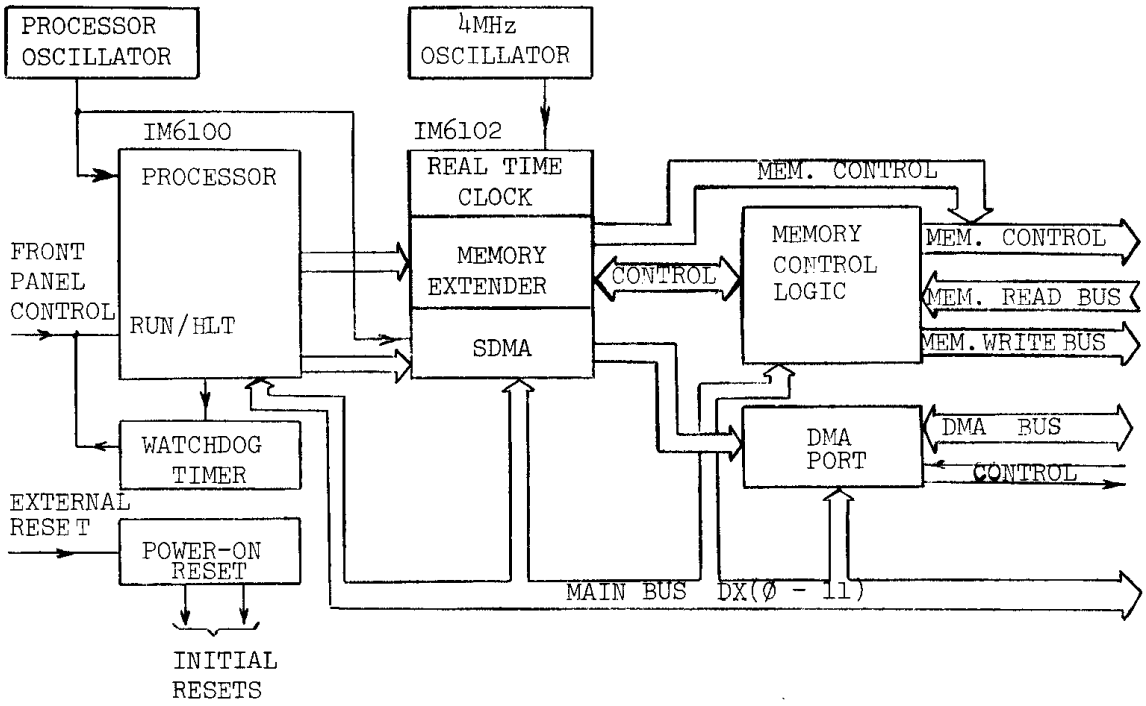

Fig. 2. - Processor Board Block Diagram

The Watchdog timer counts the number of Instruction Fetch operations from the Processor. The total count length can be adjusted in 5 stages between 4096 and 65536 counts, by using wire links on the board. The count is asynchronously set to zero whenever a write operation is received from a particular Parallel Interface Element (PIE). If the counter reaches the maximum value set, before being cleared by a write, the Processor

will be halted at the end of its current instruction. This indicates a possible error within the Processor or programme i.e. that the programme has stuck in a loop.

The “writes” which are used to reset the count are dependent on the type of programme being executed. They are not generated on a fixed interval basis.

The SDMA (Simultaneous DMA) channel is used in conjunction with the DMA port and control logic on the same board. The SDMA channel augments the throughput of the Processor during DMA operations by transferring data between memory and peripheral devices simultaneously with normal processor bus usage. This means no memory cycles are “stolen” from the Processor, but the DMA address and data are transferred onto the bus during periods that the Main Bus is inactive.

The real time clock (RTC) is a programmable function which is used by the Processor to accurately measure and count intervals or events in order to implement real time data acquisition and data processing. Timing is provided by an independent 4MHz crystal oscillator. An internal Clock Enable Register controls the mode of counting and the rate of the time base of the clock.

A memory extension controller extends the addressing capability of the Processor from 4096 words to 32768 via a 3 bit extended address which is decoded to select 1 of 8 memory fields, each of which provide 4096 words of memory.

The power on reset is used to reset the total system when it is first powered. Two signals are derived, INITIAL RESET and INITIAL RESET, and taken to other boards. These signals will also be activated by the RESET push-button on the control panel or by an external reset.

Memory The memory boards will contain 2K words of 12 bits which can either be RAM or PROM or a mixture of RAM and PROM on the same board. The memory capacity can be readily increased in blocks of 2K up to 32K. A separate memory control common to all 2K memory boards up to 32K will be provided on the Processor board. The input and output bus lines on the memory are buffered by the memory control logic so that there is no increase in loading of the processor main bus when extending the capacity of the memory up to the maximum of 32K.

Serial Input/Output Port This is used to interface a number of external users with the processor. The interconnection of devices using a priority system is shown in Fig. 3. A device with a high priority will inhibit devices further down the chain from outputting data. The common Busy line is used to indicate when a device is actively transferring data. It is used by the external devices only and is not directly sampled by the processor. Data is

transferred in 8 bit bytes each byte is preceeded by 4 bits of sync. and 4 bits of address. Transfer time for 8 bit byte 640µS. The port employs a single line for data and address, plus priority control lines to ensure that only one external device may use the port at a time.

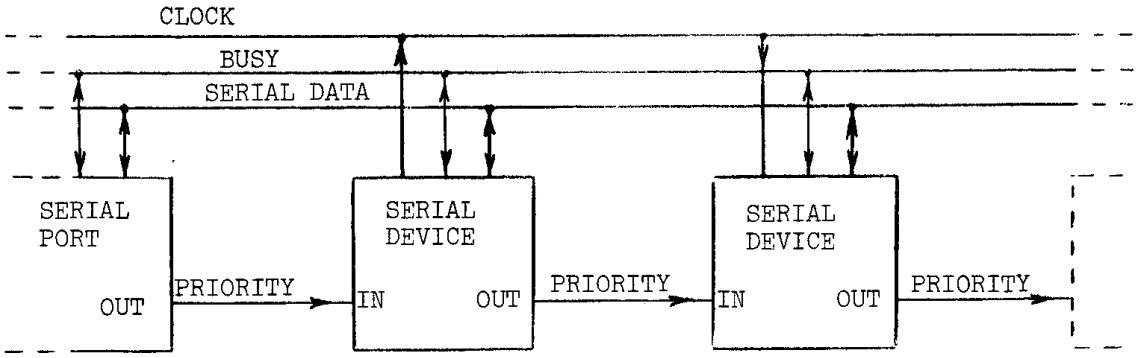

Fig-3. Interconnection of Serial Devices in a Priority System

An output is performed by setting a control bistable. If the Busy line is not active, indicating that no other device is trying to write to the processor, the priority logic will put the Busy line into its active state. The state of the control bistable can be read by the Processor to indicate if an output transfer is being made, or is ready to be made. The priority logic enables the Data, preceded by the sync and address of the device, to be clocked out onto the Data Line and inhibits data from being received by the Port. After 16 clock periods the control bistable is reset, thus inhibiting further data output and releasing the Busy line.

A serial input can only occur if data is not being output. When a particular sync combination is recognised, a bistable is reset which prevents the recognition of another correct sync combination for 13 clock periods. After this time the Port will again search for a correct sync signal. The enabled sync is used to given an interrupt and also latch the data plus address bits. The Processor can determine from which device the data has come by recognising the address bits.

There are two modes of operation under which an interrupt is given to the Processor. When the Processor is the master it will give out a Flag which allows an interrupt to occur every time that the port recognises a correct sync, The second mode can be used to receive data only from a particular serial device designated by the address set up on the port itself.

The 25KHz clock required for the serial port and external serial devices is generated on the serial port board and is derived from the 4MHz crystal oscillator.

Parallel I/O Ports Up to 52 parallel ports can be used with the processor for transferring data between external devices and the processor.

Each port consists of twelve data lines, one flag output and one sense input. (The sense input can be changed to an interrupt input by software whenever required). The flag and sense lines are supplied for control purposes, but may not always be required. The fact that the twelve data lines are dedicated unidirectional lines means that outputs can be changed and inputs read either when required or at fixed time intervals. The control lines will be useful for either high speed data or data that does not change very rapidly. i.e. is slow compared with the executive cycle time. With the latter it may save time if the data is read only when it changes instead of at fixed time intervals. This is possible if the interrupt facility is used.

With high speed data the control lines can be used on a hand shake basis to indicate when new data is ready and when it has been read. Data would be output followed by the flag. The flag is sensed by the sense input of an input port and the data is then read. The input port then outputs a flag which is sensed by the sense input of the output port enabling it to output fresh data. The sequence is then repeated.

Unit-Packaging To exploit the advantages of the flexible design as regards memory and input/output port expansibility a modular packaging scheme has been adapted for the construction of the Engineering model. This allows any number of modules to be assembled to form a complete unit capable of satisfying a wide range of mission requirements. Any number of modules can be stacked together. Each module contains two printed circuit boards. Printed circuit board interconnections being provided by a stacking connector which removes the need to provide intermodular wiring.

The stacking connector is a one part connector, consisting of a simple moulding containing contacts which carry connections through from the top to the bottom of the moulding. It is sandwiched between the circuit boards and carries the connections from the upper surface of each board to the lower surface of the adjacent board throughout the stacked assembly.

Through bolts are used to assemble the stacked modules into a single unit. The bolts pass through clearance holes in each module frame, with pilots being used in each hole for the accurate alignment of the bolts and the modules.

Power supply modules and electrical distribution to other units would be provided by mounting additional modules at the ends of these stacked unit. The input/output ports modules have their own input/output connectors mounted as part of module. External connectors would be the Cannon Royal ‘D’ type.

The components are mainly of the dual-in-line package type. Eight layer multilaminate boards are used for the processor and memory. The ports being double sided boards.

The dimensions of the modules in 242 x 154 x 19-7mm. Thus a typical processor having say 16K of memory and 2 input/output ports would be 118.2mm high.

A cut away diagram of the E-model processor is shown in Fig.4.

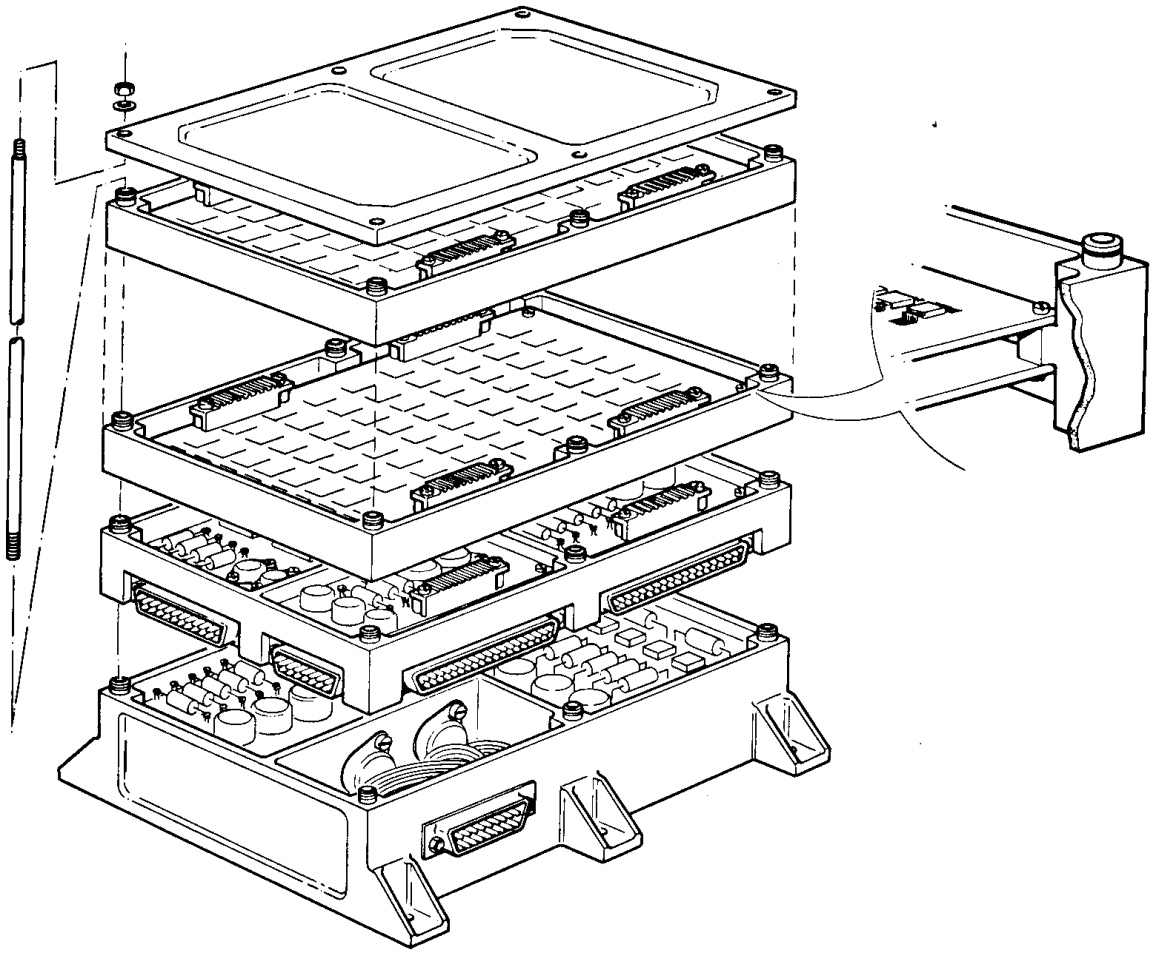

Fig.4. ‘E’ Model Construction

Bench Model and Peripheral Test Set The Bench Model and a Test Set are used for testing the ‘E’ Model and for software development. The Bench Model is a modified proprietary processor development system containing the 6100 processor and memory. The Test Set is a custom designed unit which allows full manual control of the processor and the display of all necessary data. Both units are housed in a standard 19" rack and interface with the E Model via cable looms.

The Bench Model provides the same facilities as the ‘E’ Model plus a control panel and teletype interface. Programs can be loaded into the Bench Model via the teletype and

checked for correct operation with the aid of the control panel. The final program can then be punched on paper tape via the teletype. The Bench Model control panel allows data to be loaded direct into main memory; the program counter to be changed and the extended address to be set. Controls are also available for single stepping through the program and inhibiting write to parts of the main memory so that it effectively changes from RAM to PROM. A loading program is also contained in a memory dedicated to the control panel which allows binary tape to be read from the teletype.

The Test Set allows the serial and parallel I/O ports to be checked manually.

Testing is performed by using the test set as a master and sending data to the processor. The returned data is then displayed and checked. Controls for the 'E' Model are provided which allow the 'E' Model to be stopped, started, reset and the Watchdog Timer to be enabled or disenabled. Interrupts to the 'E' Model can be given manually or from a signal generator. By setting the appropriate switches on the Test Set the 'E' Model is made, on being reset, to set up the DMA port for program or data transfer to/from the Bench model.

A PROM programmer is driven from the Bench model and used to blow PROMs for the 'E' Model.

Software A large library of fully debugged and proven programs (saftware) is available from the Digital Equipment Corporation (DEC). These programs were developed for use on their PDP 8 range of minicomputers and are available for use on the 6100 processor systems. Intersil also sell test programs to check the control panel memory, main memory and processor. BAC have written programs to test the I/O ports that will be supplied with the Bench Model and 'E' Model. Some of the software available is described below.

Extended Software Package QF081-CB This is a set of programs and literature that assists the user to create, edit, debug and correct his program. A flow chart of the program writing procedure is shown below.

First a flow chart is prepared for the program. This is very important when writing assembler programs since they are machine specific and a flow chart enables another assembler program to be written for a more advanced machine at a later date. The program is then to be coded using the symbolic IM6100 instruction set.

A symbolic program tape is then punched. This is performed by loading one of the programs, called Symbolic Editor, supplied as part of this package. This is loaded using the Bin Bootstrap program in the control panel memory. The Symbolic Editor enables a program to be typed in from the Teletype keyboard and mistakes to be corrected, verified and recorrected, if necessary. It includes a search feature to scan the text for occurrences

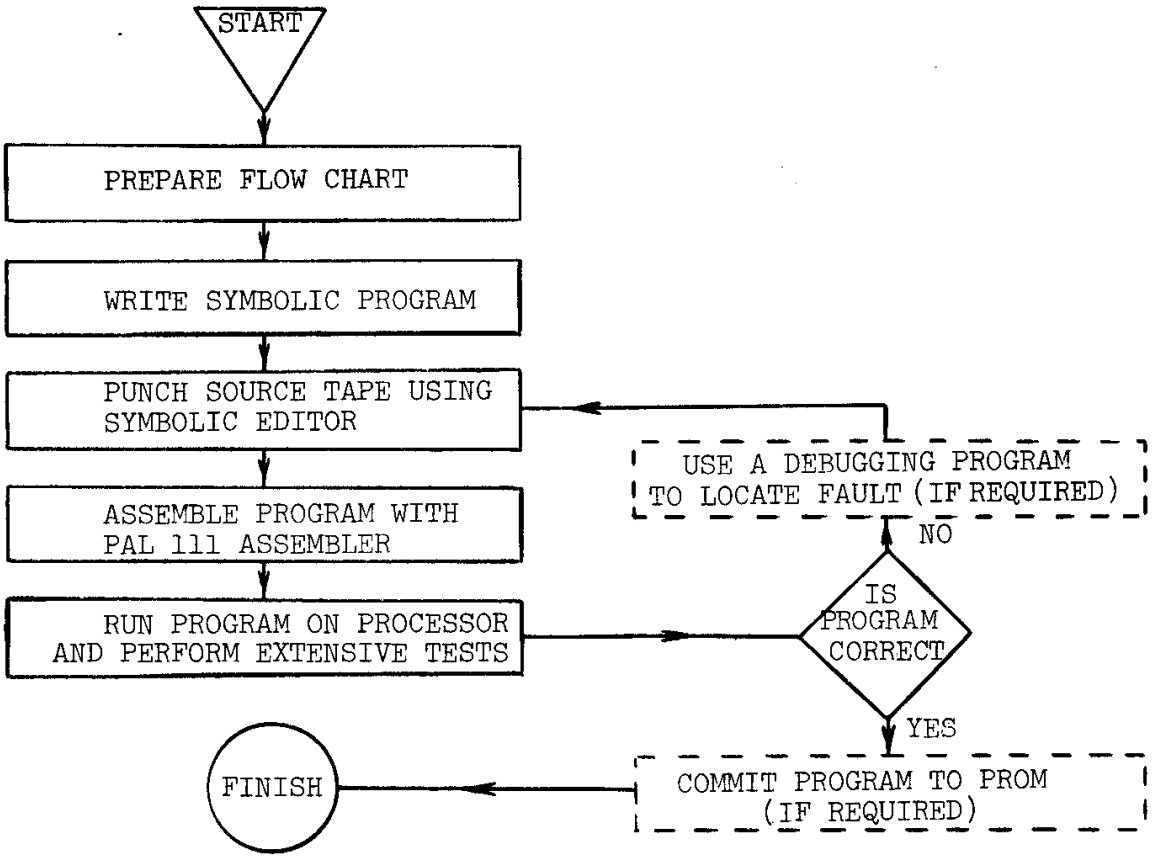

Fig. 5. Flow Chart of Program Writing Procedure

of a specified character and allows blocks of text to be inserted, deleted, appended, listed or changed. Finally, the Symbolic Editor punches the Symbolic Program tape (source program tape).

The program is now converted to machine code using the PAL-111 Assembler supplied as part of this package. The PAL-111 Assembler is loaded using the Bin Bootstrap program in the control panel memory. The source program tape (containing the program in Symbolic code) can now be read. This is a three pass program on the first pass the Assembler produces a symbol table containing all the users symbols. On the second pass the binary equivalent of the source tape (or machine code tape) is punched. The third pass is optional, it produces a printed assemble listing of the program instructions with the location, generated binary and source code side by side on each line.

The user program can now be run on the computer by loading it using the Bin Bootstrap program in the control panel memory. Thorough and exhaustive testing must now follow to check the user program. If correct, the program can then be committed to PROM.

The commitment to PROM should not be performed too early since it is not unknown for a fault to be identified after several months. Also it may be desirable to modify the program.

If required the program can be run using other programs from this package, these include the Dynamic Debugging Technique (DDT) or the Octal Debugging Technique (ODT). These are loaded using the Bin Bootstrap program in the control panel memory. They allow the user to run the binary program on the computer and use the Teletype Keyboard to control program execution, examine registers, change their contents and make alterations to the program. With the DDT, the user can debug the programs, using the symbolic language, with the DDT performing all translations to and from the binary representation, ODT has the same capabilities as the DDT, except that the programmer must use octal representation instead of the mnemonic symbols. The use of DDT and ODT is optional, hence their box is shown dotted in Figure 5. They are used mainly for small programs or parts of a large program as they consume memory sapce. ODT consumes less memory space than DDT but requres octal representation.

Other programs are supplied as part of this package, these are as follows:

- Binary Loader:
  This is used to read binary paper tapes into memory. It is supplied on a paper tape in RIM (Read In Mode) format and a RIM loader must first be keyed in from the control panel. It provides an alternative to the Binary Loader which is in the control panel memory of the Intercept.

- RIM and BIN (Binary) Punch:
  Used to punch data from memory on to paper tape in either RIM or binary format.

- Octal Memory Dump:
  Used to write the contents of selected memory locations on the teletype.

- Mathematical Routines:
  These provide for addition, subtraction, multiplication and division using 23 bit floating point numbers. They also provide for the calculation of logarithms, exponentials and basic trionometric functions.

FOPAL-111 PAL-111 Fortran Cross Assembler This is the same as the PAL-111 Assembler but written in standard FORTRAN. Hence it can be run on a large computer installation with sufficient memory capacity to eliminate the tedious rerunning of the source tape.

PROM Based Control Panel and Main Memory Test

4K RAM and Processor Test

PDP8/E Diagnostic Software Kit

These three programs are used to perform extensive tests on the processor, control panel, memory and the teletype. They are used to check the Intercept and the Engineering Model.

Special software will be available to test the complete processor system and will include:-

- Load programs or data into the Engineering Model via the DMA ports.

- Transfer set patterns (all ones, all zeros, alternate ones and zeros, etc.) between the Intercept and Engineering Model via the DMA, parallel and serial input/output ports and check for errors. The input and output of a parallel port can be connected together. This enables the parallel ports of one machine to be checked independently of the other.

Loading programs into the 'E' Model Programs can be loaded into the 'E' model in two ways. By committing the program to PROM then plugging the PROM into the 'E' model or via the DMA port into RAM. This is demonstrated by the diagram below:-

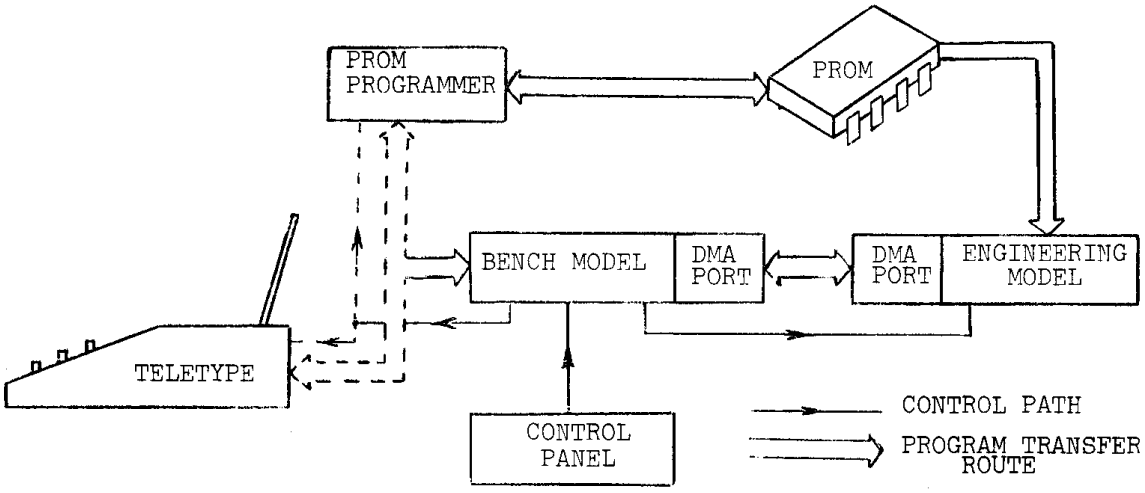

Fig.6. Loading programs into the 'E' Model

If the program is required in PROM in the 'E' model it is first input to the Bench model via the teletype, together with a PROM programming program. The teletype is then replaced by the PROM programmer and the program is loaded into PROM under the control of the Bench model control panel. The PROM contents can also be read back and checked. The PROMs are then plugged or assembled into the 'E' model. Finally the 'E' model memory

contents can be read into the Bench model via the two DMA ports and the PROM contents again checked. Program can be loaded into RAM if required. All the necessary control lines for this are brought out and program can be loaded from the Intercept memory via the two DMA ports into the memory of the Engineering Model. This is a useful facility in a flight model since it allows reporgramming in Space via Telecommand. It is performed by means of a section of PROM in the main memory and by using the switches on the Peripheral Test Set control panel.

Instruction set The processor has 6 twelve bit registers, a programmable logic array, an arithmetic and logic unit and associated gating and timing circuitry.

The 12 bit word is considered the best compromise between the mass and power requirements of the store and tolerable quantization error. Quantization error (one sigma level) is plotted in fig.7. for different word lengths. From this it is considered that a word length of 12 bits is sufficient for most applications. If more accuracy or more processing steps are required a double length word of 24 bits can be used.

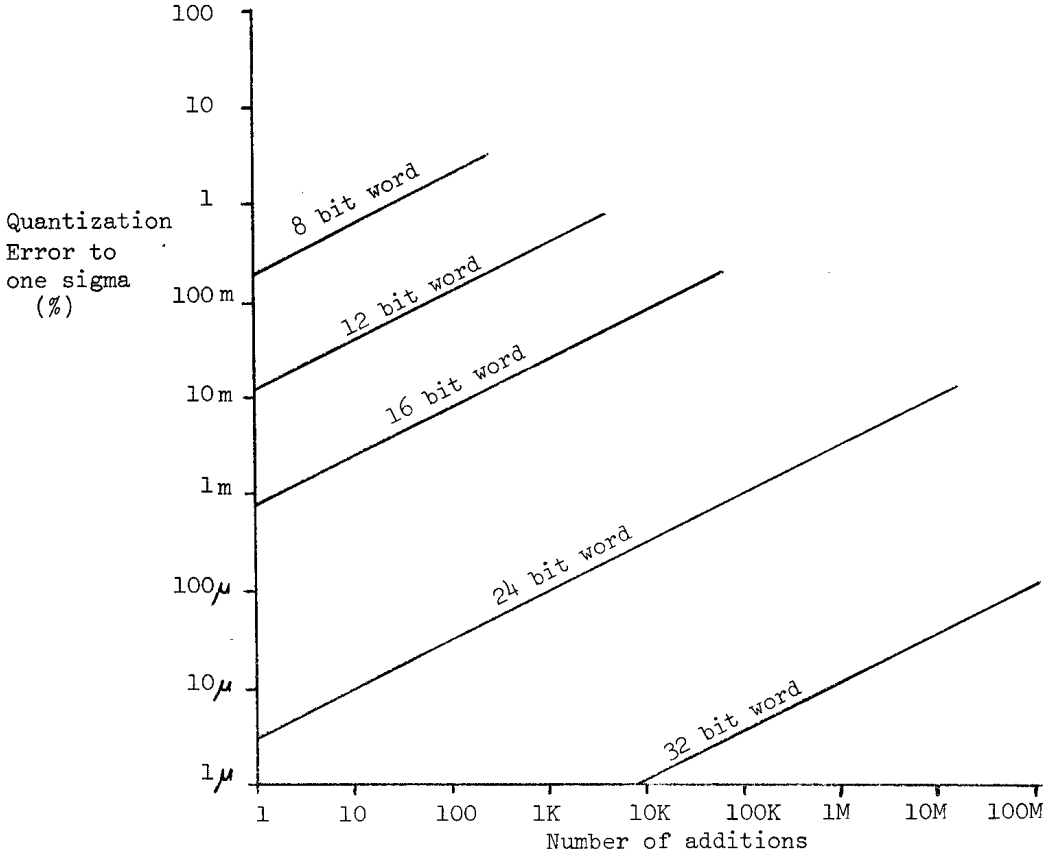

Fig 7. Graph of Quantization error

Programming is performed in machine or assembler code. Each instruction is held as a 12 bit binary word and the processor makes no distinction between instructions and data. Instructions can be manipulated as data and data executed as an instruction. There are three classes of instructions:

> Memory Reference Instructions (MRI); Operate Instructions (OFR) and Input/Output Transfer Instructions (IOT).

The full memory is comprised of eight Fields each consisting of 4K words of 12 bits each. The Field is selected by software after which all MRI instructions refer to that Field. Each memory Field is divided into 32 Pages of 128 words each. Memory Reference Instructions operate on the contents of a memory location or use the contents of a memory location to operate on the Accumulator or the Program Counter. The first 3 bits of a Memory Reference Instructions specify the operation code and the low order 9 bits the operand address. The operand address is given as a word in the current Page or in Page 0. Also direct or indirect addressing is indicated. With indirect addressing the data taken from the current Page or Page 0 is used as the address of the operand. The permitted operation codes are 0 (logical AND), 1 (binary add), 2 (increment and skip if zero), 3 (write to memory from the Accumulator) 4 (jump to Subroutine) and 5 (jump).

All Operate Instructions have the operation code 7 and perform operations on the Accumulator. They are split into 3 groups of microinstructions. Group one microinstructions perform logical operations on the contents of the Accumulator such as rotating, complementing, incrementing, clearing and setting. Group two microinstructions are used to test the contents of the Accumulator to see if it is negative, positive or zero. They are also used to read the Switch register. Group three microinstructions are used to perform logical operations on the Accumulator and an extra processor register known as the MQ register. The MQ register can be loaded from the Accumulator, cleared, have it contents read into the Accumulator or the contents of the two registers can be swapped.

An Input/Output Instructions have the operation code 6 and initiate the operation of and transfer data between the processor and peripheral devices. The second (or middle) 6 bits are used to select a device and the lowest order 3 bits specify the control. The control indicates if a read, write test etc. is required, the actual code is dependent on the Input/Output device.

After an instruction is completely sequenced, the major state generator scans the internal priority network. The state of the priority network decides the next sequence of the processor. The internal priority is processor reset, control panel memory select, run/halt, Direct Memory Access request, peripheral device interrupt request and finally an instruction fetch cycle is entered.

Conclusions The microcomputer described is a general purpose machine that has many applications. It features an all CMOS design with extensive use of LSI giving high speed, low power consumption and small size.

A modular packaging scheme is used that allows any number of different modules to be assembled to form a complete unit. In this way a unit can be assembled that best fits a particular set of mission requirements.

An extensive range of software already exists for the microcomputer. The architecture and instructions used are the same as for the PDP8/E minicomputer manufactured by DEC. Hence the large library of DEC software is available for use on the microcomputer.

A Bench model of the microcomputer is available for control and program development. This has the same facilities as the 'E' model microcomputer with the added facilities of a control panel and teletype. This can be used in conjunction with a manual test set to test and load programs into the 'E' model.

RAM and PROM can be mixed as desired (in blocks of 256 words) in the memory of the 'E' model microcomputer. The memory is expansible in blocks of 2K words.

Both Serial and Parallel digital Input/Output ports are provided. These interface at 10 volt CMOS levels. The system can be operated off 5 volts if required, it will then interface directly with TTL.

Dropping the supply voltage from 10 volts to 5 volts reduces the power consumption to 25% but also halves the speed of operation.

Quantization error is a very important consideration during usage and will be the dominant error if an insufficient number of bits are used. The use of too many bits is wasteful of memory, processing time, mass and power.

*"This paper is based upon work performed under the sponsorship of International Telecommunications Satalite Organization (INTELSAT)." (Any views expressed are not necessarily those of INTELSAT.)