INTRODUCTION

by Mark DiVecchio
Section Head - Logic Design

The following series of articles were written by the members of the Engineering Department at the San Diego facility who were directly responsible for the design and test of the AS/6100. Due to space limitations, the articles were kept to a semi-technical nature but we will be happy to answer any further questions you might have. Just contact the author directly.

SERVICE PROCESSOR FEATURES

by Frank Dean
Section Head - SVP Software

'A COMPUTER WITHIN A COMPUTER'

Located within the mainframe of an AS/6100 is a microcomputer system called the service processor (SVP). The service processor performs a multitude of diverse system services including system status monitoring, display, and control.

The SVP consists of:

1. Two microprocessors connected in an attached processor configuration.
2. 56K RAM.
3. 8K ROM.
4. Logic boards to control the following devices:
   a. Two flexible diskette drives.
   b. Operator/service console (CRT/keyboard).
   c. Optional alternate console.
   d. Optional printer.
   e. Remote support modem.
   f. Console clock.
   g. Interface between the SVP and the main processor CPU.

Three new features supported by the AS/6100 service processor are discussed below.

'ALMOST LIKE BEING THERE'

Occassionally a problem may occur with an AS/6100 that's too tough for the on-site Field Engineer to solve. A NAS AS/6100 specialist is then needed and fast. But it takes time and money to fly a specialist to the customer's site. What's the alternative? Remote Support, of course. The on-site Field Engineer simply phones the NAS Remote Support Center and requests specialist services. The specialist sits down to his console keyboard and proceeds to monitor, control, and diagnose the problem of the ailing AS/6100 hundreds of miles away.

To the specialist it's almost as if he were at the site. To the customer it's almost like having him there.

How is this done? Well, it takes both service processor hardware and software. A modem and a modem controller are needed to transform digital data into audio (phone) signals and vice versa. The service processor software sends the specialist's keystrokes to the site where they are acted upon as if they were keyed in at the AS/6100s console keyboard. This keyed data results in changes to the CRT screen of the AS/6100 operator's console. These CRT

changes are then transmitted to the specialist's console at the Remote Support Center. The fixed portion of all CRT display frames are permanently stored at the Remote Support Center so only the variable portion of each CRT frame need be transmitted. The service processor performs data encoding, error checking and correction, and data compression to minimize data transmission time.

'DO IT IN SOFTWARE'

An AS/6100 has certain parameters that can be specifically configured according to each customer's individual requirements. In some computers this configuration is accomplished via hardware switches which are directly readable by the Central Processing Unit (CPU), but not by the service processor.

In the AS/6100, a better way was found to accomplish this configuration. Instead of hardware switches, the data is entered via the service console keyboard under the control of the service processor software. There are several advantages to this method:

1. No switches means less hardware, less expense, less space, less that can fail.
2. SVP software makes data entry physically easier and reduces the chance of erroneous entries. It can show the user exactly what the configuration is, whether it is valid, and if it is invalid, why.
3. SVP software can send the configuration data to the Remote Support Center and a remote specialist can examine and/or modify the configuration as necessary via phone line.

'WHEN DID IT HAPPEN?'

Did a power glitch occur? A memory failure? A remote support session? When?

The AS/6100 can answer this thanks to a National Semiconductor MM58174 real time clock chip. This chip is the heart of the service processor's clock which keeps track of the year, month, day, hour, minute, and second. It is set at installation time and switches automatically to battery power if the AS/6100 is powered-down. Whenever an exceptional event occurs, the AS/6100 service processor reads the clock and logs the time stamp along with other event data. Thus a valuable chronological event record is maintained.

THE AS/6100 SERIALIZATION SCHEME

by Colin Isenman
Senior Engineer

1. The serializing function in a computer of this scale is a key element in the initial bring-up of a new machine and in the field maintenance and troubleshooting role. It is the feature which allows any kind of diagnostics (especially "remote") to investigate the condition of many registers, busses, latches, and status flags in the hardware.

2. In the AS/5000 Series machines, this function is performed by a set of randomly distributed multiplexer devices which form a successive funnel arrangement to encompass the whole machine. The single output line which results from this funnel is fed across to the service processor, which also supplies the requisite addressing to summon the data. In machines of this era, the addresses are decoded as required to perform the desired function.

3. In the AS/6100, some subtle design changes have made the serialization system more elegant, relevent, and easy to use. Essentially, the machine is broken down into logic cards. Each card is now a self-contained serialization unit with address lines going in and a single data line going out. The flexibility was given to the logic designer to assign the bytes in the card serialization in a way that made sense in terms of this hardware architecture, and also minimized the amount of logic required to implement the functions. Also, the number of address lines fed to the logic card was minimized to assure the simplest design possible. Further, the logic cards were assigned to "groups" on a hardware location basis. These cards need only to recognize their own group and ignore all others.

   The key to their reduction in logic was the introduction of a translation between the console log address, displayed on the console CRT, and the hardware address recognized by the logic cards. This translation is accomplished with the use of ECL programmable ROM devices, resident in the console interface logic.

   A further reduction in logic was obtained by utilizing a feature of the printed circuit technology used for the backpanels. With adequate care, it is possible to make a single data bus which travels around every logic card in the machine. Since each logic card recognizes its own identity, it can be arranged only to send data onto this bus when required and to have no effect on the transmission of signals along this bus at other times. This technique has been used to eliminate the large number of ICs and substantial wiring associated with the funnel arrangement used on previous machines.

THE AS/6100 CLOCK SYSTEM

by Colin Isenman
Senior Engineer

The AS/6100 makes major advances in our system technology. In particular, the advent of multilayer fine-line printed circuit boards allows photographic accuracy and repeatability between systems and a high degree of predictability is thus available at the design stage. The AS/6100 uses this technological ability as the cornerstone of its clock system, allowing the vigorous enforcement of strict design standards and a statistically conservative design methodology.

In essence, the clock system consists of four parts:

1. The high stability crystal oscillator.
2. The clock generator and timing circuitry.
3. The clock distribution system.
4. The design of the individual logic cards.

These are described in the following paragraphs.

1. THE OSCILLATOR

The AS/6100 employs a crystal oscillator as its timing source. The oscillator is strictly specified and tested to ensure its conformance. The major demands placed upon it are high accuracy and stability over a range of temperature and power supply loads, and a well controlled waveform to ensure full ECL compatibility.

2. THE GENERATION AND TIMING CIRCUITRY

In a machine of this complexity, several different types of clocks are required; for example, gated and ungated, very narrow or very wide. Each of these is generated in its own generation circuitry, which is fed from the master oscillator via the timing circuits. So, the timing adjustments are before, rather than after the generation of the clock pulses. This ensures that the consistency between clocks (also known as skew) is excellent. At this stage of an AS/6100, the alignment is within one quarter of a nanosecond. This timing adjustment is the only one required in the entire machine, and is performed at board test in the factory. Consequently, all clock cards are equivalent in performance, and therefore field replaceable without further adjustments.

In the AS/6100, clock width is under the direct control of the crystal oscillator. It cannot be incorrectly set. With this kind of design, it is now possible to vary the machine speed with no readjustment. The AS/6100 makes use of this facility by providing speed margining in addition to the traditional power margining, and also allows operation at half of the normal speed to aid in shooting logical problems without any timing related effects confusing the picture.

3. THE CLOCK DISTRIBUTION SYSTEM

Having generated the stream of closely aligned clocks, the problem of getting them to their final destination (the individual logic card) remains. The AS/6100 takes special care to ensure that the clocks arrive in positive condition. Firstly, the printed circuit from the clock generators to the distribution amplifiers is carefully designed to be the same for all clocks. Secondly, the amplifiers themselves are all pre-screened on the Sentry™ automatic test equipment to have a well controlled delay. Thirdly, the printed circuitry from the output of the distribution amplifier to the edge of the clock card is again carefully designed. In combination, this means that the only significant variation introduced in the clock system up to this point is the delay variation of the distribution amplifiers, which are pre-tested. So, we can assemble the clock card secure in the knowledge that it falls well within the design limits.

As the clocks leave the clock card, they are fed through precision co-axial cable assemblies to the destination cards. This is crucial in the system design philosophy, for it allows uniform timing across the machine, both on boards which are close to and far removed from the clock card.

4. THE INDIVIDUAL LOGIC CARDS

No matter what it's function, all AS/6100 logic cards share a common set of design rules. Each was clearly enunciated and strictly enforced during the design stage to ensure a quality product with designed-in reliability. As an example, the designer was responsible for guaranteeing that the lengths of the clock signal connections within his board fell within defined limits, and that certain physical constraints were used on the individual logic cards as on the clock card itself, when a clock signal was involved.

The result of this design activity is a machine with excellent predictability of timing. The major reasons that this is possible and that it continues to be possible on machine after machine is the intelligent use of fine-line printed circuit technology, and a commitment on the part of the logic designers to build in reliability and ease of maintenance.

THE AS/6100 ECL CUSTOM LOGIC ARRAY (ECLA)

by Mark DiVecchio
Section Head - Logic Design

The AS/6100 uses state-of-the-art ECL Custom Logic Array (ECLA) technology. The ECLA is a VLSI device that bridges the gap between custom designed VLSI and standard off-the-shelf devices. The ECLA starts its life as a substrate of unconnected transistors and resistors. The logic designer can specify that groups of these transistors be connected together to form logic cells containing elements such as AND gates, Flip-Flops, and Multiplexers. The designer then specifies how these cells are connected to perform the required logic function. The benefit in this is that the substrate is common among all ECLA and can therefore be fabricated in large quantities at a favorable price.

The ECLA will give the AS/6100 new levels of reliabiilty and serviceability. The ECLA means fewer parts, fewer connections, and less heat to dissipate. This not only increases reliability but also performance since the AS/6100 could be physically small, that smallness translated into a more powerful processor. Improved serviceability comes from more logic on each Field Replaceable Unit (FRU). Built in diagnostic aids will easily isolate to the failing Field Replaceable Unit (FRU) in most cases.

During the design of the AS/6100, ECLA were used for mainly two purposes: 1) machine speed enhancements, and 2) logic reduction. One of the most speed critical circuits in the AS/6100 is the High Speed Buffer Index Array Compare. The operational speed of this circuit was almost doubled through the use of the ECLA and each of these ECLA replaced about 19 standard ECL ICs.

At the other end of the spectrum, the Mover logic in the CPU, which is not inherently speed critical, benefited from ECLA in that two ECLA devices replaced about 172 standard ECL ICs.

The design of the ECLA followed a comprehensive and detailed process. The incorporation of the ECLA into the AS/6100 necessitated changes in the hardware algorithms for the Decimal Multiply, the Convert functions, and the Binary Multiply. These changes let us use the logic reduction feature of the ECLA to reduce the component count of the XAR from over 500 in the AS/5000 down to 250 in the AS/6100 while at the same time almost doubling its computational power.

Since it is extremely difficult to change the design of an ECLA once it is "cast in silicon," a great deal of time and effort is spent at the paper design phase to ensure functional correctness before the design is committed to chip fabrication.

STEP 1 - LOGIC DESIGN

The project engineers responsible for each functional unit in the AS/6100 selected the appropriate candidiate hardware for inclusion into the ECLA. These selections were reviewed by a group of all of

the project engineers and managers. Then two engineers were assigned to develop the detail design of each ECLA. One had primary responsibility and the other served to double check the first (of course, he had primary responsibilities for another ECLA himself). Once the detail design was complete, it was again reviewed. Once approved, it was entered into our Computer Aided Design (CAD) system. This performed more checks on the design. At this point, the ECLA path split. The design drafting department took the detail design and produced placement and interconnection information on our CALMA system. At the same time, the design engineer was producing a set of test patterns on a software simulation of this design. Once these two tasks were complete and following a final review and double check, the CALMA information (called a MASK tape) and the test patterns were sent to the vendor for fabrication.

STEP 2 - FABRICATION

At the vendor, a silicon wafer four inches in diameter with the pre-etched substrate is selected and the placement and interconnection information from the MASK tape is used to place the final layers of metal interonnection on the wafer. Following this processing, the dozens of ECLA on each wafer are tested, sectioned, mounted on a ceramic carrier, and retested. The testing uses both predefined voltage and current parameters as well as the logic test patterns generated by the design engineer.

STEP 3 - SYSTEM TEST

Once the parts were back in San Diego, extensive system level tests were performed. First though, the ECLA was retested on our Sentry VII Automatic Test Equipment (ATE). After verifying that the ECLA device met all of our rigid requirements, the design engineer built up a test circuit to permit his design to be tested in a system simulator.

Our success rate in ECLA design was remarkably high. Due to the investment in the front end of the design only one ECLA had a significant problem detected during system test which required redesign. Our procedures enabled us to find the problem and correct it with no impact to the overall design effort.

As a testament to our efforts, our primary ECLA vendor thought so highly of one of our designs that they selected it to be a standard product offered by that vendor. The Arithmetic Logic Unit (ALU) ECLA will not only be in the AS/6100 but may end up being used industry wide.

In summary, the ECLA design team took on a leading edge effort in the use of this VLSI Custom Logic Array in a commercially available processor. The effort for the AS/6100 was a significant stepping stone toward the use of even denser VLSI in our future products.

ECPS/VSE EMULATION

by San-Yen Shi
Senior Firmware Programmer

INTRODUCTION

When IBM announced that the 4300 processors would be able to run in both the System/370 mode and the Extended Control Program Support: Virtual Storage Extended (ECPS:VSE) Mode, NAS Engineering in San Diego decided to emulate the ECPS:VSE mode for the new AS/6100 Series. The ECPS:VSE mode includes a new storage-control facility, called one-level addressing, for creating a single virtual storage of up to 16,777,216 bytes, with both the CPU and the channel address directly using one uniform set of virtual addresses. Mapping the virtual storage onto the real storage is performed internal to the machine.

The one-level addressing facility provides new instructions and interruptions which the control program uses to determine which parts of virtual storage are currently mapped into real storage and thereby are made addressable. These instructions and interruptions, and the associated internal address-mapping functions, take the place of dynamic address translation (DAT) and channel indirect addressing in System/370 mode.

The ECPS:VSE mode also includes a new status-saving function, called Machine Save, which preserves the entire CPU state and the first 2,048 (2K) bytes of storage. The operator uses Machine Save in preparation for a complete storage dump. Machine Save replaces the store-status function of System/370 mode, which necessarily alters some of the storage to be dumped.

EMULATION OVERVIEW

The ECPS:VSE emulation feature consists of microcode that enables AS/6100 to operate in ECPS:VSE mode making it compatible with the IBM 4300 processors running in the same mode. The AS/6100, while running in the ECPS:VSE mode, will execute microcode sequences for each of the 4341 single level address translation instructions. This allows the user to run programs under DOS/VSE in ECPS:VSE mode with no changes to the operating system or the programs.

The ECPS:VSE emulation allows the system to operate in single level addressing mode. Instruction addresses are translated during execution and addresses in channel command words (CCW) are translated during I/O operations. The single level virtual address technique allows for one virtual storage space of up to 16 million bytes.

IMPLEMENTATION

One major problem in emulating the ECPS:VSE mode is that the AS/6100 does not have the channel translation lookaside buffer to translate single level addresses into real addresses. Therefore, the AS/6100 uses a set of tables in main memory to translate virtual addresses. These tables are used exclusively by microcode to do the same functions for channels that the dynamic address translation hardware accomplishes in System/370 mode for the CPU.

In both System/370 mode and ECPS:VSE mode, each virtual address, whether for an instruction or data, is translated. A 24-bit virtual address is computed using base, displacement, and index. The virtual address is then translated. The result of this translation is a real address that indicates the actual storage location to be used. However, in ECPS:VSE mode this real address is not available to the user but is used only by the microprogram.

TABLE STRUCTURE

ECPS:VSE mode reserves main storage locations for several tables used for the firmware emulation of single level addressing. Three of these tables are the nuclei of the emulation.

1. Address Translation Table

   This table is used for translating virtual addresses to real addresses and contains other page status. (This table is also known as the page description table.) Each entry in this table contains information about the page status, key, and page frame address associated with that page. Virtual storage is divided into pages, each containing 2,048 (2K) bytes of contiguous storage starting on a 2K-byte address boundary.

2. Page Table

   This is identical to the page table defined for use in System/370 mode with 2K-byte pages. Note that the invalid bit in this table indicates the addressability of the page.

3. Segment Table

   This table has the same format as that for a System/370 mode segment table with 64K-byte segments.

CPU STRUCTURE

There are three new facilities affected by resets when in the ECPS:VSE mode:

- Capacity counts
- Page description table (address translation table)
- Machine save information

There are several new CPU instructions for ECPS:VSE mode. These instructions change the page state in the page description table. In ECPS:VSE mode, a page may be in one of these states:

- Disconnected

- Connected

- Addressable

If disconnected, the page does not have a page frame assigned to it. If connected, the page has a page frame assigned to it. If addressable, the page has a page frame assigned to it, and the CPU and I/O channels may access the page if the protection mechanism permits.

CHANNEL STRUCTURE

There are three major changes in channel microprogram sequences that provide the channels with virtual address capability required for the ECPS:VSE mode compatibility. The changes are in sequences which make use of storage addresses:

- CCW accesses

- Initial CCW data access

- Subsequent CCW data access when page boundary is crossed

CCW ACCESS

Storage accesses made to fetch CCWs are done in one common channel microcode sequence. The CCW address translation process involves:

1. Finding the page state of the CCW address.
2. Checking the page state of the CCW address.
3. If the page is disconnected, then generate a channel protection check interruption.
4. Else use the real address from the page description table to fetch the CCW.

INITIAL CCW DATA ACCESS

After accesses are made to fetch CCWs, the channel must be initialized with a storage protection (SP) key and the CCW data address. The data address is translated by a method similar to the CCW address translation. However, at this time no error can be presented if the CCW's data address is in a disconnected page since there exist many CCW command codes which require no data to be moved. Thus, a channel protection check interruption is generated only if the page status is disconnected and data is to be accessed.

SUBSEQUENT CCW DATA ACCESS WHEN PAGE BOUNDARY CROSSED

After accesses are made to fetch a CCW and the channel has been initialized with the CAW SP key and the CCW data address, data transfers will proceed until the boundary of the page is reached. At this time, the old virtual data address will be fetched, updated by 2,048, and then used as the new data address to be translated. Error conditions will be handled as in the initial CCW data address.

SUMMARY

ECPS:VSE emulation is fully implemented in firmware. By maintaining the software compatibility with IBM and without increasing the hardware cost, ECPS:VSE emulation will enable NAS AS/6100 Series processors to compete along all fronts.

AS/6100 FAULT ISOLATION SYSTEM

by Dyke Summers
Section Head - Diagnostics

The fault isolation system for the AS/6100 was designed to reduce mean time to repair and also reduce the level of training required for field personnel.

The purpose of the fault isolation system is to provide an isolation code whenever an unrecoverable machine check occurs. This isolation code is displayed on the console CRT along with the control store address register at the time the machine check occured.

The customer, when phoning for service, will give the Field Engineer the isolation code. The Field Engineer will look up the code in a fault isolation dictionary and determine the most likely FRU required to service the machine.

IMPLEMENTATION

To provide flexibility and reduce costs, a 100% software approach was used. The software, which operates out of 4K of service processor RAM, resides on the IMPL diskette and is rolled in only when required.

The system consists of two major software components.

a. The fault isolation equations which are written in a simple, specialized language. These statements are written as equations which will test log buffer data, set or test flags, branch or set isolation codes.

b. An interpreter to evaluate the equations. The interpreter, after evaluating the fault isolation equations, displays the isolation codes on the service processor CRT and then returns control to the service processor main supervisor.

   The interpreter operates like a small computer with its own instruction set and registers. Each instruction must contain three elements:

   1. A mnemonic for a single 'execution operand' (EOP).
   2. A data field containing one or more log buffer addresses or internal flag names.
   3. One or more result operands (ROP).

   The EOP, which is a single logical operation, will operate on all elements of the data field. Each EOP, in addition to the logical operation, contains an implied test. If the test condition is met, the ROP will be executed, and then control will pass to the next instruction in sequence. If the condition is not met, the next instruction is executed.

The following two statements and their interpretation will illustrate the sentence structure and the interpreter's operation.

1. AND 46-1, 37-3, 48-P, FLAG-6, CODE-8B, CSBAR2.
   (a) Bit 1 of log byte 46, bit 3 of log byte 37, bit P of log byte 48, and FLAG 6 will be anded.
   (b) If all of these bits are 1s, the test condition has been met and the ROPs are executed in order. CODE-8B will cause the hex number 8B to be placed in the display buffer. CSBAR2 will cause the value of the control store address register, at the time of the machine check, to be appended to the isolation code (8B).
   (c) If any of the bits are zero, the test condition is not met and the next statement in sequence is executed.
2. ORBY 66, 67, 158, SFLAG-6, GOTO CPU1.
   (a) Bytes 66, 67, 68, and 158 of the log buffer are tested for non-zero.
   (b) If any of the bytes are non-zero, the test condition is met and the ROPs are processed in order. SFLAG-6 causes FLAG-6 to be set. GOTO CPU1 causes control to pass to the statement labled CPU1.
   (c) If all of the bytes are zero, the next instruction is executed.

There are approximately 360 error latches in the AS/6100. The ability to operate on many error latches in a single operation and express the result in a single flag setting makes it possible to do a very comprehensive log out analysis in a minimum of space.

AS/6100 PCB DEVELOPMENT

by Ted Langlet
Section Head - CAD

OVERVIEW

The AS/6100 computer system consists of 36 printed circuit boards housed in two large cages with printed circuit backplanes. Each of the printed circuit boards (PCBs) are approximately 16 inches square and have 360 backplane signal connections and 120 frontplane signal connections. All PCBs consist of eight distinct layers laminated together; four layers provide chip interconnections and four additional layers provide necessary electrical power and ground for the chip components.

A total of 30 unique PCB types consisting of various combinations of ECL circuit components were developed; the AS/6100 uses multiple copies of certain PCBs such as the I/O channel boards. A typical PCB has four 68-pin ECLA, 15 RAMs (memory components), and 150 MSI and SSI 16-pin components. The typical PCB also has 750 logic nets (which are groups of component pins to be electrically common), and 2,000 pin-to-pin pairs to be interconnected.

DEVELOPMENT CYCLE

Design of a PCB occurs in several stages. Initially, the logic design engineer specifies the PCB circuit design using logic schematics, i.e., interconnected graphic symbols. This information is entered into a Design Automatic System (DAS) which verifies data validity and generates the PCB logic netlist.

The next stage in the development of the AS/6100 PCBs involved entering the logic netlist and PCB geometry data into a leased computer program called SCI-CARDS. SCI-CARDS offers several options to programmatically determine the best locations for the chip components on the PCB. After the components have been positioned, SCI-CARDS generates pin-to-pin interconnect solutions through a series of increasingly complex routing algorithms. Various routing strategies were employed during this phase to obtain the most desirable interconnection solution; each strategy required approximately six hours of AS/5000 computer time to complete. An average 98% routing completion was achieved; 40 connections per PCB were not completed.

Following SCI-CARDS processing, PCB development proceeded through a manual stage to complete the logic netlist interconnections. First, the SCI-CARDS solution was transferred to a CALMA graphics system where pen plots of the interconnection were prepared. A team of printed circuit board designers then worked with the pen plots to obtain a fully interconnected PCB. The marked-up pen plots were then returned to the CALMA support group for input into the CALMA system. The last step of this stage of PCB development involved running a computer program on the CALMA system to verify

that the manually prepared interconnection data did not violate physical clearances and that it was consistent with the original logic netlist.

The last stage of PCB development started with the generation of a photo-plotter numerical control tape of the CALMA verified interconnection data. Artwork masters of the eight PCB layers were photo-ploted in-house using a Gerber photo-plotter; a complete set of artwork films required about 30 hours of photo-plotting time. Finally, the PCB interconnect solution was transferred to DAS and a variety of manufacturing related documents were prepared including parts lists, continuity and functional test tapes for quality assurance testing, and assembly instructions.

SIGNAL INTERCONNECTIONS

by Bevitt Norris
Staff Engineer

Printed circuitry is used for the majority of signal interconnections in the AS/6100 to obtain highest possible performance and predictability of design.

Factors to be considered and benefits attainable with printed circuitry are:

1. Controlled Impedance

   At the fast switching times of the logic family and ECLA used in the AS/6100, interconnections longer than a few inches must be treated as transmission lines, or uncontrolled ringing due to mismatched impedances will prevent accurate timing. With printed circuitry, the width of lines and spacing to the signal return planes can be accurately controlled and since the dielectric constant of the board material is uniform, the impedance of all lines can be made the same. Then with the line terminated in its characteristic impedance, there is no reflection to cause trouble.

   Controlled impedance wiring is not only less susceptible to external noise but minimizes the generation of noise. This becomes increasingly important in light of the new FCC regulations on electromagnetic emissions.

2. Repeatability

   Since printed circuit boards from system to system are identical, circuit performance will be identical. This has obvious benefits in production and test.

3. Density

   Use of printed circuitry provides extremely high circuit density while still maintaining impedance control. The printed circuit boards in the AS/6100 contain four layers of controlled impedance wiring in a thickness of .093 inch. All layers are separated by signal return planes which also distribute DC power.

   An average logic board contains about 2,000 interconnections.

4. Crosstalk

   Besides reflections due to mismatched impedances, one of the most serious problems with open wiring is signal crosstalk and the variability of this from system to system. Controlled impedance wiring minimizes this type of interference.

The requirement for high circuit density with a signal line spacing of .025 inch and considerations of crosstalk and variable loading of signal lines determined the selection of 50 ohms as the impedance level of the high speed signal interconnections.

5. Maintenance (MTTR)

With troubleshooting raised from the IC level to the board level, Mean Time To Repair will be significantly reduced.

Printed wiring is not confined to the plug in circuit boards. The technique is carried through to the backplanes that interconnect all the boards. Each backplane contains six layers of controlled impedance printed wiring and four signal return planes that help distribute power, all within a thickness of 1/8 inch.

Much of the signal wiring in the backplane is simply a continuation of a 50 ohm controlled impedance line from one board to another. However, the signal busses that interconnect a number of different boards presented a potentially severe loading problem that could lower the impedance of a simple 50 ohm line to an unacceptable level. This problem has been overcome by including one signal layer with an unloaded impedance of 100 ohms. Where bus connections to the backplane are close together, the backplane interconnections are made using the 100 ohm layer. The artificial loading of the stubs into the boards lowers the impedance of these lines to 50 ohms within an acceptable tolerance. Where board connections are widely separated, backplane wiring of busses is via a 50 ohm layer and the discrete reflection from one board is within an acceptable level.

In summary, the use of advanced printed circuit technology provides gains in logic speed, design predictability, consistency, interference suppression, logic density, and ease of maintenance and repair.