

Signal Processing Element Functional Description

Part 1 - Microprogrammed Control Unit, Buffer Store, and Storage Control Unit

JOHN P. IHNAT, WILLIAM R. SMITH, JOHN D. ROBERTS, JR.,
Y. S. WU, AND BRUCE WALD

*Information Systems Group
Office of Associate Director of Research for Electronics*

September 12, 1972



NAVAL RESEARCH LABORATORY
Washington, D.C.

REPORTS IN THIS SERIES

**"Microprogrammed Control Unit (MCU) Programming Reference Manual,"
John D. Roberts, Jr., NRL Report 7476, Aug. 15, 1972**

**"Signal Processing Element Users' Reference Manual," William R. Smith and
John P. Ihnat, NRL Report 7488, Sept. 5, 1972**

CONTENTS

| | |
|---------------------------------------|-----|
| Abstract | iii |
| Problem Status | iii |
| Authorization | iii |
| | |
| 1. INTRODUCTION | 1 |
| | |
| 2. ORGANIZATION | 2 |
| 2.1 Objectives | 2 |
| 2.2 Technology | 2 |
| 2.3 Structure | 2 |
| | |
| 3. HARDWARE DESCRIPTION | 2 |
| 3.1 Sequence Unit | 2 |
| 3.1.1 Next Address Formation | 2 |
| 3.1.2 Logic | 5 |
| 3.2 Storage Control and Bussing | 6 |
| 3.2.1 Function | 6 |
| 3.2.2 Priority | 6 |
| 3.2.3 Addressing | 6 |
| 3.2.4 Buffer Storage Data Channels | 6 |
| 3.3 Data Flow | 6 |
| 3.3.1 Local Storage | 6 |
| 3.3.2 Registers | 8 |
| 3.3.3 Operand Selection | 8 |
| 3.3.4 Arithmetic and Logic Unit (ALU) | 9 |
| 3.3.5 Timing | 9 |
| 3.4 Clock | 10 |
| 3.4.1 Control | 10 |
| 3.4.2 Sequence Unit | 10 |
| 3.4.3 Data Flow | 12 |
| 3.4.4 Hard Stop Conditions | 12 |
| 3.4.5 Cycle Time | 12 |
| 3.5 Maintenance Panels | 12 |
| 3.5.1 MCU Switches | 12 |
| 3.5.2 SCU Switches | 16 |
| 3.5.3 MCU Indicators | 16 |
| 3.5.4 SCU Indicators | 17 |
| | |
| 4. STORAGE CONTROL UNIT INTERFACE | 17 |
| 4.1 Inputs to SCU | 17 |
| 4.2 Outputs from SCU | 18 |
| 4.3 Performance | 18 |
| 4.4 Timing | 18 |

| | | |
|-------|-----------------------------------|----|
| 5. | INTERRUPT CONTROL UNIT | 18 |
| 5.1 | Function | 18 |
| 5.2 | ICU Storage Elements | 18 |
| 5.3 | ICU Operation | 20 |
| 5.4 | Interrupt Return | 20 |
| 6. | I/O SYSTEM ELEMENTS | 21 |
| 6.1 | Organization | 21 |
| 6.2 | Unbuffered Channel | 21 |
| 6.2.1 | Z Bus | 23 |
| 6.2.2 | MCU-to-Z-Bus Interface | 23 |
| 6.2.3 | Unbuffered I/O Operation | 24 |
| 6.3 | Buffered Channel Controller | 25 |
| 6.3.1 | SCC Command and Format | 26 |
| 6.3.2 | SCC Status Information and Format | 26 |
| 7. | CONTROL STORE LOAD | 27 |
| 8. | MCU CONTROL FIELD SUMMARY | 27 |
| 9. | CONTROL FIELD DEFINITIONS | 28 |
| 9.1 | CØND Field | 28 |
| 9.2 | NØT Field | 28 |
| 9.3 | NEXT Field | 29 |
| 9.4 | SAUX Field | 29 |
| 9.5 | WBUF Field | 29 |
| 9.6 | FSU Field | 30 |
| 9.7 | SARA Field | 30 |
| 9.8 | SARB Field | 30 |
| 9.9 | SLSA Field | 30 |
| 9.10 | SLSB Field | 31 |
| 9.11 | ADIN Field | 31 |
| 9.12 | ADØP Field | 32 |
| 9.13 | SAAL Field | 32 |
| 9.14 | SBAL Field | 32 |
| 9.15 | DAAL Field | 33 |
| 9.16 | DBAL Field | 33 |
| 9.17 | LIT Field | 33 |

ABSTRACT

The NRL Signal Processing Element (SPE) is being developed to provide a high-performance signal processing facility for radar, sonar, and communication systems. It is intended to be compatible with the Navy's All Applications Digital Computer (AADC). The SPE consists of four major subsystems; a Microprogrammed Control Unit (MCU), a Buffer Store and Storage Control Unit (SCU), a Signal Processing Arithmetic Unit (SPAU), and Input/Output (I/O) units.

The MCU serves as system supervisor and data organizer for the SPAU and other I/O devices. The MCU includes a 64-bit Control Store, two local stores, an arithmetic element, two busses to buffer memory, an unbuffered byte channel, and a priority interrupt system. Its cycle time is 150 nanoseconds (nsec).

The buffer store and SCU consists of eight fixed-priority Direct Memory Access (DMA), or Buffered, Channels communicating on a 150-nsec cycle basis with up to eight 32-bit-by-4K memories.

The SPAU is a highly parallel, high-speed arithmetic unit capable of performing complex signal processing operations such as FFT and recursive filtering. It is microprogrammed for flexibility in adapting signal processing algorithms. Four multiplies and six additions can be performed in 300 nsec.

SPE I/O and internal communications are provided by DMA buffer data channels, an unbuffered byte channel, and a priority interrupt system. The unbuffered byte channel called the Z Bus communicates both data and control information to all I/O devices. The unbuffered channel burst data rate is 2 million 16-bit words per second.

PROBLEM STATUS

This is an interim report; work on these problems is continuing.

AUTHORIZATION

NRL Problems B02-06 and B02-10
Project Nos. 3F21-241-601 (formerly WF-15-241-601),
2F00-241-601, XF-53-241-003, and XF-21-241-015-K152

Manuscript submitted August 25, 1972.

SIGNAL PROCESSING ELEMENT FUNCTIONAL DESCRIPTION

PART 1 — MICROPROGRAMMED CONTROL UNIT, BUFFER STORE, AND STORAGE CONTROL UNIT

1. INTRODUCTION

The NRL Signal Processing Element (SPE) is being developed to provide a high-performance signal processing facility for radar, sonar, and communication systems. The design of the SPE provides for efficient, flexible solutions to problems suited to digital signal processing machines.

The SPE is intended to be compatible with the Navy All Applications Digital Computer (AADC) technology now under development and to be implemented in that technology as part of the AADC system. The SPE can also be used as a stand-alone processor.

The SPE consists of the following elements:

- Microprogrammed Control Unit (MCU)
- Buffer store
- Storage Control Unit (SCU)
- Input/Output System (I/O)
- Signal Processing Arithmetic Unit (SPAU)

The advanced development models of the SPE elements are implemented with "off-the-shelf" components. Bipolar monolithic storage devices and TTL Schottky family logic are used. Performance specifications include

| | |
|-------------------------------------|----------|
| MCU basic microinstruction | 150 nsec |
| Buffer memory cycle | 150 nsec |
| SPAU-equivalent complex operation | 300 nsec |
| (four multiplications and six adds) | |

This report includes functional descriptions of the MCU, SCU, and Buffer Memories. Descriptions of the I/O Selector Channel and SPAU will be reported separately.

The SPE is designed as a tool for processing digital data streams. The heart of the SPE is the MCU which serves as system supervisor and data organizer for the SPAU and other I/O devices in the system. Microprogrammed operations in the MCU process 16-bit-wide data accessed from 32-bit-wide buffer memories, and control buffered and unbuffered I/O operations to and from SPE devices.

The SPAU performs special data processing operations such as FFT, recursive filtering, and correlation under direction of the MCU. Parallel organization of fast multiply and add logic units allow for high-speed execution of these functions. Interfacing between the SPAU and MCU is via buffer memories and the I/O system.

It is the responsibility of the SCU to manage accesses to buffer memories by the elements of the SPE. The MCU, SPAU, and other buffered devices in the system access

buffer memories independently under their own control, and the SCU resolves conflicts for buffer cycles on a priority basis.

The I/O system is designed to allow expansion of the SPE so that multiple MCU's and SPAU's can communicate and coordinate processing of increased data bandwidths.

Figure 1 is a block diagram of the SPE.

2. ORGANIZATION

2.1 Objectives

The purpose of this development program is to provide a high-speed microprogrammable I/O processor capable of performing 16/32 bit precision operations at main memory speeds. An efficient interrupt mechanism is provided. The architecture allows the programmer maximum control of the hardware, resulting in efficient program implementation.

2.2 Technology

The design speeds are based on storage units of 150-nsec or less cycle time and 80-nsec access time. Three-nsec Schottky TTL devices are used in the logic.

2.3 Structure

The MCU* design (Fig. 2) is centered around four/eight Buffer Storage Modules (BSM) for data and one Control Store (CS) for microprogramming. Two high-speed (20-nsec) Local Stores (LS) provide for effective use of the fast Arithmetic and Logic Unit (ALU). The Storage Control Unit manages bus traffic between the Buffer Storage Modules and the MCU, SPAU, and I/O channels. The Control Store is controlled by the CS sequence unit which combines control word information and certain MCU status conditions into the subsequent CS address. The remainder of the control word is used to control data flow between MCU internal storage and other devices in the system and to I/O channels.

3. HARDWARE DESCRIPTION

3.1 Sequence Unit

The Sequence Unit comprises that logic which controls the sequence of MCU operations by determining the order in which control words are fetched from the Control Store. Sequence Unit operations are controlled by Control Word fields *CØND*, *NØT*, *NEXT*, and *LIT*; Machine Status; Buffer Memory conditions; and Interrupt conditions.

3.1.1 Next Address Formation

The sequence unit has a number of next address operations, both conditional and unconditional, as well as an Interrupt mode.

*A description of the internal operations of the MCU is contained in the second report in this series (see inside front cover).

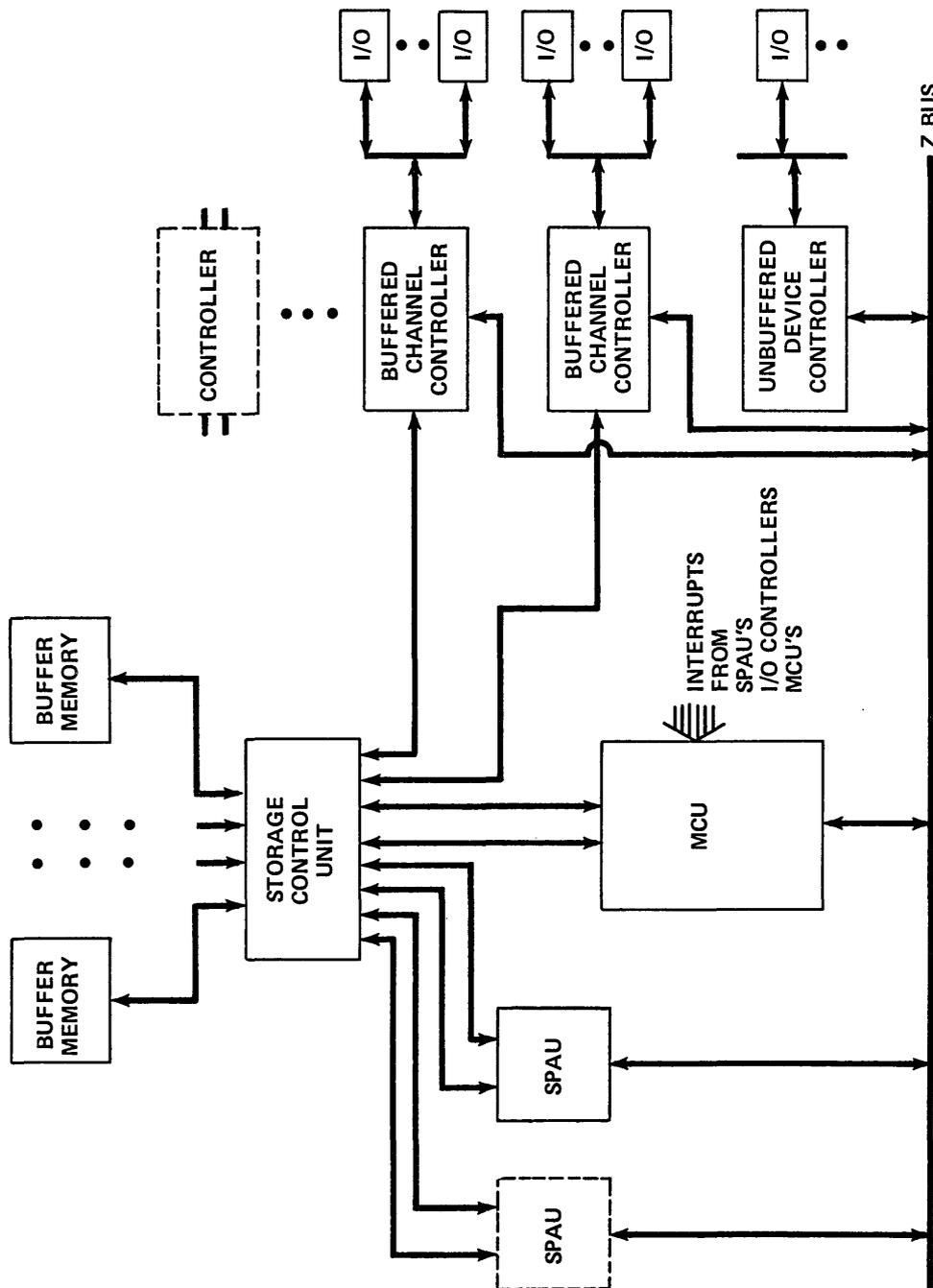


Fig. 1 - SPE system

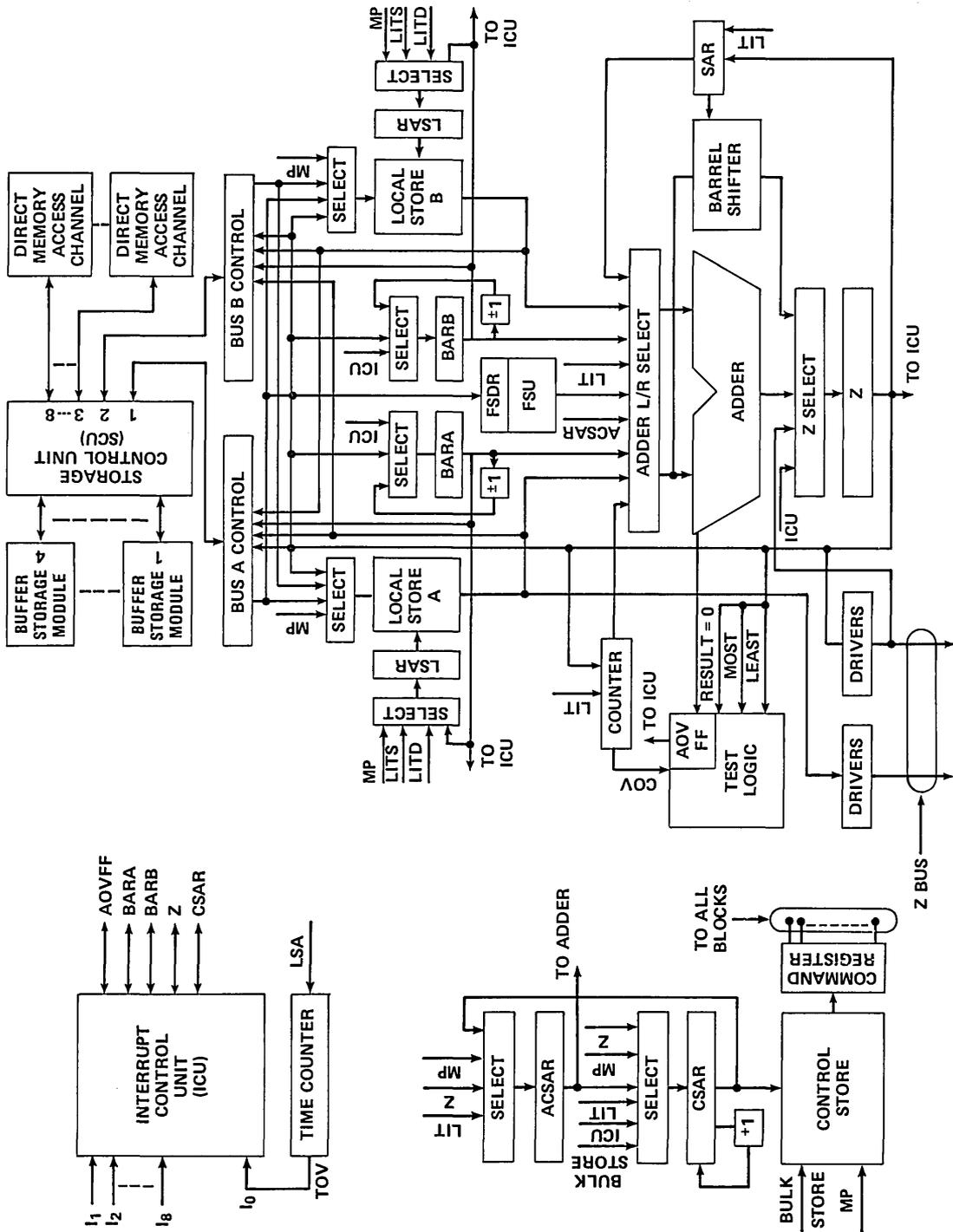


Fig. 2 - Microprogrammed Control Unit (MCU)

- a. Step - Increment CS Address Register (CSAR) by one.
- b. Skip - Increment CSAR by two.
- c. Call - Jump to address specified by LIT field. CSAR plus one is stored in the Alternate CS Address Register (ACSAR) for the subsequent return jump.
- d. Jump (LIT) - Jump to address specified by LIT field.
- e. Jump (ACSAR) - Jump to address specified by contents of the ACSAR (return jump).
- f. Jump (Z) - Jump to address specified by the contents of the Z register (ALU result register).
- g. Interrupt - The address of the next Control Store Word is specified by the Interrupt Control Unit (ICU).
- h. Interrupt Return - The next Control Store Address is specified by the ICU.
- i. I/O Reject - The next Control Store Address is specified by the LIT field.

All next address formations except Interrupt and I/O Reject can be conditioned upon the true or false states of certain MCU conditions as follows:

- a. Z register, most significant bit = 1
- b. Z register, least significant bit = 1
- c. Z register equal zero
- d. ALU adder overflow
- e. Counter register overflow.

The default next address is a step.

3.1.2 Logic

The sequence unit assembles a 12-bit address and accesses Control Store Words (CSW) from a maximum of 4096 words of 64-bit length. It generates the control signals necessary to access the Control Store memory.

3.1.2.1 Control

The NEXT, COND, and NOT fields of the CSW determine the sequence unit operation. The 3-bit NEXT field determines the conditional next address, and the 3-bit COND and 1-bit NOT fields determine whether the conditioned next address or the default step is generated.

3.1.2.2 Registers

The sequence unit uses two 12-bit address registers to enable efficient control of sequencing. The Control Store Address Register (CSAR) contains the current address which is sent to the control store. The CSAR can be incremented, loaded from the LIT field of the Control Register, from the Z register, from the Alternate CSAR (ACSAR), the ICU, or the Maintenance Panel (MP).

To enable efficient response to interrupts, the ICU contains a Secondary CSAR (SCSAR) to which the contents of CSAR are transferred every MCU cycle. An interrupt causes the sequence unit to load the CSAR with an address provided by the ICU.

3.2 Storage Control and Bussing

3.2.1 Function

The Storage Control (SCU), Fig. 3, serves as an interface between the Buffer Storage Modules and their users. Although included with the MCU for the purpose of this specification, the SCU will act as a stand-alone unit. The SCU can control up to 8/16 two-way channels on a priority basis, priority being determined by the respective cable positions.

The SCU includes a priority determination unit, a Buffer Storage Module address selection unit, and a bus control unit. Each of the Buffer Storage Modules has a word width of 32 bits, a 13-bit address register, and up to 8K of Memory.

The SCU resolves requests for a particular Buffer Storage Module by examining requests for memory on a cycle-by-cycle basis. Memory selection and conflicts shall be resolved by examining the high-order address bits, given that a request for memory has occurred. When a conflict arises between higher and lower priority channels, the response to the lower priority channel is delayed until the higher priority channel has been serviced. If the MCU is delayed through such an above conflict, the MCU will be inhibited by suspended clock pulses.

3.2.2 Priority

Priority is resolved for each Buffer Storage Module (BSM) on a memory cycle basis. For each memory cycle, the priority unit examines all requests for storage and determines which BSM is to be used by which requesting user. A request for service will remain pending until it is no longer preempted by a higher priority user.

3.2.3 Addressing

The buffer memories have half-word addressing capability. The SPE system memory words are 32 bits with a maximum of 8K in each buffer. Therefore, a 16-bit address must be transmitted to the SCU. In addition, a control line selecting full- or half-word addressing is required. Up to 32K words of BSM can be addressed; therefore, a 15-bit address is required. The MCU BAR's are 16 bits to permit selection of half (16-bit) words.

3.2.4 Buffer Storage Data Channels

There are 8/16 two-way Buffer Storage data channels, each capable of transmitting a full 32-bit word every memory cycle.

3.3 Data Flow

3.3.1 Local Storage

Two 16-word, random access, Local Stores (LS) are contained in the data flow. The Local Stores can provide 16-bit inputs to the ALU and to the Buffer Stores. The Local

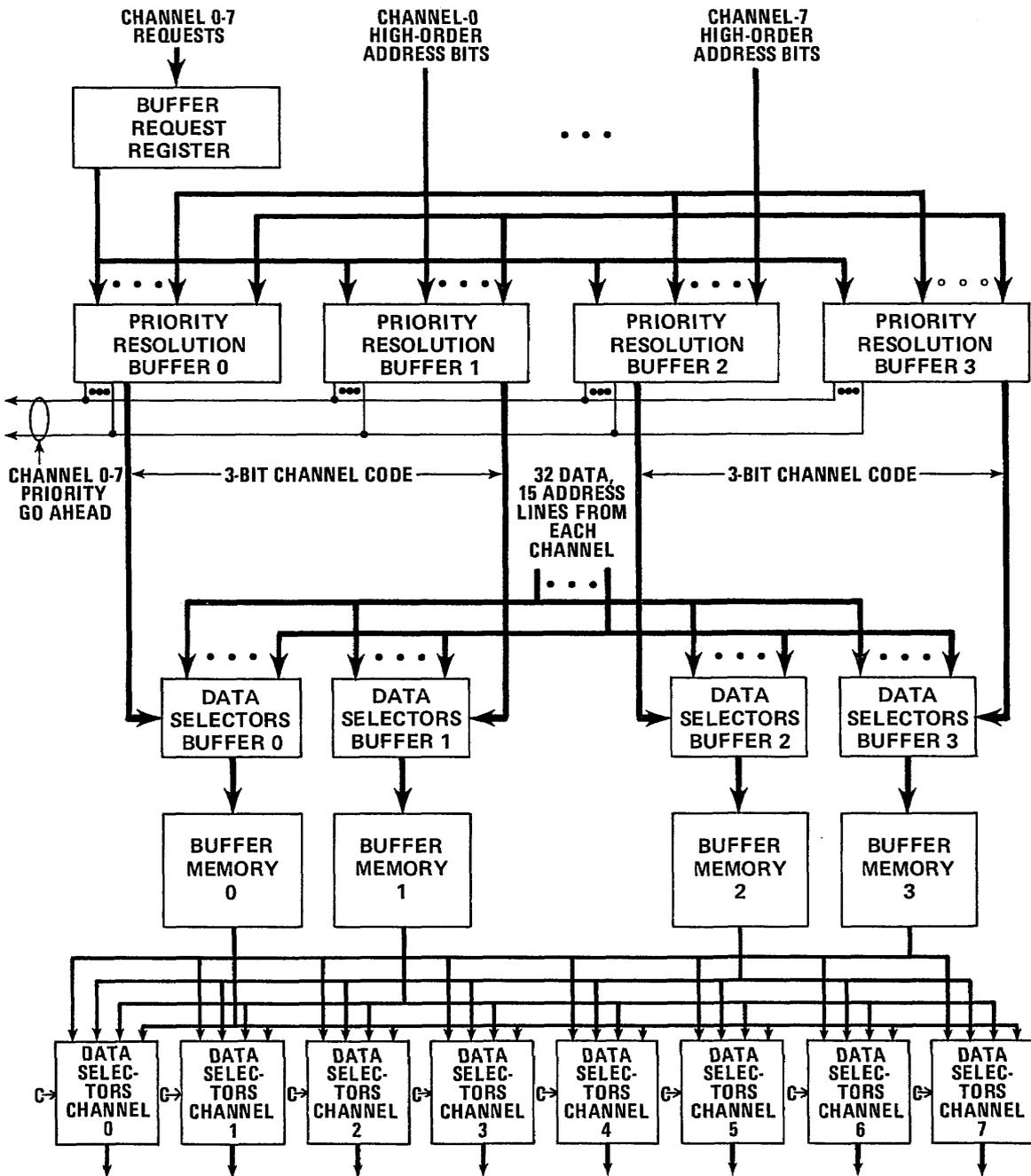


Fig. 3 - Storage Control Unit (SCU). Each channel data width is 32 bits and each channel is two-way. Input C to the data out selectors consists of 3-channel, high-order address bits plus a write control to disable outputs.

Stores have inputs from the Buffer Storage Control Unit (via Bus A and Bus B), the Z register, and the maintenance panel. Each Local Store is addressed from the LS-Address-Literal fields of the Control Register or from a Buffer Address register. The Local Stores can be accessed from and written into in one MCU cycle.

3.3.2 Registers

There are two Buffer Store Address Registers (BARA and BARB) in the data flow. These have inputs to the ALU, Local Store Address Selector, ICU, and SCU. They receive inputs from the ICU and from the ALU. The BAR's provide 16-bit pointers which are used to select and address Buffer Stores and also to address local stores. The BAR's can be incremented and decremented.

The Z register is a 16-bit register used to receive and hold the ALU result for subsequent transmission to a destination and for condition testing. The Z register can also connect directly to the external interface to allow I/O operations. The Z register has as destinations the Local Stores, Buffer Stores, CSAR, ACSAR, Shift Amount Register (SAR) (and Counter), and the ICU.

The Shift Amount Register (SAR) is a 4-bit register which provides control input to the barrel shifter in the ALU. The SAR has inputs from the Z register and the LIT field of the CR.

The Counter is a 16-bit register used to facilitate indexing operations in the MCU. It has inputs from the Z register and the CR LIT field. The Counter also has lines to the MCU test logic to enable incrementing and overflow checking.

The Field Select Data Register (FSDR) is a 32-bit register used to hold full 32-bit words from buffer memory to be fed to the Field Select Unit (FSU). The FSU feeds selected fields of the FSDR directly to the ALU.

3.3.3 Operand Selection

Selection of operands to the ALU is controlled by the 4-bit CS field ADIN. Left (L) and Right (R) inputs to the ALU are available in the following combinations:

| Left Input | Right Input |
|------------|-------------|
| LSA | LIT |
| LSA | FSU |
| LSA | BARB |
| LSA | LSB |
| LSB | LIT |
| LSB | FSU |
| LSB | BARB |
| LSB | BARA |
| BARA | LIT |
| BARA | LSA |
| BARB | LIT |
| BARB | BARA |
| FSU | LIT |
| CNTR | LIT |
| ACSAR | LIT |
| SAR | LIT |

Sources at the left input are available for shifting and unary operations.

3.3.4 Arithmetic and Logic Unit (ALU)

The ALU is a 16-bit parallel adder, logic unit, and barrel shifter that performs a function on the selected L and R operands. The ALU output is stored in the Z register when the function is completed and held as a source to the destination registers later in the control cycle. Z is held through the beginning of the next control cycle to enable condition testing and transfers of results to Buffer Store. Z is available until loaded with the next ALU result halfway through the next control cycle.

The ALU functions and definitions are as follows:

| | |
|---------------------------------------|--|
| NOP | - No operation specified. Z is left unchanged. |
| L plus R \rightarrow Z | - The left and right operands are added algebraically with the result set into Z. |
| L minus R \rightarrow Z | - This is a two's complement subtraction with the result going to Z. |
| L plus 1 \rightarrow Z | - Increment the left input and set the result into Z. |
| L minus 1 \rightarrow Z | - Decrement L and set the result into Z. |
| L \rightarrow Z | - Set the left input into Z. |
| \bar{L} \rightarrow Z | - Set the one's complement of the left input into Z. |
| R \rightarrow Z | - Set the right input into Z. |
| 0 \rightarrow Z | - Set zero into Z. |
| L OR R \rightarrow Z | - A bit-by-bit logic OR of the L and R inputs is set into Z. |
| L AND R \rightarrow Z | - The bit-by-bit logical AND of the L and R inputs is set into Z. |
| L XOR R \rightarrow Z | - The bit-by-bit logical exclusive OR of L and R is set into Z. |
| L EQU R \rightarrow Z | - The bit-by-bit logical equivalence of L and R is set into Z. |
| LEFT SHIFT L \rightarrow Z | - The L input to the ALU is left shifted by an amount contained in the SAR and the results set into Z. The right bits are zero filled. |
| RIGHT SHIFT L \rightarrow Z | - The L input is right shifted with a zero fill. |
| LEFT CIRCULAR SHIFT L \rightarrow Z | - The L input is left circular shifted by an amount contained in the SAR and the result set into Z. |

3.3.5 Timing

For an ALU operation, the operand selection, function code, LIT field (used as an operand), and the result register selection code will come from the current Control Store Word. All of the above specified operations occur in the same control store cycle. All registers and memory contents used as operands are results of previous cycles. It should be noted that the Buffer Stores, because of their relatively slow speed (150-nsec cycle time) compared to Local Stores or other registers, are not available as direct operand inputs to the ALU. The Buffer Stores can be sources to the Local Stores, which is completed at the latter part of the control cycle; or, a Buffer Store can be loaded from Local Store or Z, which takes place over the first part of the control cycle.

Control Store branch conditions come from the results of ALU and Counter operations. Branch instructions are conditioned according to the results of the last cycle and specify the control word address for the next cycle.

See Fig. 4 for a timing chart of MCU operations.

3.4 Clock

The ALU cycle is capable of operating in 150 nsecs with Buffer Store and Control Store cycles of 150 nsec.

3.4.1 Control

The basic machine cycle is a 150-nsec cycle. Normal MCU operation can be described by three concurrent cycles: the ALU cycle, the Control Memory Cycle, and the Buffer Store cycle. The ALU cycle comprises data source selection and addressing, ALU operations, result register loading, and destination selection and addressing. The Control Store cycle operations comprise next-address generation, Control Store read, Control Register set, Control Register decode, and branch condition check. The Buffer Store cycle operations are Buffer Store Address set and Buffer Store read or write.

MCU cycle operations are controlled by a single 4-phase system clock. Each clock pulse of the four is derived from one basic clock pulse by a suitable delay and distributed to different sections of the machine as required.

The Sequence Unit (SU) and Maintenance Panel will control the dispatch of system clock pulses to provide for handling of interrupts, resolution of Buffer Store accessing conflicts, stops due to error conditions or CS address compare functions, and for control of single-pulse operations.

Hard stops will be accomplished by suspending clock pulses to the sequence unit and data flow logic. Single-cycle and single-phase operation from the maintenance console will turn on the clock control trigger for one cycle or one phase, after which they will return to hard stop condition. While in the running mode, a data storage conflict will cause the SU to suspend clock pulses to the data flow for one cycle or more until the conflict has been lifted.

3.4.2 Sequence Unit

The clock pulses are under the control of the maintenance panel and SU. Figure 4 shows the timing of Control Store address operations. Pulse P/3 initiates the first CSAR increment operation for the next CS address. Depending on the status of the selected address successor and test conditions, the next pulse P/1 initiates one of the following operations: another CSAR increment in case of a skip successor; Z register or LIT field to CSAR in case of a jump successor; ACSAR register to CSAR in case of a return jump; or no operation on CSAR in case of a step successor. P/1 also initiates a CSAR-to-ACSAR transfer in event of a Call jump where the return address must be saved. The CSAR can also be loaded with addresses from the ICU in the event of interrupt or interrupt return operations (see Section 6). Pulse P/3 loads the Command Register (CR) with the instruction to be executed.

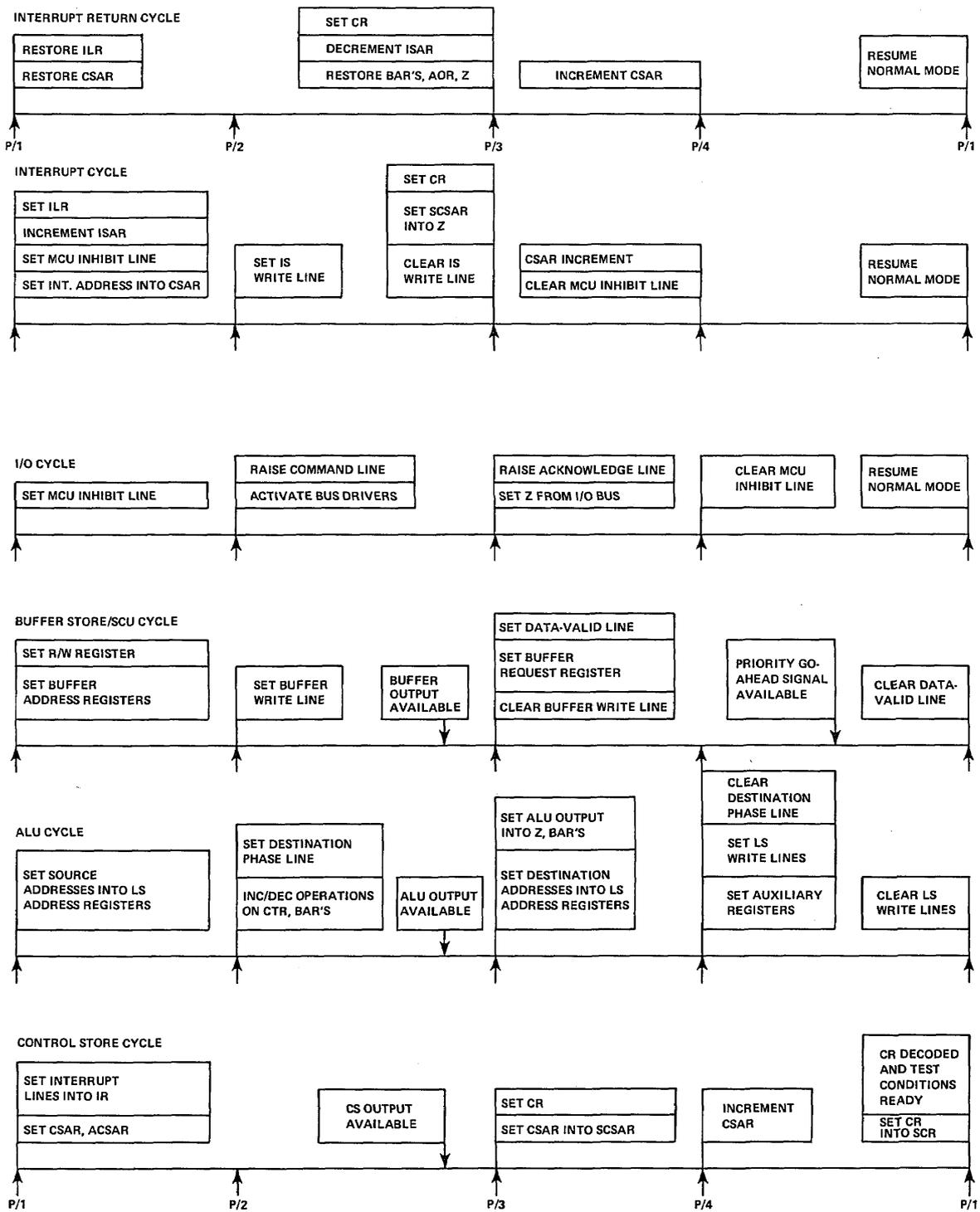


Fig. 4 - MCU cycle event timing

3.4.3 Data Flow

All four clock pulses are involved in the data flow. These control the data-source address register set, the Z-register set, the destination address register set, and the destination register set.

3.4.4 Hard Stop Conditions

Register contents will be as follows:

- a. CR - Instruction to be executed
- b. CSAR - Address plus one of instruction to be executed
- c. ACSAR - Contents from last cycle
- d. SCSAR - Address of instruction to be executed
- e. Local Stores - Results of last cycle
- f. LS and Buffer Address Registers - Results of last cycle
- g. CNTR and SAR - Results of last cycle
- h. Z - Results of last cycle.

3.4.5 Cycle Time

The cycle time is 150 nsecs.

3.5 Maintenance Panels

There are two maintenance panels (MP), one for the MCU (Fig. 5), and one for the SCU (Fig. 6). The MP's contain control switches and indicators for operation and maintenance. The SCU maintenance panel interface shall be pluggable and identical to a DMA channel.

3.5.1 MCU Switches

The maintenance panel can be used to modify a microprogrammed sequence, to stop the machine, and to initialize the machine. The control switches to perform these operations are listed below.

1. **START:** Starts the microprogram sequence after a machine stop.
2. **RUN, SINGLE CYCLE, SINGLE PULSE:** A key switch with three positions. In the **RUN** position, the MCU runs at normal speed. In the **SINGLE CYCLE** position, the MCU advances a cycle at a time (P/1→P/2→P/3→P/4). In the **SINGLE PULSE** position, the MCU advances a pulse at a time whenever the **SINGLE PULSE** switch is depressed.
3. **EXECUTE, SEQUENCE:** A key switch with two positions. In the **execute** position, the command register outputs are enabled and displayed. In the **sequence** position, the CS output bus is displayed and the Command Register load is inhibited.

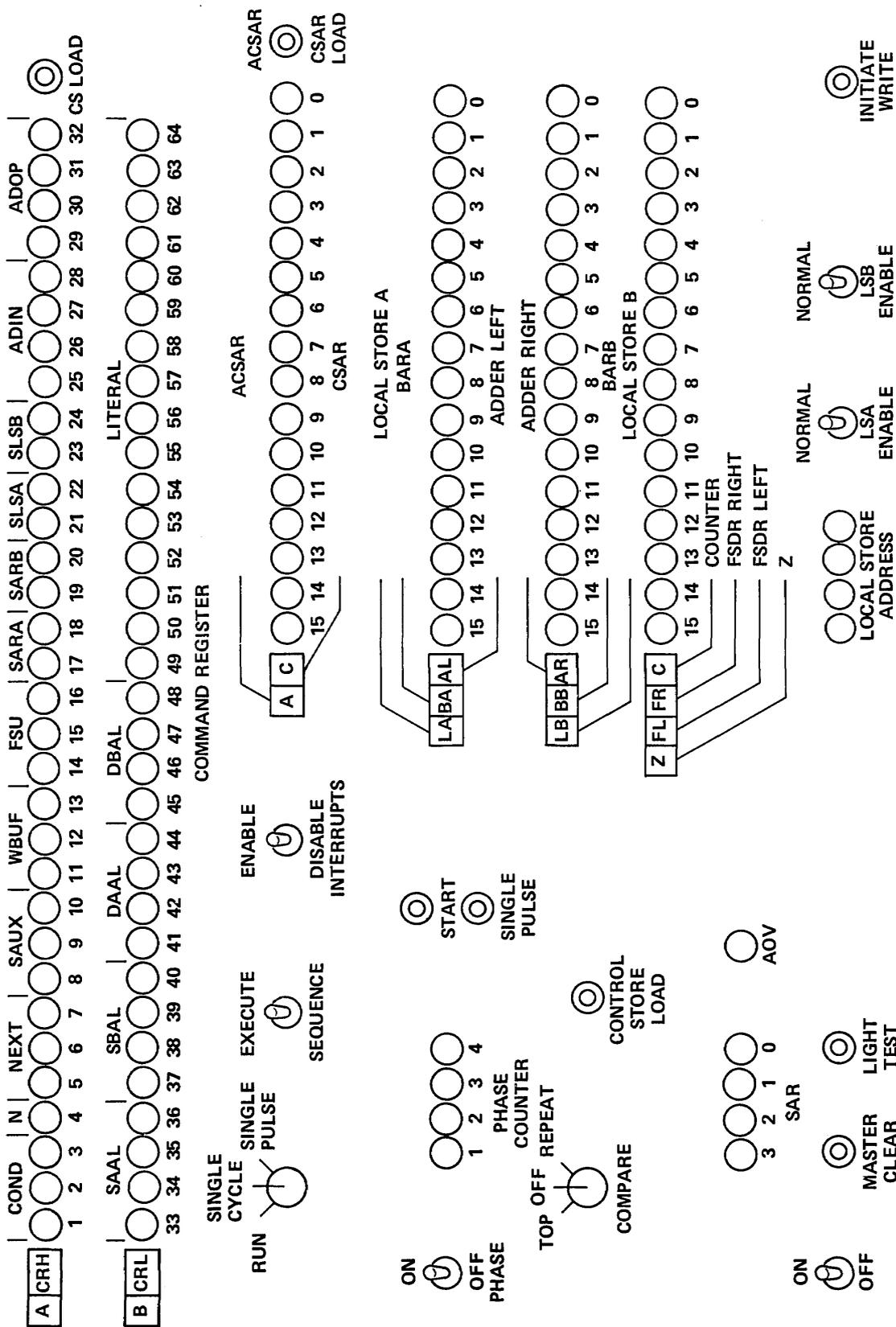


Fig. 5 - MCU maintenance panel

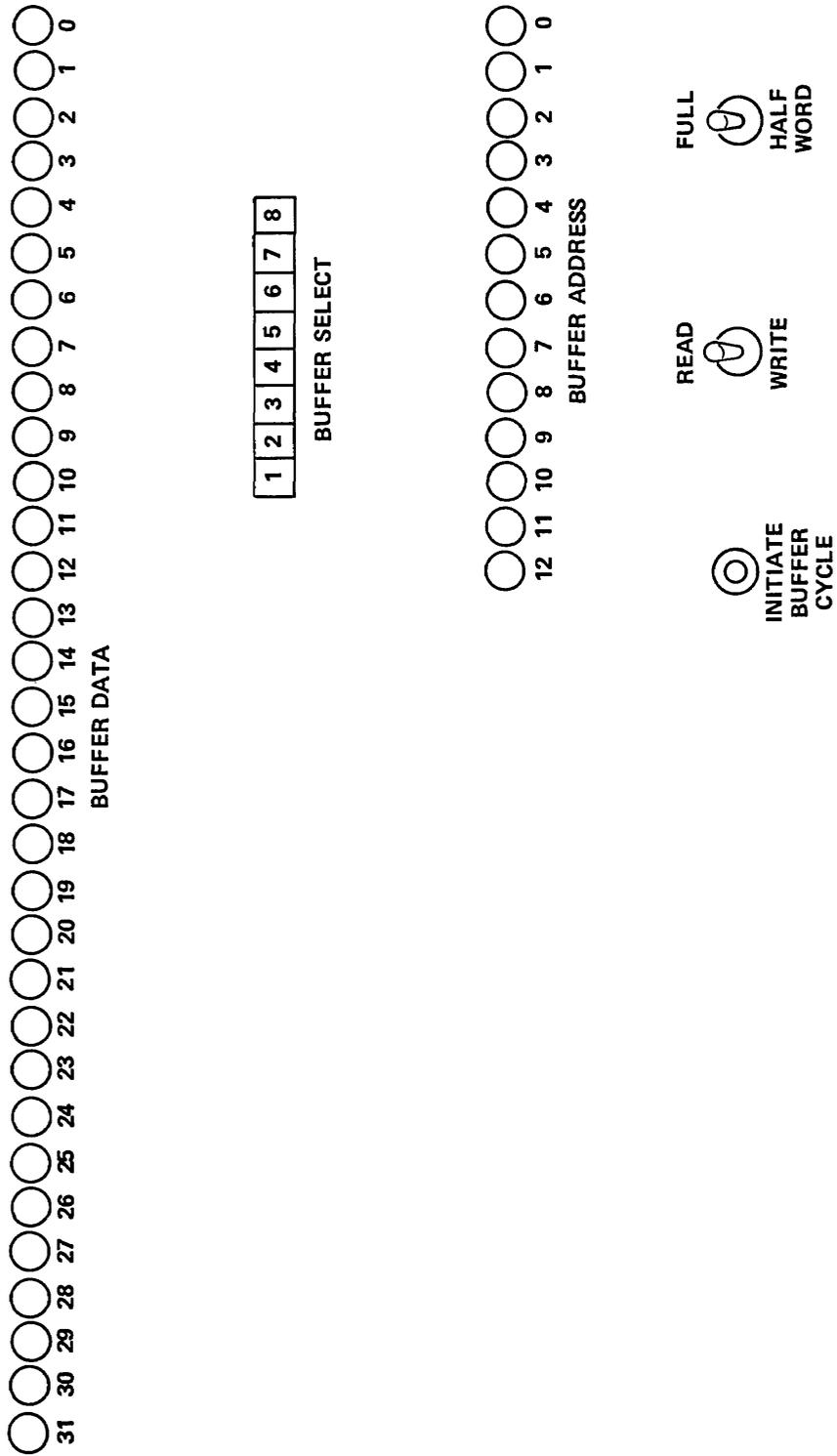


Fig. 6 - SCU maintenance panel

4. **DISABLE INTERRUPTS:** A two-position switch. In one position, the MCU operates normally. In the other position, all interrupts are disabled.

5. **PHASE ON, OFF:** A key switch with two positions — in the ON position, the phase counter is permitted to advance; in the OFF position, the advance of the phase counter is inhibited. Pulses are taken from the single-pulse switch.

6. **SINGLE PULSE:** A pushbutton that produces a single pulse at each depression.

7. **COMPARE:**

a. **CS STOP** causes the machine to stop when the CSAR matches the value set in the 12 low-order data keys.

b. **CS REPEAT** causes the machine to repeat the address set in the address keys.

8. **CS LOAD:** Initiates the control store load sequence. This switch can only be activated in the STOP state.

9. Six sets of pushbuttons are used to select for display registers as described under Section 3.5.3. They are

A. 1. A (BUS A)
2. CRH (High-Order Command Register)

B. 1. B (BUS B)
2. CRL (Low-Order Command Register)

C. 1. A (ACSAR)
2. C (CSAR)

D. 1. LA (LOCAL STORE A)
2. BA (BARA)
3. AL (ADDER LEFT SELECT)

E. 1. LB (LOCAL STORE B)
2. BB (BARB)
3. AR (ADDER RIGHT SELECT)

F. 1. Z
2. FL (FIELD SELECT DATA REGISTER - LEFT)
3. FR (FIELD SELECT DATA REGISTER - RIGHT)
4. C (COUNTER).

10. **ADDRESS KEYS:** 16 keys/indicators used to specify addresses entered into the MCU.

11. **DATA KEYS:** 64 keys/indicators used to specify data entered into the MCU.

12. **ACSAR, CSAR LOAD:** A pushbutton used to load the Alternate and Control Store Address Registers with the value in the address keys. The desired register is selected by depressing pushbuttons A and C. This switch can only be activated in the STOP state.

13. **CS LOAD KEY:** Causes the 64 data keys to be entered into the Control Store Word specified by the CSAR. These switches can only be activated in the Stop state.

14. **LS ADDRESS KEYS:** Four keys/indicators used to specify Local Store Addresses.

15. **LSA ENABLE:** A two-position switch used to select the LS ADDRESS KEYS as input to the LSA address register. This switch can only be activated in the Stop state.

16. **LSB ENABLE:** A two-position switch used to select the LS ADDRESS KEYS as input to the LSB address register. This switch can only be activated in the Stop state.

17. **INITIATE WRITE:** A pushbutton used to load the data specified in the low-order data keys into the Local Store word specified by the LS Address Keys. This switch is active only in the Stop state and if either LSA ENABLE or LSB ENABLE is active.

18. **MASTER CLEAR:** A pushbutton that, when depressed, clears the MCU to initial conditions.

19. **LAMP TEST:** A pushbutton switch which activates all display lights for test purposes.

20. **POWER ON/OFF:** Sequences power on or off.

3.5.2 SCU Switches

1. **BUFFER DATA KEYS:** 32 switches/indicators used to specify data to be entered into a buffer memory.

2. **BUFFER SELECT:** 8 pushbuttons used to select buffers. These buttons supply the high-order Buffer Address bits.

3. **BUFFER ADDRESS KEYS:** 13 switches/indicators used to specify the buffer memory low-order address bits.

4. **INITIATE BUFFER CYCLE:** A pushbutton which initiates buffer memory cycles.

5. **READ, WRITE KEY:** A two-position switch that determines whether retrieval or storage cycle is executed when INITIATE BUFFER CYCLE pushbutton is depressed.

6. **FULL, HALF-WORD KEY:** A two-position switch. In the FULL position 32 bits of information are transferred to or from Buffer Memory. In the HALF position, 16 bits of information are transferred to or from Buffer Memory. The high- or low-order 16 bits selected out of the 32-bit data word are determined by the state of the low-order address bit.

3.5.3 MCU Indicators

The MCU maintenance panel provides display capability in the MCU. The following are displayed.

1. The 64-bit Command Register (CR). The high-order 32 CR bits can be selected to display the Bus A outputs. The low-order 32 CR bits can be selected to display the Bus B outputs.

2. The 12-bit Alternate Control Store Address Register (ACSAR) or the Control Store Address Register (CSAR) can be selected for display in a set of indicators.

3. Buffer Address Register A (BARA) or the left adder input can be selected for display in a set of indicators.

4. Buffer Address Register B (BARB) or the right adder input can be selected for display in a set of indicators.

5. Register Z, Field Select Data Register (FSDR) Left, FSDR Right, and Counter (CTR), all 16 bits, share the same indicators. The desired register contents is displayed by depressing the respective pushbutton.

6. The four-bit Shift Amount Register (SAR) is displayed.

7. The 4-bit Phase Counter is displayed. The Phase Counter indicates which of the pulses P/1, P/2, P/3, or P/4 of the four-phase clock has occurred.

8. Adder Overflow (AOV) bit is displayed.

3.5.4 SCU Indicators

The SCU maintenance panel provides display capability. Displayed are the following:

1. 32 bits of Buffer Data are displayed.
2. 13 bits of Buffer Address are displayed.

4. STORAGE CONTROL UNIT INTERFACE

4.1 Inputs to SCU

Storage Request — This line is made active to request a single storage cycle, either fetch or store. It may be made active at any time and must remain active until the Priority Response line is received from the SCU.

Read/Write (R/W) — An active signal on this line along with a Storage Request indicates a request to store in a Buffer Store. An inactive signal indicates a fetch request. This line must remain stable from the beginning of the Storage Request until the Priority Response signal is received from the SCU and the pulse P/1 of the next cycle has occurred.

Address Lines — There are 16 address lines. Fifteen lines are used to specify the Buffer and word location. The 16th, low-order, line selects the low- or high-order 16 bits of the 32-bit addressed word. Timing considerations are the same as for the R/W signal.

Full/Half Word Control — The state of this line determines whether a full or half-word is being selected. Timing considerations are the same as that for the R/W signal.

Data Input — There are 32 two-way data lines. During input, they must remain stable from the beginning of the Storage Request until after the Buffer Store write signal is disabled (P/3 of the granted cycle).

4.2 Outputs from SCU

Priority Response — When active, it indicates to the receiving user that its storage request is being honored. The signal will be valid at the SCU interface 15 nsec before P/1 of the next cycle.

Data Valid — Used on fetch cycles only, this line indicates that the requested data are on the Data Out Lines. This line will rise at pulse P/3 and fall at pulse P/1 of the next cycle.

Data Output — There are 32 two-way data lines. During data output, they will be stable during the period the Data Valid line is active.

4.3 Performance

With the demand-response type of interface, the maximum data rate between the SCU and a channel controller depends upon the controller's proximity to the SCU. Maximum data rate is limited by the maximum speed of the storage.

4.4 Timing

SCU operations are timed by the four-phase system clock. Refer to Fig. 4 for the sequence of SCU operations.

5. INTERRUPT CONTROL UNIT

5.1 Function

The Interrupt Control Unit (ICU), Fig. 7, is responsible for monitoring interrupt lines, resolving interrupt priorities, and initiating MCU response to interrupts. The ICU has as inputs, 7/15 interrupt lines, clock lines from the MCU Sequence Unit, and the command register. The ICU has outputs to the Sequence Unit and the CSAR. The ICU is also connected to certain MCU registers for the purpose of saving their contents to allow proper resumption of preinterrupt activity.

5.2 ICU Storage Elements

1. A 7/15-bit Interrupt Register (IR) in which the status of the interrupt lines is captured every cycle during normal MCU operation.

2. An 8-word by 64-bit Interrupt Stack (IS) into which six registers are pushed and saved at each interrupt. The six registers are SCSAR (secondary CSAR), Z, BARA, BARB, ILR (interrupt level register), and AOR (adder overflow register).

The interrupt stack is pushed down automatically by an interrupt arrival and popped up by a jump return from interrupt. Popping up restores the saved registers to allow continuation of preinterrupt activity.

3. A 4-bit Interrupt Stack Address Register (ISAR) which holds the IS location of the MCU register contents saved at the last interrupt. The ISAR is an up/down counter. It is incremented once at each interrupt before the MCU registers are saved into the IS and decremented once at each jump return from interrupt after the MCU registers are restored with the contents of the IS word.

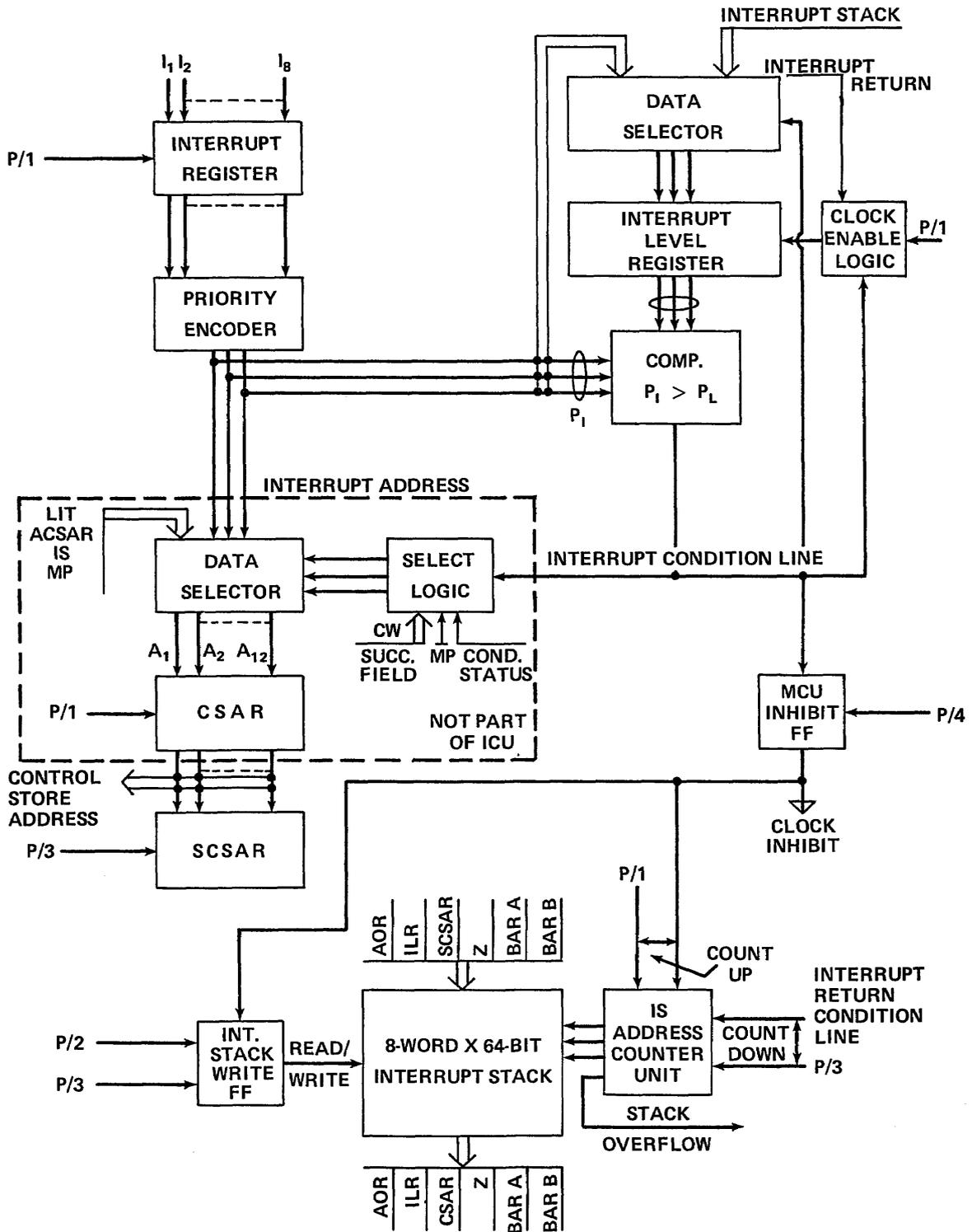


Fig. 7 - Interrupt Control Unit (ICU)

4. A 3-bit Interrupt Level Register (ILR) which holds the level that an incoming interrupt must exceed in order to be recognized by the ICU. The ILR is automatically raised to the level of an interrupt which is accepted by the ICU.

5. Secondary Control Store Address Register (SCSAR), which is not to be confused with the ACSAR. This register is loaded, every cycle, with the contents of the CSAR. This is done at P/3 which is after the CSAR contents are formed for the next cycle but before the CSAR is incremented at P/4 for the cycle after that. Therefore, SCSAR always contains, at P/1, the address of the CS word about to be executed.

5.3 ICU Operation

Any device wishing the attention of the MCU can raise its interrupt line at any time. At P/1, the ICU captures the state of all interrupt lines in the Interrupt Register. Priority-resolving logic then generates an interrupt address from the highest priority interrupt present whose level is greater than that contained in the Interrupt Level Register. If there is an allowable interrupt, an Interrupt Condition Line is raised prior to P/4. This line enables the MCU inhibit flipflop to be set at P/4. The MCU inhibit flipflop output inhibits normal MCU operations. At P/4 of the interrupt cycle the MCU flipflop is cleared.

The MCU interrupt cycle is as follows. At P/1, the CSAR is loaded with the interrupt address generated by the ICU. This points the control store sequence to the first address of the interrupt routine which is to service the interrupt. Also at P/1, the ISAR is incremented by one to prepare the IS to accept the MCU registers which are to be saved, and the ILR is loaded with the level of the new interrupt. At P/2, the IS write line is set, and the SCSAR, Z, BARA, BARB, ILR, and AOR register contents are loaded into the IS location specified by the ISAR. At P/3, the IS write line is cleared. P/3 also sets the contents of SCSAR into Z. This leaves in Z the CS address about to be executed when the interrupt occurred. At P/4, the MCU inhibit flipflop is cleared. By P/4, the first word of the interrupt routine has been read from the Control Store and has been loaded in the Control Register, and the Interrupt Condition Line has dropped, as the new contents of the ILR inhibit acceptance of the interrupts currently in the IR. This allows the MCU to proceed with normal interrupt-routine processing and allows acceptance of subsequent interrupts, subject to the new level set in the ILR.

5.4 Interrupt Return

While the MCU is processing the interrupt routine, the interrupting device can be directed to drop its interrupt line through the use of an I/O command word. In order for the MCU to return to its preinterrupt processing condition, an Interrupt Return Jump is included in the Control Store Word successor field. During the interrupt return cycle, the Interrupt Return Jump causes the CSAR, Z, BARA, BARB, ILR, and AOR registers to be restored from the IS location addressed by the ISAR. Note that no other operations can be specified in the same cycle that an Interrupt Return Jump is specified in order that there will be no conflict with the operations required for restoring registers. At P/3, the ISAR is decremented by one to prepare the IS for the possible occurrence of a later Interrupt Return Jump.

By P/3, the Control Store Word at the return-address location is available and is clocked into the Control Register for the resumption of processing at P/1.

6. I/O SYSTEM ELEMENTS

6.1 Organization

There are two types of input/output channels in the SPE: Direct Memory Access (DMA) buffered channels and a single unbuffered channel called the "Z bus." Eight/sixteen buffered channels enable high-speed data transfer between buffer memories and system devices or MCU's. The Z bus allows direct communication under MCU control and on a word-by-word basis between the Z register of an MCU and all devices connected to the Z bus. The Z bus also enables direct MCU-to-MCU communication.

Figure 8 shows an SPE configuration with I/O system elements and interconnections.

All buffered channel communications pass through the Storage Control Unit (SCU). It is the responsibility of the SCU to manage buffer memory requests originating from MCU's, SPAU's, and system channel controllers. Devices which access buffer memory over buffered channels are interfaced to the buffered channels by Selector Channel Controllers (SCC). SCC's also interface with the Z bus and are responsible for interpreting device requests coming over the Z bus from MCU's. These requests originate in the form of MCU I/O instructions and can call upon an SCC to initiate various device I/O operations over its buffered channel interface.

The SCC's are intended to be standard I/O elements interfacing between buffered channels and Device Controllers (DC). DC's interface between SCC's and I/O devices and must be tailored to meet the interface requirements of a particular device type. DC's interface to SCC's over Z-bus-compatible connections. This allows a DC to connect directly to the Z bus for direct unbuffered communication with an MCU or to connect to an SCC for buffered channel communication.

SCC's and DC's can request MCU action via interrupt lines provided in the MCU's for such purposes. Separate SCC's or DC's sharing a single interrupt line must have hardware to resolve competition among the units for interrupt service.

An MCU generating an I/O request addressed to another MCU for the purpose of MCU-to-MCU communication causes the addressed MCU to raise an internal interrupt line. An I/O acknowledge instruction by the interrupted MCU completes the data transfer over the Z bus.

6.2 Unbuffered Channel

The MCU exchanges commands and unbuffered data with devices over a bidirectional, byte-multiplexed channel called the "Z bus." The channel can be interrupt driven with information transferred from/to the Z register upon execution of an I/O instruction. There is but one Z bus regardless of the number of MCU's a system may contain. MCU's are in charge of Z-bus usage and resolve Z-bus access among themselves on a first-come, first-served basis.

Devices wishing Z-bus access must alert the MCU via an interrupt. A single interrupt line may serve many devices via a Device Controller which controller must resolve simultaneous requests for service. The interrupt line is deactivated upon transmission of an I/O command by the servicing MCU to the interrupting device's controller. Different interrupt lines servicing different sets of devices are subject to the priority resolution scheme built into the MCU interrupt mechanism, priority being determined by the position of the interrupt line on the MCU. The maximum burst transfer rate over the Z bus, based upon an MCU cycle time of 150 nsec, is 2 MHz.

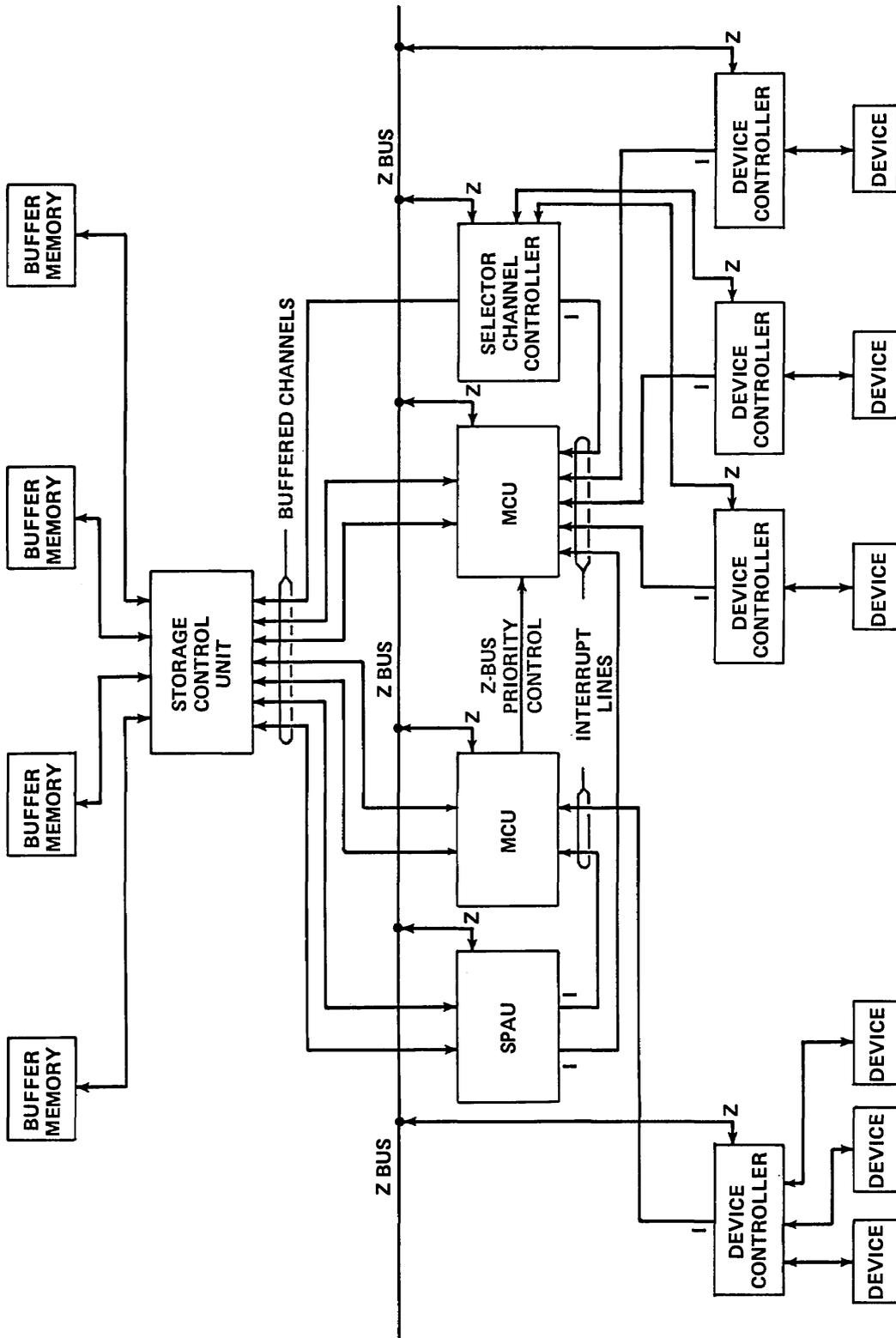


Fig. 8 - SPE configuration illustrating the I/O system

6.2.1 Z Bus

The Z bus consists of 30 lines, 16 of which are bidirectional data lines. The remaining 14 are control lines, as follows:

- a. Eight Device Address Lines (bidirectional)
- b. Input/Output Line (output)
- c. Data/Control Line (output)
- d. Continue Line (output)
- e. Command Line (bidirectional)
- f. Acknowledge Line (bidirectional)
- g. Reject Line (input).

The device address lines alert a specified device controller or channel controller to connect to the Z-bus data lines for an information transfer.

The Input/Output Line specifies the direction of information transfer.

The Data/Control Line specifies whether the transfer will involve data or control information. If control information is specified, an input command will bring a status word from the device to the MCU; an output command will send a command word from the MCU to the device.

The Continue Line is used to specify, where appropriate, that a device be prepared to accept another I/O word as continuation of the present I/O.

The Command Line alerts all devices on the Z bus that an I/O command has been put on the bus.

The Acknowledge Line is raised by a device to acknowledge to the MCU that the device has accepted the I/O command.

The Reject Line is raised by a device to notify the MCU that the device cannot accept the I/O command.

The Device Address Lines, Command Line, and Acknowledge Line take part in MCU-to-MCU communication as well as MCU-to-device communication and are therefore bidirectional. An MCU continually monitors the Command Line and Device Address Lines for an I/O command directed at it by another MCU. An MCU can raise the Acknowledge Line in response to an I/O command in the same manner as a device.

A separate line called the Z-Bus Priority Control Line passes from one MCU to another in a multiple-MCU system. This line allows each MCU, if it does not wish the Z bus, to pass on the bus-access privilege to the next MCU in line in decreasing priority.

6.2.2 MCU-to-Z-Bus Interface

The MCU interfaces with the Z bus, Fig. 9, through its Z register and Local Store A (LSA). The Z-bus data lines input to and are driven from the Z register. The Z register must be loaded with data or command information prior to an I/O output command. An I/O input operation loads the Z register from the Z bus for subsequent use.

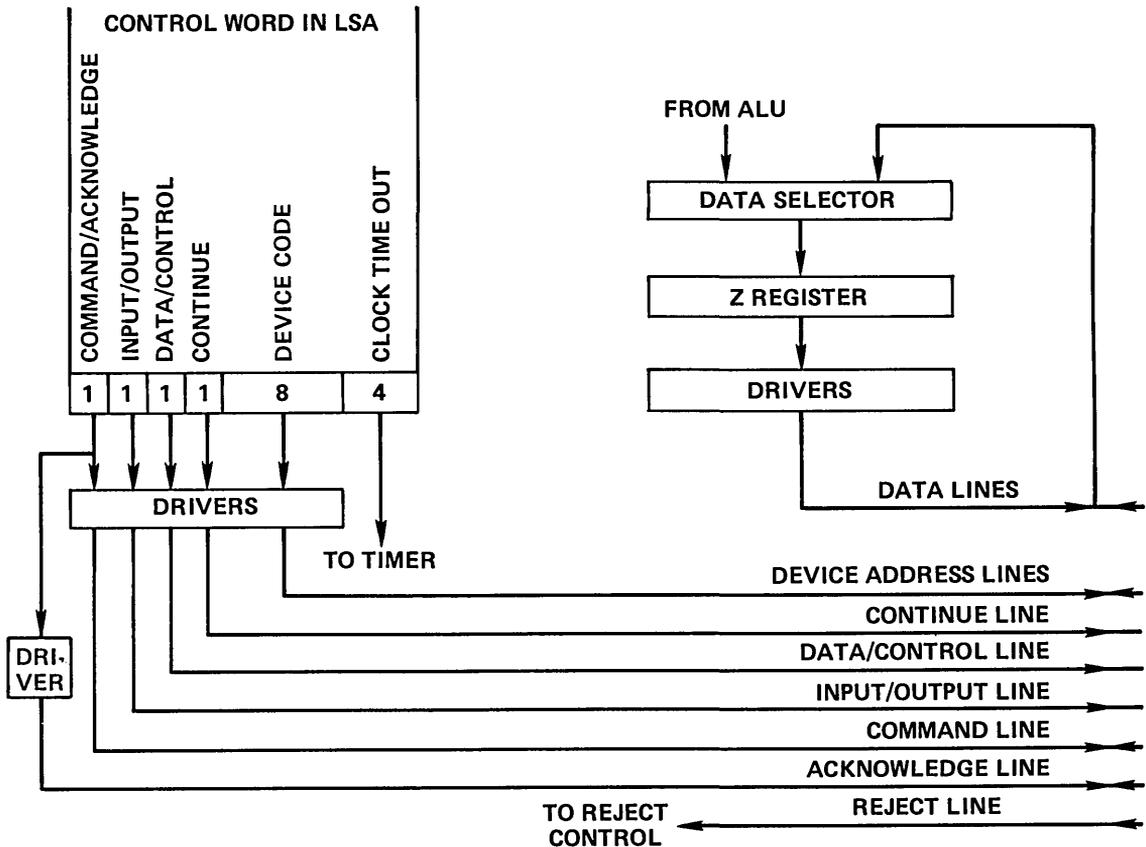


Fig. 9 - MCU-to-Z-bus interface

A control word must be set up in LSA prior to the I/O command. The LSA location containing the control word is specified by the LSA Address Field, and 12 bits of the selected word directly drive 12 control lines of the Z bus as shown in Fig. 9. The Acknowledge Line is driven indirectly through I/O control logic. The Reject Line inputs to the MCU Sequence Unit and causes a program jump to the Control Store location specified by the LIT field contents in the I/O command word.

Four bits of the command word set up in LSA do not go out over the Z bus but go to a clock time-out counter in the MCU. These bits allow the program to specify a time interval in powers of 2 from 2^0 to 2^{15} clock cycles. This interval limits the time that the MCU will wait inhibited in the I/O state for the return of an Acknowledge signal from an addressed device. In the event of a time out, an internal interrupt is generated and the MCU leaves the I/O state to process that interrupt.

6.2.3 Unbuffered I/O Operation

I/O operations are initiated by an I/O instruction in an MCU. Devices cannot initiate I/O operations directly but do so by appealing to an MCU through the interrupt system. An I/O command is specified by a code point of 7 in the command word successor field. Prior to the I/O instruction, the I/O control word must be set up in LSA and, if the transfer is to be an output, data or command information must be set up in the Z register.

Also, the programmer must specify in the LSA address field the location of the I/O control word in LSA, and the address of the Reject jump must be specified in the LIT field.

The MCU Command Register is loaded every cycle at P/3. If an I/O operation is specified, the MCU raises an I/O Request Line to the Z-Bus Control Unit which resolves requests for the Z bus from different MCU's. Access to the bus is granted strictly on a first-come, first-served basis. If the Z bus is available, the MCU will receive a Z-Bus Available Line which, combined with the I/O Request Line, initiates the I/O cycle at P/1. Absence of the Z-Bus Available signal inhibits MCU operation until the bus becomes available at a later cycle.

The MCU I/O cycle proceeds as follows. P/2 sets the I/O Command FF or P/3 the I/O Acknowledge FF, depending on the state of the Command/Acknowledge bit in the I/O control word read from LSA. An I/O command causes the control word drivers from LSA to be energized, and, if the command is an output, the data drivers from the Z register are also energized. Further clock pulses are suspended to MCU operation until the receipt of the Acknowledge Line signal in conjunction with P/3 clears the I/O Command FF, thereby deenergizing all Z-bus drivers. Simultaneously, if the I/O command is an input, the information on the Z-bus data lines is clocked into the Z register. The MCU then proceeds with normal operation.

If the I/O operation is an Acknowledge, the I/O Acknowledge FF is set at P/3. This energizes a driver on the Acknowledge Line only. Simultaneously, the Z-bus data lines are clocked into the Z register and the MCU continues with normal operation.

If the Reject Line is raised in response to an I/O Command, the MCU remains inhibited until the conjunction of P/1 and the Reject Line signal which clears the I/O Command FF and clocks the contents of the LIT field into the CSAR. This causes the MCU to resume normal processing at the I/O Reject jump address.

6.3 Buffered Channel Controller

The Buffered Channel Controller can take several forms. In the SPE system a Selector Channel Controller is specified as a generalized Direct Memory Access Controller.

The SCC interfaces with the Storage Control Unit (SCU), the Z bus, and up to 8 Device Controllers (DC). Each DC may be attached to up to 4 SCC's.

The SCC interfaces with the Z bus to receive commands from and to transmit status information to an MCU. The SCC, in addition to the Z-bus lines, has an interrupt line to alert an MCU of a change in device status.

The SCC-to-DC interface is identical to the Z-bus interface except that the data width is 32 lines. Eight, 16, or 32 data-line DC's may be connected to the SCC. The particular width for an SCC-DC operation is determined by a 2-bit field (WDC) in the MCU I/O Command to the SCC.

The SCC contains a 32-bit assembly register at the SCC-DC interface and a 32-bit data exchange register at the SCU-SCC interface. Two incrementing address registers and a word counter are contained in the SCC.

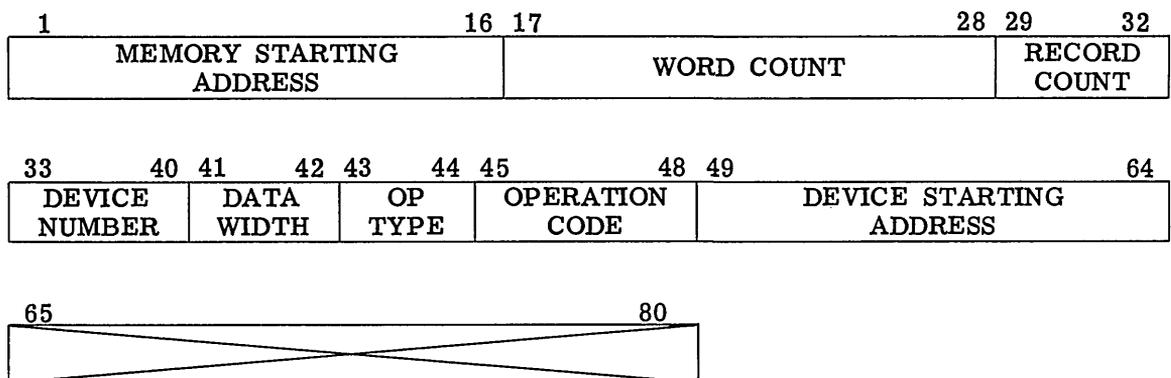
If an SCC is connected to the highest priority SCU position, a transfer rate to buffer memory of 192 Mbits/sec is possible. The maximum transfer rate is dependent on the round-trip line delay between the SCC and DC.

6.3.1 SCC Command and Format

The SCC command format includes the following fields:

- a. Buffer Memory Starting Address (15 bits)
- b. Word Count (12 bits)
- c. Record Count (4 bits)
- d. Device Number (8 bits)
- e. SCC-DC Data Width (2 bits)
- f. Operation Type (2 bits)
- g. Operation Code (4 bits)
- h. Device Starting Address.

SCC COMMAND FORMAT



6.3.2 SCC Status Information and Format

The SCC status field includes the following:

- a. Next Memory Address (15 bits)
- b. Word Count (12 bits)
- c. Record Count (4 bits)
- d. Device Number (8 bits)
- e. SCC-DC Data Width (2 bits)
- f. Operation Type (2 bits)
- g. Operation Code (4 bits)
- h. Next Device Address (16 bits)
- i. Status (16 bits).

SCC STATUS FORMAT

| | | | | | | | | | | | | | | | | | | | | | |
|---------------------|-------|--|------------|--|---------|--|----------------|--|---------------------|--|----|--|--|--|--|------------|-------|--|----|--------------|--|
| 1 | 16 17 | | | | | | | | | | | | | | | | 28 29 | | 32 | | |
| NEXT MEMORY ADDRESS | | | | | | | | | | | | | | | | WORD COUNT | | | | RECORD COUNT | |
| 33 | | | 40 41 | | 42 43 | | 44 45 | | 48 49 | | 64 | | | | | | | | | | |
| DEVICE NUMBER | | | DATA WIDTH | | OP TYPE | | OPERATION CODE | | NEXT DEVICE ADDRESS | | | | | | | | | | | | |
| 65 | | | | | | | | | | | | | | | | 80 | | | | | |
| STATUS | | | | | | | | | | | | | | | | | | | | | |

7. CONTROL STORE LOAD

The control store will be loaded from Bulk Store via an extension of the Direct Memory Access (DMA) channel which connects the Bulk Store to the Storage Control Unit (SCU). The SCU interface is idle during the transfer.

The control store load is initiated by an MCU I/O instruction which transmits the page number and initial control store address to the Bulk Store. If the Bulk Store accepts the command, the MCU remains inhibited while the desired page is written into the control store under control of the Bulk Store. When the transfer is complete, the Bulk Store raises an acknowledge signal to the MCU which results in the MCU continuing to the next instruction in sequence.

The operation is timed by a counter in the MCU, in the same manner described in Section 6, to prevent lockup caused by equipment failures.

8. MCU CONTROL FIELD SUMMARY

| Field | CR Bits | Functions |
|-------|---------|---|
| COND | 1-3 | Condition test select |
| NOT | 4 | True or false of condition |
| NEXT | 5-7 | Next control store address selection |
| SAUX | 8-10 | Set auxiliary register |
| WBUF | 19-21 | Buffer write operation |
| FSU | 22-24 | Definition field for the Field Select Unit |
| SARA | 11-12 | Source for Buffer Address Register A (BARA) |
| SARB | 13-14 | Source for Buffer Address Register B (BARB) |
| SLSA | 15-16 | Source for Local Store A input |
| SLSB | 17-18 | Source for Local Store B input |
| ADIN | 25-28 | Adder input selection |
| ADOP | 29-32 | Adder/shifter operation selection |

| <u>Field</u> | <u>CR Bits</u> | <u>Functions</u> |
|--------------|----------------|-------------------------------------|
| SAAL | 33-36 | Source Address Literal for LSA |
| SBAL | 37-40 | Source Address Literal for LSB |
| DAAL | 41-44 | Destination Address Literal for LSA |
| DBAL | 45-48 | Destination Address Literal for LSB |
| LIT | 49-64 | Literal (all purpose-data, address) |

9. CONTROL FIELD DEFINITIONS

9.1 CØND Field

The CØND field is a 3-bit field defining an MCU hardware condition which controls the operation of the control store address successor specified in the NEXT field.

| <u>Code Point</u> | <u>Symbol</u> | <u>Definition</u> |
|-------------------|---------------|---|
| 0 | — | No condition |
| 1 | MST | Most significant bit of ALU result true |
| 2 | LST | Least significant bit of ALU result true |
| 3 | AØV | Adder overflow |
| 4 | CØV | Counter Register overflow* |
| 5 | ZERO | ALU result = 0 |
| 6 | — | Spare |
| 7 | I/O | Indicates that the rest of the control word is to be treated by the MCU as an I/O instruction |

9.2 NØT Field

The NØT field is a 1-bit field indicating whether the true or false value of the condition specified in the CØND field is to control the control store successor operation.

| <u>Code Point</u> | <u>Symbol</u> | <u>Definition</u> |
|-------------------|---------------|-------------------|
| 0 | T | Specifies CØND |
| 1 | F | Specifies CØND |

*The counter register is incremented by one each time its condition is checked by the CØND field.

9.3 NEXT Field

The NEXT field is a 3-bit field specifying the next control store address to be formed if the condition specified by the CØND and NØT fields is satisfied.

| <u>Code Point</u> | <u>Symbol</u> | <u>Next CSAR</u> | <u>Next ACSAR</u> |
|-------------------|---------------|------------------|-------------------|
| 1 | SKIP | CSAR+2 | NC |
| 2 | SAVE | CSAR+1 | CSAR+1 |
| 3 | CALL | LIT | CSAR+1 |
| 4 | JUMPL | LIT | NC |
| 5 | JUMPA | ACSAR | NC |
| 6 | JUMPZ | Z | NC |
| 7 | INTRET | Interrupt Stack* | NC |

If the condition specified is not satisfied, the default successor is a Step.

9.4 SAUX Field

The SAUX field is a 3-bit field which specifies the 16-bit LIT field or Z register as source for a transfer into an auxiliary register: Counter (CNTR), Shift Amount Register (SAR), Alternate Control Store Address Register (ACSAR).

| <u>Code Point</u> | <u>Transfer</u> | <u>Code Point</u> | <u>Transfer</u> |
|-------------------|-----------------|-------------------|-----------------|
| 0 | None | 4 | Z to SAR |
| 1 | LIT to ACSAR | 5 | LIT to CTR |
| 2 | Z to ACSAR | 6 | Z to CTR |
| 3 | LIT to SAR | 7 | Spare |

9.5 WBUF Field

The WBUF (Write Buffer) field is a 3-bit field which designates the MCU source to be transferred to Buffer Storage Modules specified by BARA or BARB.

| <u>Code Point</u> | <u>Transfer</u> |
|-------------------|------------------------------|
| 0 | No Write |
| 1 | BUF(BARA)←LSA |
| 2 | BUF(BARB)←LSA |
| 3 | BUF(BARA)←LSB |
| 4 | BUF(BARB)←LSB |
| 5 | BUF(BARA)←Z |
| 6 | BUF(BARB)←Z |
| 7 | BUF(BARA)←LSA, BUF(BARB)←LSB |

*See Section 4.

9.6 FSU Field

The FSU (Field Select Unit) field is a 3-bit control word field used as a data field selector. A 0 code point causes a transfer over Bus A of a Buffer word into the FSU Data Register (FSDR). Any other FSU code point will cause a masked select and right shift of the selected field in the FSDR to be made available, right justified, at the FSU output.

| <u>Code Point</u> | <u>Definition</u> | <u>Code Point</u> | <u>Definition</u> |
|-------------------|-------------------|-------------------|-------------------|
| 0 | FSDR-BUF(BARA) | 4 | FSDR Field 4 |
| 1 | FSDR Field 1 | 5 | FSDR Field 5 |
| 2 | FSDR Field 2 | 6 | FSDR Field 6 |
| 3 | FSDR Field 3 | 7 | FSDR Field 7 |

9.7 SARA Field

The SARA (Source for buffer Address Register A (BARA)) field is a 2-bit field specifying the source for BARA.

| <u>Code Point</u> | <u>Source</u> |
|-------------------|-------------------------|
| 0 | None |
| 1 | -1, decrement BARA by 1 |
| 2 | +1, increment BARA by 1 |
| 3 | Z |

9.8 SARB Field

The SARB (Source for buffer Address Register B (BARB)) field is a 2-bit field specifying the source for BARB.

| <u>Code Point</u> | <u>Source</u> |
|-------------------|-------------------------|
| 0 | None |
| 1 | -1, decrement BARB by 1 |
| 2 | +1, increment BARB by 1 |
| 3 | Z |

9.9 SLSA Field

The SLSA (Source for Local Store A) field is a 2-bit field specifying the source for LSA input.

| <u>Code Point</u> | <u>Source</u> |
|-------------------|---------------|
| 0 | None |
| 1 | BUF(BARA) |
| 2 | BUF(BARB) |
| 3 | Z |

9.10 SLSB Field

The SLSB (Source for Local Sore B) field is a 2-bit field specifying the source for LSB input.

| <u>Code Point</u> | <u>Source</u> |
|-------------------|---------------|
| 0 | None |
| 1 | BUF(BARA) |
| 2 | BUF(BARB) |
| 3 | Z |

9.11 ADIN Field

The ADIN field is a 4-bit field which specifies the left (L) and right (R) inputs to the ALU.

| <u>Code Point</u> | <u>L Input</u> | <u>R Input</u> |
|-------------------|----------------|----------------|
| 0 | LSA | LIT |
| 1 | LSA | FSU |
| 2 | LSA | BARB |
| 3 | LSA | LSB |
| 4 | LSB | LIT |
| 5 | LSB | FSU |
| 6 | LSB | BARB |
| 7 | LSB | BARA |
| 8 | BARA | LIT |
| 9 | BARA | LSA |
| 10 | BARB | LIT |
| 11 | BARB | BARA |
| 12 | FSU | LIT |
| 13 | CNTR | LIT |
| 14 | ACSAR | LIT |
| 15 | SAR | LIT |

9.12 ADØP Field

ADØP is a 4-bit field which specifies the operation that the ALU is to perform on the L and R inputs. The ALU result goes into the Z register.

| <u>Code Point</u> | <u>Symbol</u> | <u>Definition</u> |
|-------------------|-------------------------|---|
| 0 | NOP | No operation (Z remains unchanged) |
| 1 | $L+R\rightarrow Z$ | L plus R |
| 2 | $L-R\rightarrow Z$ | L minus R |
| 3 | $R-L\rightarrow Z$ | R minus L |
| 4 | $L+1\rightarrow Z$ | L plus 1 |
| 5 | $L-1\rightarrow Z$ | L minus 1 |
| 6 | $L\rightarrow Z$ | L |
| 7 | $\bar{L}\rightarrow Z$ | L complement |
| 8 | $R\rightarrow Z$ | R |
| 9 | L OR R $\rightarrow Z$ | Bit logical OR of L and R |
| 10 | L AND R $\rightarrow Z$ | Bit logical AND of L and R |
| 11 | L XOR R $\rightarrow Z$ | Bit logical EXCLUSIVE-OR of L and R |
| 12 | L EQU R $\rightarrow Z$ | Bit logical EQUIVALENCE of L and R |
| 13 | LEFT L $\rightarrow Z$ | Left shift L, zero fill (shift controlled by SAR) |
| 14 | RIGHT L $\rightarrow Z$ | Right shift L, zero fill |
| 15 | CIRC L $\rightarrow Z$ | Left circular shift L |

9.13 SAAL Field

The SAAL field is a 4-bit field specifying the address of the register in Local Store A which is to be a data source.

| <u>Code Point</u> | <u>Definition</u> |
|-------------------|--------------------------------|
| 0 | LSA address = contents of BARA |
| 1 | LSA address = 1 |
| 2 | LSA address = 2 |
| . | . |
| . | . |
| . | . |
| 15 | LSA address = 15 |

9.14 SBAL Field

The SBAL field is a 4-bit field specifying the address of the register in Local Store B which is to be a data source.

| <u>Code Point</u> | <u>Definition</u> |
|-------------------|--------------------------------|
| 0 | LSB address = contents of BARB |
| 1 | LSB address = 1 |
| 2 | LSB address = 2 |
| . | . |
| . | . |
| 15 | LSB address = 15 |

9.15 DAAL Field

The DAAL field is a 4-bit field specifying the address of the register in Local Store A which is to be a data destination.

| <u>Code Point</u> | <u>Definition</u> |
|-------------------|--------------------------------|
| 0 | LSA address = contents of BARA |
| 1 | LSA address = 1 |
| 2 | LSA address = 2 |
| . | . |
| . | . |
| . | . |
| 15 | LSA address = 15 |

9.16 DBAL Field

The DBAL field is a 4-bit field which specifies the address of the register in Local Store B which is to be a data destination.

| <u>Code Point</u> | <u>Definition</u> |
|-------------------|--------------------------------|
| 0 | LSA address = contents of BARB |
| 1 | LSA address = 1 |
| 2 | LSA address = 2 |
| . | . |
| . | . |
| . | . |
| 15 | LSA address = 15 |

9.17 LIT Field

The LIT field is a 16-bit field serving as a literal source of data and/or addresses to various devices in the MCU.

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

| | | | |
|---|--|--|-----------------|
| 1. ORIGINATING ACTIVITY (Corporate author) Naval Research Laboratory Washington, D.C. 20390 | | 2a. REPORT SECURITY CLASSIFICATION Unclassified | |
| | | 2b. GROUP | |
| 3. REPORT TITLE SIGNAL PROCESSING ELEMENT FUNCTIONAL DESCRIPTION: PART 1 - MICRO-PROGRAMMED CONTROL UNIT, BUFFER STORE, AND STORAGE CONTROL UNIT | | | |
| 4. DESCRIPTIVE NOTES (Type of report and inclusive dates) An interim report on continuing NRL Problems. | | | |
| 5. AUTHOR(S) (First name, middle initial, last name) John P. Ihnat, William R. Smith, John D. Roberts, Jr., Y.S. Wu, and Bruce Wald | | | |
| 6. REPORT DATE September 12, 1972 | | 7a. TOTAL NO. OF PAGES 39 | 7b. NO. OF REFS |
| 8a. CONTRACT OR GRANT NO. NRL Problems B02-06 and B02-10 | | 9a. ORIGINATOR'S REPORT NUMBER(S) NRL Report 7490 | |
| b. PROJECT NO. 3F21-241-601 (formerly WF-15-241-601) | | | |
| c. 2F00-241-601, XF-53-241-003, and | | 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) | |
| d. XF-21-241-015-K152 | | | |
| 10. DISTRIBUTION STATEMENT Approved for public release; distribution unlimited. | | | |
| 11. SUPPLEMENTARY NOTES | | 12. SPONSORING MILITARY ACTIVITY Department of the Navy (Naval Air Systems Command and Naval Command Control, Communications and Electronics Command), Washington, D.C. 20360 | |
| 13. ABSTRACT The NRL Signal Processing Element (SPE) is being developed to provide a high-performance signal processing facility for radar, sonar, and communication systems. It is intended to be compatible with the Navy's All Applications Digital Computer (AADC). The SPE consists of four major subsystems: a Microprogrammed Control Unit (MCU), a Buffer Store and Storage Control Unit (SCU), a Signal Processing Arithmetic Unit (SPAU), and Input/Output (I/O) units. The MCU serves as system supervisor and data organizer for the SPAU and other I/O devices. The MCU includes a 64-bit Control Store, two local stores, an arithmetic element, two busses to buffer memory, an unbuffered byte channel, and a priority interrupt system. Its cycle time is 150 nanoseconds (nsec). The buffer store and SCU consists of eight fixed-priority Direct Memory Access (DMA), or Buffered, Channels communicating on a 150-nsec cycle basis with up to eight 32-bit-by-4K memories. The SPAU is a highly parallel, high-speed arithmetic unit capable of performing complex signal processing operations such as FFT and recursive filtering. It is microprogrammed for flexibility in adapting signal processing algorithms. Four multiplies and six additions can be performed in 300 nsec. SPE I/O and internal communications are provided by DMA buffer data channels, an unbuffered byte channel, and a priority interrupt system. The unbuffered byte channel called the Z Bus communicates both data and control information to all I/O devices. The unbuffered channel burst data rate is 2 million 16-bit words per second. | | | |

| 14. KEY WORDS | LINK A | | LINK B | | LINK C | |
|---|--------|----|--------|----|--------|----|
| | ROLE | WT | ROLE | WT | ROLE | WT |
| Signal Processing Element Signal Processing Microprogram Control Unit Microprogramming Storage Control Unit Buffer Store Signal Processing Arithmetic Unit Digital Computers | | | | | | |