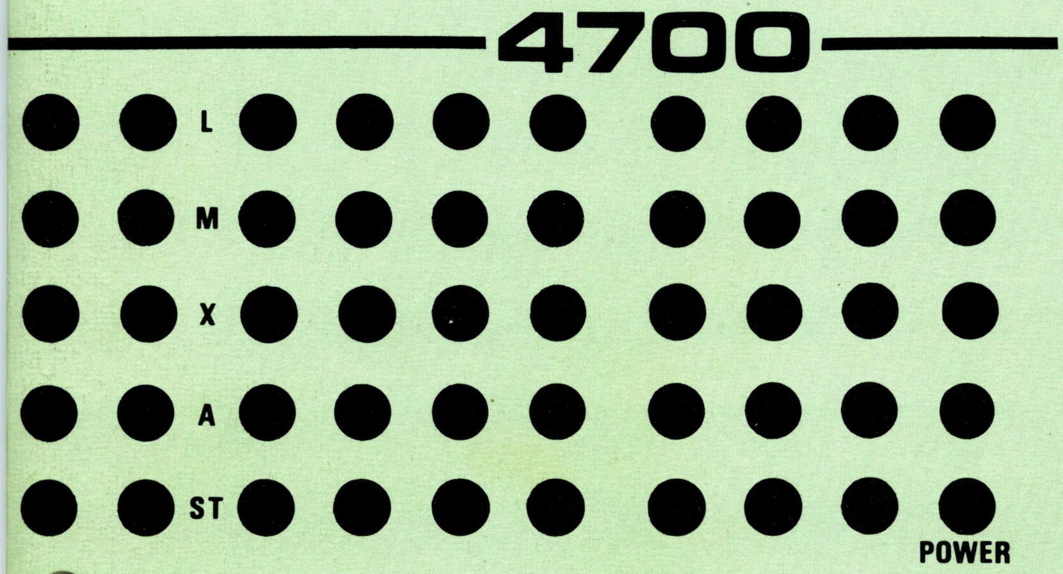
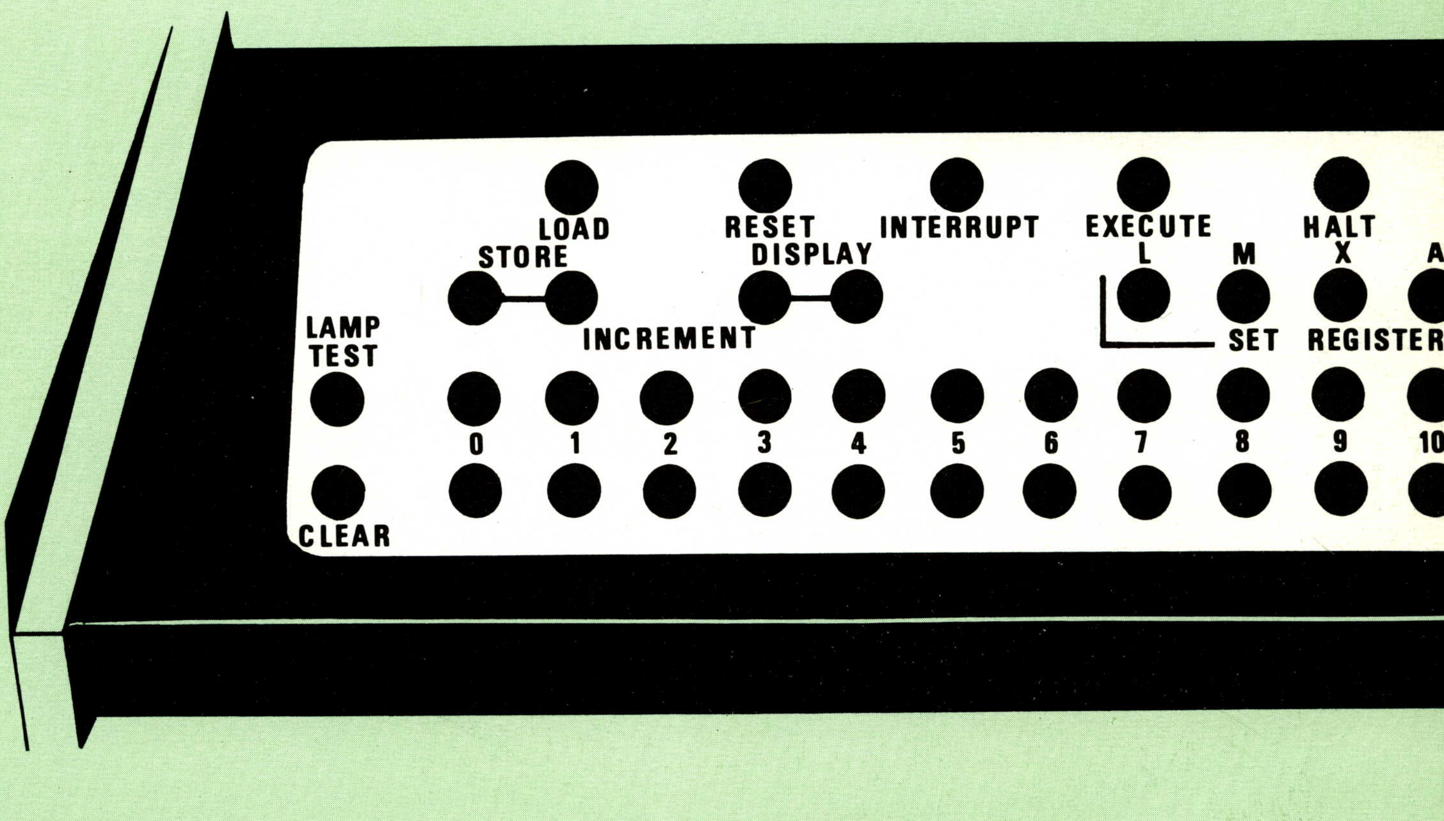
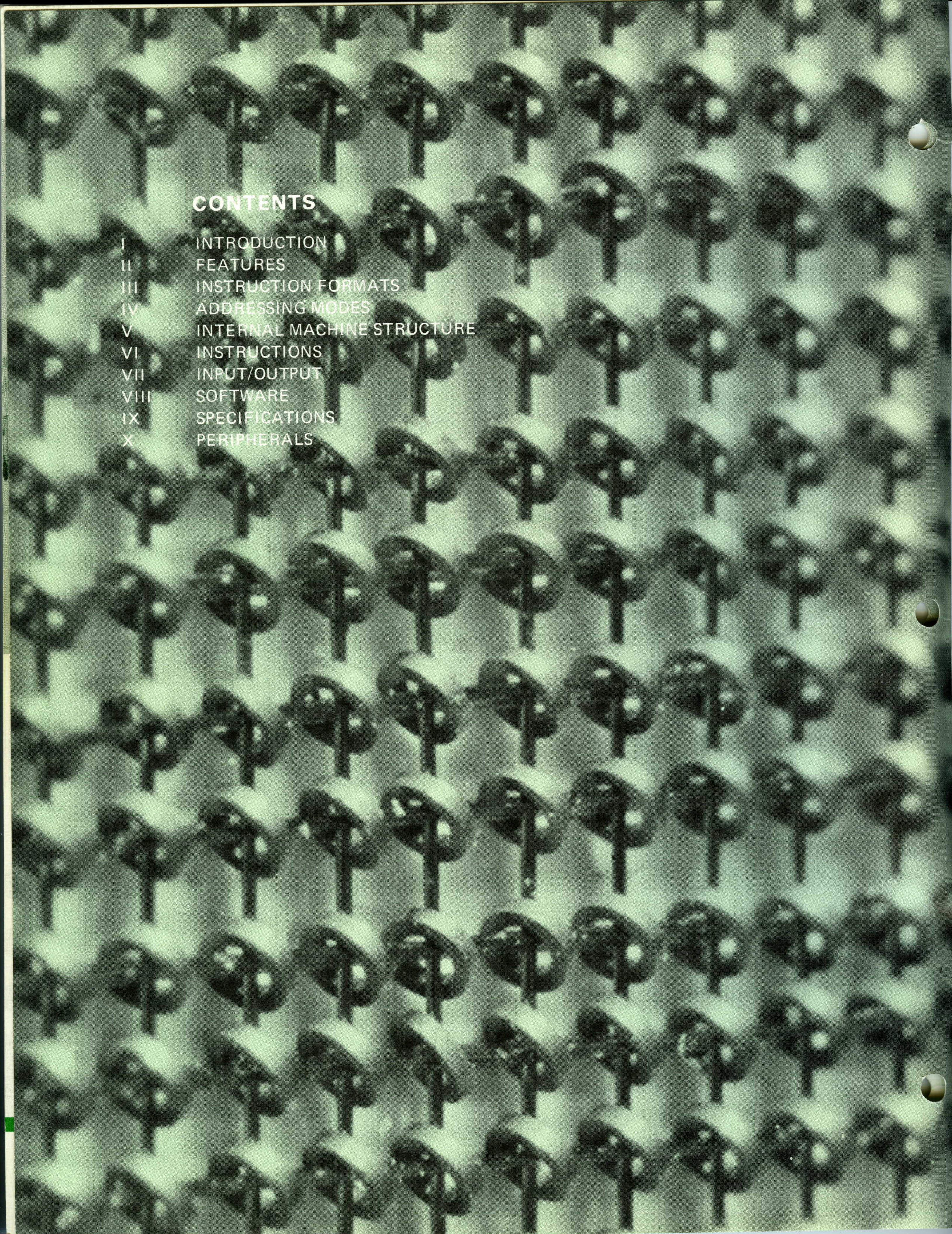
Scientific Control Corporation



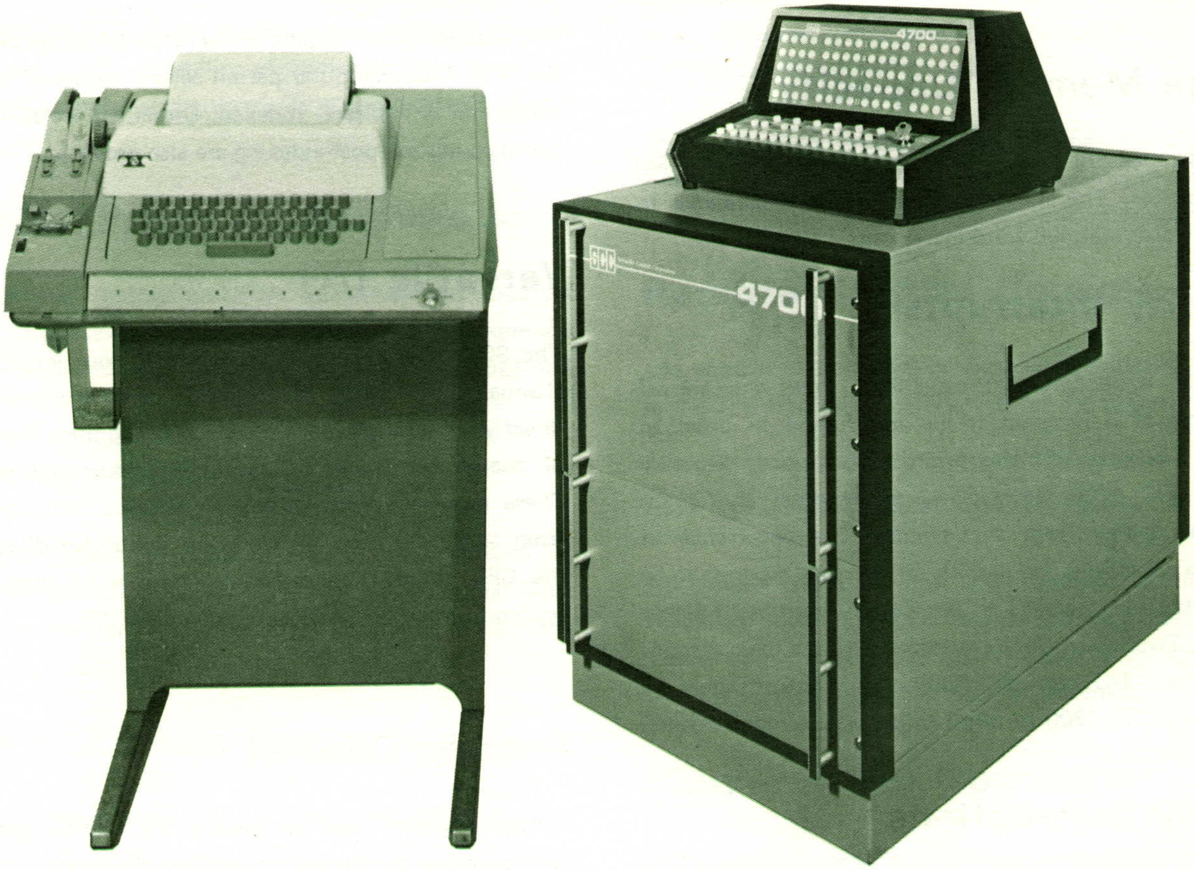
SCC 4700 COMPUTER general description





CONTENTS

INTRODUCTION



The SCC 4700 represents a new design approach to the small computer field. Advanced hardware design techniques and memory mapping concepts provide a cost performance ratio that cannot be matched in any other small computer.

Truly a third generation machine, the 4700 is a general purpose, high-speed, binary computer with a single address type of instruction. A high-speed magnetic core memory module consisting of 4096 sixteen-bit words, with a 920 nanosecond cycle time is offered to permit a wide variety of real time applications.

Outstanding design features:

- A Micro-Programmed read-only memory for internal sequence control.
- Fully integrated circuitry using the most advanced TTL integrated circuits.
- An etched circuit back-plane board eliminating "bird nest" wiring.
- Memory protection providing read, write, or execute protection.
- Byte addressable
- Real time I/O structure

FEATURES

Large Memory Capacity

The basic memory module of the SCC 4700 contains 4,096 16-bit words. The memory capability is expandable to 65,536 words in 4,096 word increments.

Priority Interrupts

The SCC 4700 interrupt capability allows the normal execution of a program to be interrupted in order to execute a program of higher priority. The priority structure of the SCC 4700 interrupt system resolves contention problems arising from the simultaneous occurrence of interrupt conditions and permits an interrupt program to be interrupted by a request to service an interrupt of higher priority. Two interrupts are standard with the basic system: console and I/O channel. Additional interrupts may be added to a SCC 4700 to fulfill varied system requirements.

Literal Instructions

Literal instructions which utilize the last nine bits of the instruction as the operand are included in the instruction set. This feature permits high speed arithmetic or logical operations involving a single character operand.

High Speed Memory

The basic memory module is a 2-1/2D 4,096 16-bit word magnetic core memory with a 920 nanosecond full cycle time. Also available is a 2-1/2D 16,384 16-bit word magnetic core memory module.

Inter-Register Instructions

A special set of instructions called the Operate Group provide logical and arithmetic operations which utilize the index register and the accumulator as operands. This feature also provides the capability of performing conditional jumps which test the result of a logical or arithmetic operation performed upon the index register and/or accumulator.

Addressing Modes

The addressing techniques available to the programmer in the SCC 4700 computer permit several modes of memory addressing – Indirect, Indexed, Direct, or Relative. Both pre-indexing and post-indexing are also provided.

Versatile I/O

The SCC 4700 may have from one to four input/output channel units which communicate with memory either directly or through the central processing unit. Two types of channel units are offered: multiplexor and selector. These provide a wide variety of input/output capability: from single character data transfer controlled directly by the CPU; to full word parallel transfer of data blocks asynchronously and independently of the CPU.

System Control

System control and autonomy in multiprogramming environments is maintained and simplified through hardware design facilitating transfer of: control; parameters; and status information between the system and the user.

Memory Accessing

A wide range of configurations: from a single memory module accessed directly by the CPU: to sixteen memory modules with each accessible by a memory map unit and up to three selector channel units is available. The memory map unit and the selector channel units access the memory modules through a 4 port channel selector unit.

The capability of asynchronous memory access by either the CPU or the selector channel units is provided by the 4 port channel selector unit. The 4-port channel selector units permit independent memory module accessing and allow complete overlapping of input, processing and output operations. Simultaneous requests of the same module are handled on a priority basis with the priority assignment being a system variable.

Hardware Multiply and Divide

Hardware multiply-divide is offered as an option to provide the user with the capability of implementing a system to achieve his particular requirements at the most economical cost.

Double Precision Arithmetic

An optional double precision instruction set provides extended precision for those applications which require the greater accuracy obtainable through additional significant digits.

Floating Point Hardware

An optional floating point instruction set is offered for scientific application areas. The floating point hardware includes addition, subtraction, multiplication, and division of data in floating point format.

Power Failure Protection

A unique power supply design insures program integrity in the event of power system failures. The SCC 4700 is not affected by power interruptions of less than 20 milliseconds duration.

An optional power failure protect feature initiates an interrupt by which the active registers can be stored in memory by the interrupt subroutines. Reserved power permits 15 milliseconds of normal operation after the detection of power failure. When power returns to the system, a power-up interrupt is generated and the active registers are restored by the software routines.

Memory Mapping and Protection

This optional unit contains a set of associative registers, through which program addresses are transformed to real core locations. It also provides three levels of memory protection: read, write, or execute.

All available memory is divided into 512 word "pages". For each page, the programmer may specify what levels of protection are necessary. (They may be called for singly or in any combination).

Accidental or unauthorized use of a memory location is prevented by checking the memory access code for the page being referenced before execution. Attempts to violate the specified protection will cause a trap.

This method of protection permits the maintenance of system integrity in a multiprogramming environment with complete security of system and application programs.

Remote Control Console

The control console may be installed at a location remote to the CPU to provide ready access for the operator.

All Integrated Circuits

All circuits utilize the most advanced high reliability, fast-switching, low noise TTL integrated circuit components. These components mount on extra large printed circuit plug-in boards in a "register slice" arrangement.

Fully Parallel Operation

Fully parallel internal processing of all instructions together with fully parallel data transfer between memory and the CPU provide a high speed system performance capability.

Micro-Programmed Hardware

The instruction repertoire is micro-programmed to provide a highly flexible and expandable instruction set with maintainability and reliability. The design and organization of the micro-programmed instruction repertoire permits the addition of additional instruction sets such as floating point arithmetic to be implemented at a modest cost.

Memory Parity

An optional feature, memory parity error detection, is available with the memory modules. An extra parity bit is included in each word of every memory module. When parity checking is desired, only the logic circuits necessary to provide parity error detection need be added.

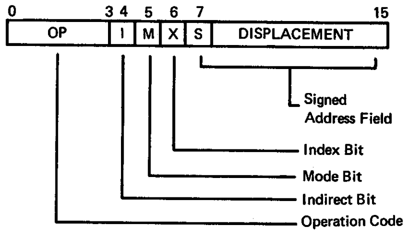
INSTRUCTION FORMATS

The formats of each type of instruction are given in this section. Which format is to be used for each instruction is indicated with the functional definition of each instruction in Section VI.

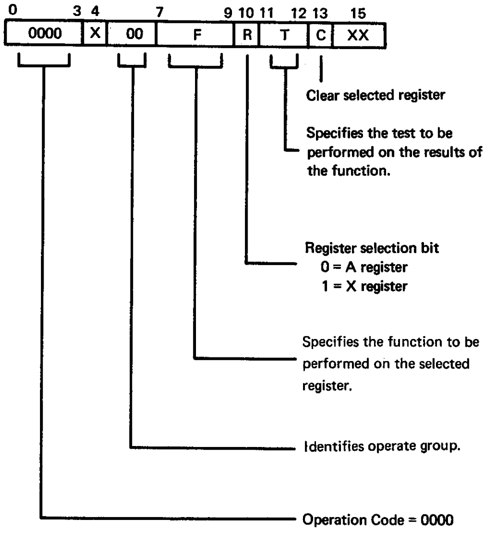
In using the indirect address word format, if the primary address is the current location plus one, the location counter is incremented to skip (L + 2) the next word in the instruction sequence. In this way, both operands and full addresses may be included "in-line".

Similarly, the JRT and SRT instructions automatically return on a pointer contained in the following location (L + 1), because they themselves do not have an address field (extended format).

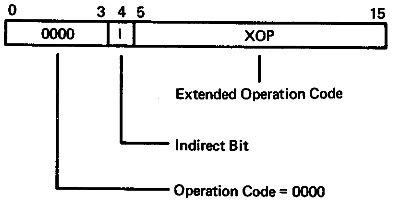
BASIC



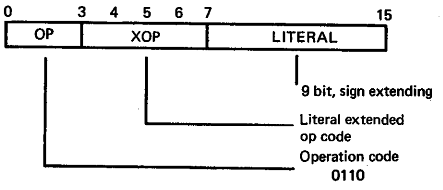
OPERATE GROUP


0
3 4
7
9 10 11
12 13
15
0000
X
00
F
R
T
C
XX
Clear selected register
Specifies the test to be performed on the results of the function.
Register selection bit
0 = A register
1 = X register
Specifies the function to be performed on the selected register.
Identifies operate group.
Operation Code = 0000

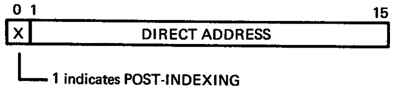
ARITHMETIC



LITERAL

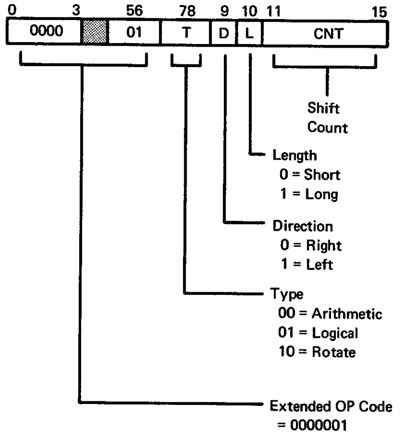


INDIRECT ADDRESS



Allows direct addressing up to 32K ($7FFF_{16}$).

SHIFT



ADDRESSING MODES

Address modification in the 4700 is based on two concepts:

PRIMARY ADDRESS

The intermediate address which is determined before indirect addressing and post-indexing are applied. It becomes the effective address if indirect addressing is not applied.

EFFECTIVE ADDRESS

The final address which is formed after all address modification and indexing have been performed.

There are five possible modes available for instructions of basic format, each of which results in a different effective address when implemented, either singly or in combination. They are:

Direct (Bit 5 = 0)

The primary address is determined by the address field of the instruction. In the direct mode (without memory mapping), the primary address always refers to the first 512 locations of memory.

Relative (Bit 5 = 1)

The primary address is the sum of the address field of the instruction and the contents of the location counter.

Primary Indexed (Bit 6 = 1)

The contents of the index register are added to the primary address (either direct or relative) to form the primary indexed address.

Indirect (Bit 4 = 1)

The indirect address bit is always applied after the contents of the primary address have been obtained. If the indirect address bit is zero, the contents of the location specified by the primary address is used as the operand of the instruction. If the indirect address bit is a one, the contents of the location specified by the primary address is interpreted not as an operand, but as a 15 bit operand address. (Bit 0 of the indirectly addressed location is tested for post-indexing).

Post-Indexed

In this mode, after the contents of the indirectly addressed location specified by the basic instruction are obtained, bit 0 is checked. If it is equal to one, the contents of the index register are added to the other 15 bits to form the effective operand address.

Addressing with Memory Mapping

When Memory Mapping is implemented, every effective address formed by memory reference instructions must be transformed into a real address before execution. Since all programs are assembled without regard to where they may be placed in core at execution time, the monitor keeps track of their real location by use of the associative registers in the memory map unit.

The format of the mapping registers is:

1. Six bits for comparison with the high order six bits of the effective address of the instruction.

2. Seven bits indicating the real core location of the page containing the instruction.

3. Three bits to set the desired memory protection for the page being used.

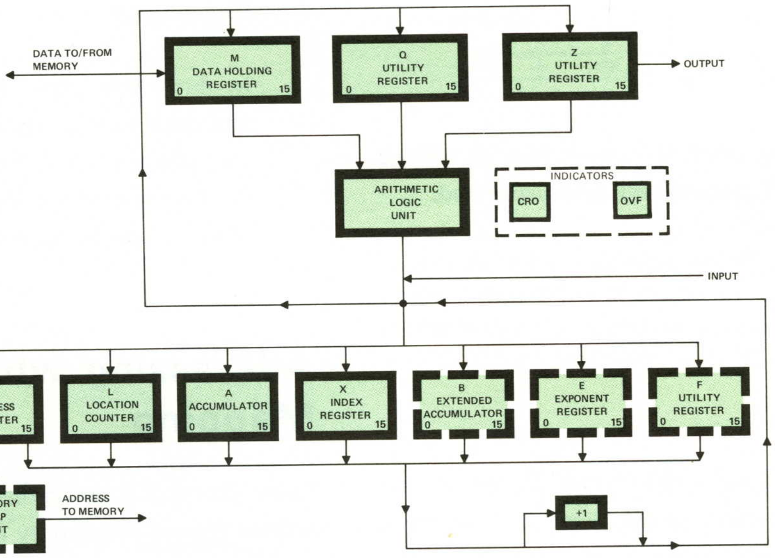
In the direct mode, the address will be relative to location 00_{16} of the page currently being referenced, plus bits 7-15 of the instruction.

INTERNAL MACHINE STRUCTURE

The figure below shows a block diagram of the internal structure of the SCC 4700, Central Processor Unit. In this illustration, all arrows indicate data paths and all transfers are 16 bit parallel transfers. Optional registers are enclosed with heavy broken lines.

As shown in this illustration, the basic 4700 Central Processor Unit contains seven 16-bit registers. The two optional registers and the memory map unit are utilized with the optional instruction sets. This internal organization provides extremely fast data and arithmetic capability.

SCC 4700
CENTRAL PROCESSOR UNIT
REGISTERS AND PARALLEL DATA PATHS



M Register — The M Register accepts data from memory and transfers data to memory. The M Register serves as a holding register for instructions and data. In the halt mode, the M Register contains the instruction which will be executed next.

Q Register — The Q Register is a utility register and holding register into the arithmetic logic unit.

Z Register — The Z Register is a utility register and holding register into the arithmetic logic unit. Output data passes through the Z to the input/output busses.

S Register — The S Register is the memory address register and supplies a 16 bit address to the memory unit (or memory map unit if present).

L Register — The L Register is the location counter (program address register) and contains the address of the instruction being fetched. In the relative mode, bits 7-15 of the instruction (address field) are added to the contents of the L to determine the effective address.

A Register — The A Register is the main accumulator. Most arithmetic and logical operations use this register to hold the result of the operation.

X Register — The X Register is used for indexing, address modification and as an extension of the accumulator for certain instructions. Inter-register operations between the X and A may be performed.

B Register (Optional)–The B Register serves as a second accumulator and extension of the A Register for double precision and floating point operations.

E Register (Optional) — The E Register (exponent register) is utilized during floating point instructions to hold the exponent of a number in floating point format.

F Register (Optional)–The F Register is a utility and holding register to provide high speed, internal operations during double-precision and floating-point instructions.

INSTRUCTIONS

The instruction repertoire of the SCC 4700 is described below. The instruction repertoire includes four optional instruction sets: 1) double-precision option, 2) floating point option, 3) multiply-divide option, and 4) memory map option. The conventions used in instruction description are as follows:

(1) A register name enclosed in parentheses denotes the contents of the register.

(2) Subscription is used to denote bit positions within a register.

(3) The right arrow symbol, " → " is read — "is transferred into."

(4) A micro-step equals 230 nanoseconds.

(5) A location enclosed within parentheses ( ) indicates an indirect address.

(6) The abbreviations are defined as:

Q = Effective Address

L = Location Counter

A = Accumulator

X = Index Register

INS = Instruction

ST = Status Register

CRO = Carry Out Indicator

0 = Reset

OVF = Overflow

MR = Map Register

B = Extended Accumulator

E = Exponent Register

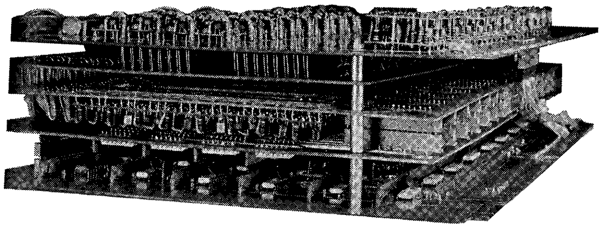
BA = Base Address

(7) The instruction format is indicated as follows:

B	Basic
L	Literal
E	Extended
D	Double-precision, multiply-divide, floating point.
S	Shift

(8) The optional instruction sets are indicated as follows:

MD	Multiply-Divide
FP	Floating Point
DP	Double Precision
MM	Memory Map



Load and Store Instructions

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
LDA	Load Accumulator (Q) → A	8	B	
STA	Store Accumulator (A) → Q	8	B	
LDX	Load Index (Q) → X	8	B	
STX	Store Index (X) → Q	8	B	
LDB	Load Byte $0 \rightarrow A_{0-7}$ $(Q) \rightarrow A_{8-15}$	9	B	
STB	Store Byte $(A)_{8-15} \rightarrow Q$	12	B	
LDL	Load A, Literal $INS_7 \rightarrow A_{0-7}$ $(INS)_{8-15} \rightarrow A$	6	L	
LDLX	Load X, Literal $(INS)_7 \rightarrow X_{0-7}$ $(INS)_{8-15} \rightarrow X_{8-15}$	6	L	
LAS	Load Accumulator From Switches (Switches) → A	5	E	
LDD	Load Double (Q) → A (Q + 1) → X	16	D	DP
STD	Store Double (A) → Q, (X) → Q + 1	16	D	DP
LDF	Load Floating (Q) → A (Q + 1) → B (Q + 2) → E	20	D	FP
STF	Store Floating A → (Q) B → (Q + 1) E → (Q + 2)	20	D	FP

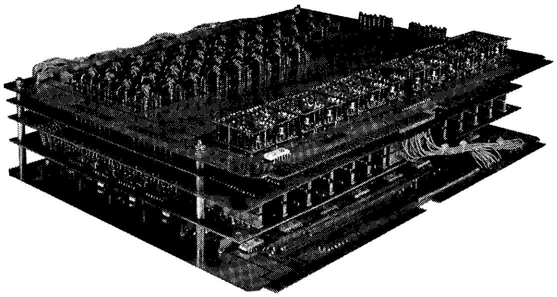
Arithmetic Instructions

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
ADD	Add to Accumulator (A) + (Q) → A	8	B	
SUB	Subtract from Accumulator (A) – (Q) → A	8	B	
ADL	Add to A, Literal $(INS)_{7-15} + (A) \rightarrow A$	6	L	
ADLX	Add to X, Literal $(INS)_{7-15} + (X) \rightarrow X$	6	L	
MPY	Multiply (A) x (Q)→A,X	25	D	MD
DIV	Divide (A,X)/(Q) → A Remainder → X	26	D	MD
MIN	Memory Increment Skip on Zero (Q) + 1 → Q If (Q)=0, (L)+2→L If (Q)≠0,(L)+1→L	10	B	
ADC	Add Carry (A) + (CRO) → A Carry → CRO 1 → OVF, if overflow	6	E	

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
DAD	Double Add (A,B)+(Q,Q+1)→A,B If overflow, 1→OFV If carryout, 1→CRO	17	D	DP
DSB	Double Subtract (A,B)-(Q,Q+1)→A,B Ifoverflow, 1→OFV If carryout, 1→CRO	17	D	DP
DMP	Double Multiply (A,B)x(Q,Q+1)→A,B Ifoverflow, 1→OVF 0→CRO	60	D	DP
DDV	Double Divide (A,B)÷(Q,Q+1)→A,B Ifoverflow, 1→OVF 0 → CRO	60	D	DP
FAD	Floating Point Add (A,B,E)+(Q,Q+1,Q+2)→A,B,E + denotes floatingadd	32	D	FP
FSB	Floating Point Subtract (A,B,E)-(Q,Q+1,Q+2)→A,B,E -denotesfloatingsubtract	35	D	FP
FMP	Floating Point Multiply (A,B,E)x(Q,Q+1,Q+1)→A,B,E x denotesfloatingmultiply	75	D	FP
FDV	Floating Point Divide (A,B,E)÷(Q,Q+1,Q+2)→A,B,E ÷denotesfloatingdivide	75	D	FP

Logical Instructions

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
AND	AND With Accumulator (A)∩(Q) → A	8	B	
XOR	Exclusive OR With Accumulator (A)⊕(Q) → A	8	B	
AAX	AND Accumulator With Index (A)∩(X) → A	7	E	
AOX	OR Accumulator With Index (A) ∩(X) → A	7	E	
ANL	AND the Accumulator, Literal $(INS)_{7-15}\cap(A)\rightarrow A$	6	L	
ANLX	AND the Index, Literal $(INS)_{7-15}\cap(X)\rightarrow X$	6	L	
XOL	Exclusive OR the Accumulator, Literal $(INS)_{7-15}\oplus(A)\rightarrow A$	6	L	
XOLX	Exclusive OR the Index, Literal $(INS)_{7-15}\oplus(X)\rightarrow X$	6	L	



Register Manipulation

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
CAX	Copy Accumulator into Index (A) → X	5	E	
CXA	Copy Index into Accumulator (X) → A	5	E	
XAX	Exchange Accumulator and Index (A) → X (X) → A	7	E	
XHA	Exchange Halves of Accumulator $(A)_{0-7} \rightarrow A_{8-15}$ $(A)_{8-15} \rightarrow A_{0-7}$	12	E	
ESA	Extend Sign of Accumulator $(A)_0 \rightarrow X_{0-15}$	5	E	
CAB	Copy Accumulator into B (A) → B	5	D	DP
CBA	Copy B into Accumulator (B) → A	5	D	DP
XAB	Exchange Accumulator and B (A) ↔ (B)	7	D	DP
CLB	Clear B $0 \rightarrow B_{0-15}$	5	D	DP

Operate Group

A special group of instructions which perform inter-register manipulations. The operations listed below correspond to actual phases of the instructions itself, and are executed in the following sequence.

First Operation

Select the register defined by Bit 10
0 = A Register
1 = X Register

Second Operation

Perform one of eight functions as specified by Bits 7 through 9

000	Test Selected Register only
001	Increment Selected Register (Sel Reg) ⊕ 1 → Sel Reg
010	Add Unselected Register to Selected Register (A) + (X) → Sel Reg
011	Exclusive OR A and X (A) ⊕ (X) → Sel Reg
100	One's complement of Selected Register (Sel Reg) → Sel Reg
101	Two's complement of Selected Register (Sel Reg) + 1 → Sel Reg
110	Decrement Selected Register (Sel Reg) — 1 → Sel Reg
111	Subtract Selected Register from Unselected Register (Unsel Reg) — (Sel Reg) → Sel Reg

Third Operation

Test the results of the above function and skip as follows:

Bits 11 and 12	
00	No Skip
01	Skip if positive (> 0)
10	Skip if negative
11	Skip if Zero

Fourth Operation

Clear selected register if Bit 13 is set. If Bit 13 = 0, do not clear selected register.

Examples

0100	One's complement of A—No skip
0108	One's complement of A—Skip if positive
0110	One's complement of A—Skip if negative
0118	One's complement of A—Skip if zero
0120	One's complement of X—No skip
0128	One's complement of X—Skip if positive
0130	One's complement of X—Skip if negative
0138	One's complement of X—Skip if zero
0140	Two's complement of A—No skip
0148	Two's complement of A—Skip if positive
0150	Two's complement of A—Skip if negative
0158	Two's complement of A—Skip if zero
0160	Two's complement of X—No skip
0168	Two's complement of X—Skip if positive
0170	Two's complement of X—Skip if negative
0178	Two's complement of X—Skip if zero
0180	Decrement A, Place result in A register—No Skip
0188	Decrement A, Place result in A register—Skip if positive
0190	Decrement A, Place result in A register—Skip if negative
0198	Decrement A, Place result in A register—Skip if zero
01A0	Decrement X, Place result in X register—No skip
01A8	Decrement X, Place result in X register—Skip if positive
01B0	Decrement X, Place result in X register—Skip if negative
01B8	Decrement X, Place result in X register—Skip if zero
01C0	Subtract A from X, Place result in A register—No skip
01C8	Subtract A from X, Place result in A register-Skip if positive
01D0	Subtract A from X, Place result in A register—Skip if negative
01D8	Subtract A from X, Place result in A register—Skip if zero
01E0	Subtract X from A, Place result in X register—No skip
01E8	Subtract X from A, Place result in X register—Skip if positive
01F0	Subtract X from A, Place result in X register—Skip if negative
01F8	Subtract X from A, Place result in X register—Skip if zero

0004	Clear A register
0024	Clear X register
0008	Test A register and skip if positive
0010	Test A register and skip if negative
0018	Test A register and skip if zero
0028	Test X register and skip if positive
0030	Test X register and skip if negative
0038	Test X register and skip if zero
0040	Increment A and no skip
0048	Increment A and skip if positive
0050	Increment A and skip if negative
0058	Increment A and skip if zero
0060	Increment X and no skip
0068	Increment X and skip if positive
0070	Increment X and skip if negative
0078	Increment X and skip if zero
0080	Add A and X, results into A register, no skip
0088	Add A and X, results into A register—Skip if positive
0090	Add A and X, results into A register—Skip if negative
0098	Add A and X, results into A register—Skip if zero
00A0	Add A and X, results into X register—no skip
00A8	Add A and X, results into X register—Skip if positive
00B0	Add A and X, results into X register—Skip if negative
00B8	Add A and X, results into X register—Skip if zero
00C0	Exclusive OR A with X, Results into A register—no skip
00C8	Exclusive OR A with X, Results into A register—Skip if positive
00D0	Exclusive OR A with X, Results into A register—Skip if negative
00D8	Exclusive OR A with X, Results into A register—Skip if zero
00E0	Exclusive OR A with X, Results into X register—no skip
00E8	Exclusive OR A with X, Results into X register—Skip if positive
00F0	Exclusive OR A with X, Results into X register—Skip if negative
00F8	Exclusive OR A with X, Results into X register—Skip if zero

Shift Instructions

The shift instructions are conditional shifts which determine the number of positions to be shifted from the "count field" assembled in the instruction $(0 < i < 32)$. The type of shift is determined by bits 7 and 8 of the instruction.

There are four types of shifts, each of which may be either "short" or "long" (and in either direction). "Short" indicates the A register; "long" means between both the A and X register; and circulate links the CRO to the register or registers being shifted.

Arithmetic left shifts do not affect the sign bit. In arithmetic right shifts, the sign bit is propagated from one position to the next, according to the number of shifts required.

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
SAR	Short Arithmetic Right Shift $(A)_i \rightarrow A_{i+1}$ for $0 \leqq i < 15$ $(A)_0 \rightarrow A_1$	6	S	
SAL	Short Arithmetic Left Shift $(A)_i \rightarrow A_{i-1}$ for $0 < i \leqq 15$ $0 \rightarrow A_{15}$	6	S	
LAR	Long Arithmetic Right Shift $(A)_i \rightarrow A_{i+1}, (X)_i \rightarrow X_{i+1},$ for $0 \leq i \leq 15$ $(A)_0 \rightarrow A_1$ $(A)_{15} \rightarrow X_0$	6	S	
LAL	Long Arithmetic Left Shift $(A)_i \rightarrow A_{i-1}, (X)_i \rightarrow X_{i-1},$ $0 < i \leq 15$ $(X)_0 \rightarrow A_{15}$ $0 \rightarrow X_{15}$	6	S	
SLR	Short Logical Right Shift $(A)_i \rightarrow A_{i+1}, 0 \leq i < 15$ $0 \rightarrow A_0$	6	S	
SLL	Short Logical Left Shift $(A)_i \rightarrow A_{i-1}, 0 < i \leq 15$ $0 \rightarrow A_{15}$	6	S	
LLR	Long Logical Right Shift $(A)_i \rightarrow A_{i+1}, (X)_i \rightarrow X_{i+1},$ $0 \leq i < 15$ $0 \rightarrow A_0$ $(A)_{15} \rightarrow (X)_0$	6	S	
LLL	Long Logical Left Shift $(A)_i \rightarrow A_{i-1}, (X)_i \rightarrow X_{i-1},$ $0 < i \leq 15$ $(X)_0 \rightarrow A_{15}$ $0 \rightarrow 15$	6	S	
SRR	Short Rotate Right $(A)_i \rightarrow A_{i+1}$ for $0 \leq i < 15$ $(A)_{15} \rightarrow A_0$	6	S	
SRL	Short Rotate Left $(A)_i \rightarrow A_{i-1}$ for $0 < i \leq 15$ $(A)_0 \rightarrow A_{15}$	6	S	
LRR	Long Rotate Right $(A)_i \rightarrow A_{i+1}, (X)_i \rightarrow X_{i+1}$ for $0 \leq i < 15$ $(X)_{15} \rightarrow A_0$ $(A)_{15} \rightarrow X_0$	6	S	

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
LRL	Long Rotate Left $(A)_i \to A_{i-1}, (X)_i \to X_{i-1},$ $0 < i \leq 15$ $(X)_0 \to A_{15}$ $(A)_0 \to X_{15}$	6	E	
SCR	Short Circulate Right $(A)_i \to A_{i+1}, 0 \leq i < 15$ $(A)_{15} \to CRO$ $(CRO) \to A_0$	6	E	
SCL	Short Circulate Left $(A)_i \to A_{i-1}, 0 < i \leq 15$ $(A)_0 \to CRO$ $(CRO) \to A_{15}$	6	E	
LCR	Long Circulate Right $(A)_i \to A_{i+1}, (X)_i \to X_{i+1},$ $0 \leq i < 15$ $(CRO) \to A_0$ $(A)_{15} \to X_0$ $(X)_0 \to CRO$	6	E	
LCL	Long Circulate Left $(A)_i \to A_{i-1}, (X)_i \to X_{i-1},$ $0 < i \leq 15$ $(CRO) \to X_{15}$ $(X)_0 \to A_{15}$ $(A)_0 \to CRO$	6	E	

Extended Shift Instructions

The double shift and double rotate instructions treat the A and B registers as a single 32 bit register. The double circulate instructions treat the CRO, A and B as a single 33 bit register. The 32 or 33 bit register may be manipulated 0 through 31_{10} bit positions. The Normalize and Decrement X Register is a conditional left shift which decrements the X register by the number of positions shifted.

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
DAR	Double Arithmetic Right Shift	12+2N	S	DP
DAL	Double Arithmetic Left Shift	12+2N	S	DP
DLR	Double Logical Right Shift	12+2N	S	DP
DLL	Double Logical Left Shift	12+2N	S	DP
DRR	Double Rotate Right	12+2N	S	DP
DRL	Double Rotate Left	12+2N	S	DP
DCR	Double Circulate Right	12+2N	S	DP
DCL	Double Circulate Left	12+2N	S	DP
NDX	Normalize and Decrement X	12+2N	S	DP

Jump and Skip Instructions

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
JMP	Jump Q → L If Bit 4=1, Q=(L)+(INS)	4 7-15	B	
JSL	Jump and Store Location L + 1 → ((Q)) Q + 2 → L	17	B	
JRT	Jump Return ((L + 1)) → L	16	E	
COT	Carry Out Test (L) + 1 → L if CRO = 1 (L) + 2 → L if CRO = 0	5	E	

OFT	Overflow Test
	(L) + 1 → L,0 → OVF;
	if OVF = 1
	(L) + 2 → L, if OVF = 0
		5	E	

Privileged Instructions

With the memory mapping option certain instructions can be executed only in the system mode. These are: all of the system and I/O instructions; and some of the control instructions. The control instructions which are privileged, are indicated by asterisks.

All system calls will be executed from a base address (BA). A table of pointers to monitor entry points will follow the base address. As many as 64 entry points may be designated by INS_{10-15}. When encountered, a trap to BA + INS_{10-15} will occur and the instruction will be executed as defined.

Control Instructions

Mnemonic	Instruction	Micro-Steps	Instruction Format	Optional Set
HLT	Halt	6	E	*
ENA	Enable Interrupts Enable Interrupt System	6	E	*
DIS	Disable Interrupts Disable Interrupt System	6	E	*
CLI	Clear Interrupt Clear Current Priority Interrupt	6	E	*
SCO	Set Carry Out 1 → CRO	5	E	
RCO	Reset Carry Out 0 → CRO	5	E	

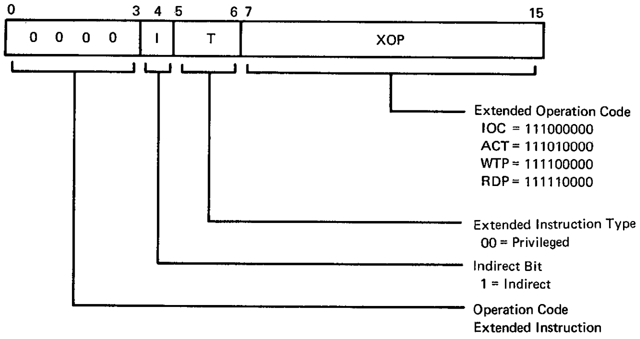
System Instructions

LMR	Load Map Register (A) → Memory Mapping Unit	8	E	MM
SRT	System Return (((L+1)))→L (((L+1))+1)→Status (((L+1))+2)→A (((L+1))+3)→X	24	E	MM
SCL	System Call (L)→((BA+INS_{10-15})) (ST)→((BA+INS_{10-15})+1) (A)→((BA+INS_{10-15})+2) (X)→((BA+INS_{10-15})+3) O→CRO,OVF (BA+INS_{10-15})→L	20	E	MM

INPUT/OUTPUT

Input/Output Instructions

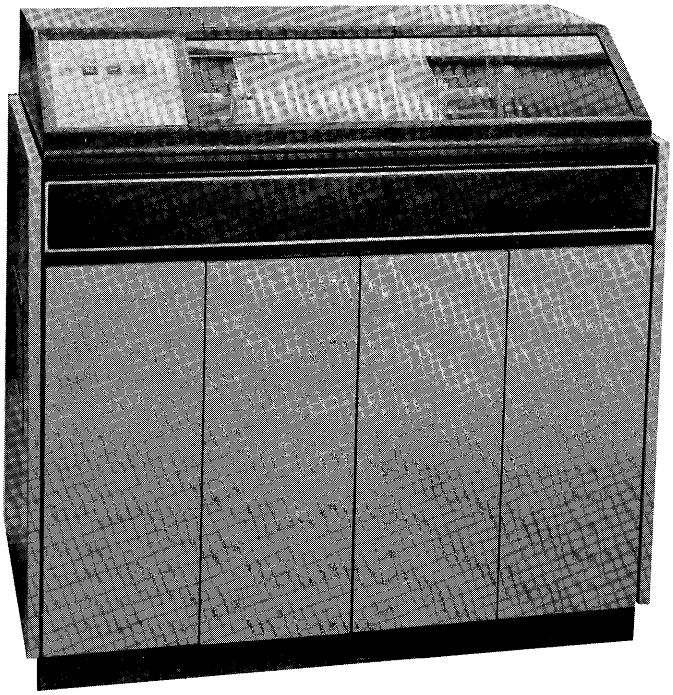
The four I/O instructions are double-word instructions. The first word specifies the I/O instruction and is in the extended instruction format. The second word specifies additional command and control information if bit 4 of the I/O instruction (first word) is reset. If bit 4 of the I/O instruction is set, the second word specifies a 15 bit address pointer and bit 0 of the second word is interrogated for post-indexing information. The double-word I/O instructions occupy contiguous memory locations and may start on even or odd-numbered locations. The format of the first word of the I/O instructions is:



The I/O Instructions for the 4700 are:

Mnemonic	Instruction	Instruction Format	Optional Set
ACT	Activate Parallel I/O	I/O	
RDP	Read Parallel	I/O	
WTP	Write Parallel	I/O	
IOC	Input/Output Control	I/O	

Mnemonic	Instruction	Instruction Format	Optional Set
SIO	Start I/O (Select Device)	E	
HIO	Halt I/O (Disconnect Immediately)	E	
TWC	Terminate When Complete	E	
ODC	Output Data Character	E	
IDC	Input Data Character	E	
OUS	Output Unit Status From A	E	
IUS	Input Unit Status into A	E	
ECA	Execute Command in A	E	
IIU	Input Interrupting Unit into X	E	
SDR	Skip if Device Ready	E	



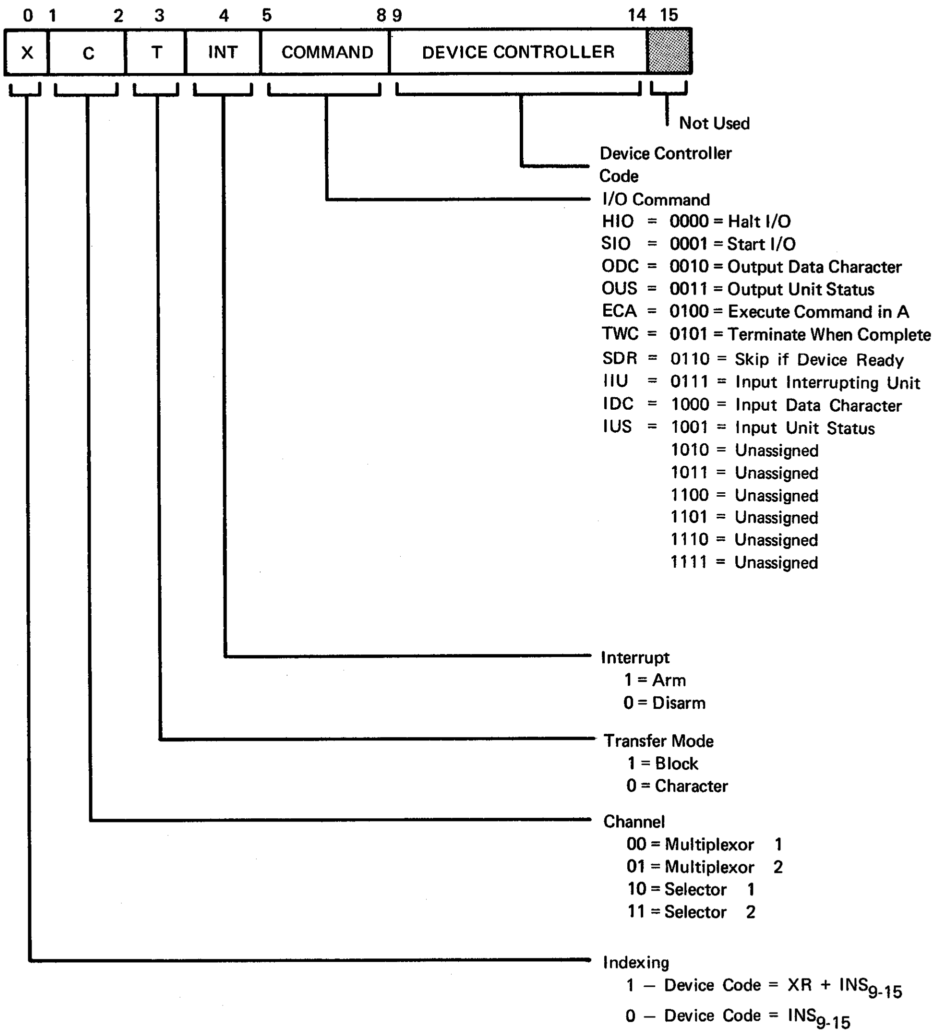
The ACT (Activate Parallel I/O) instruction activates I/O devices which utilize the parallel input/output interface. The second word of the ACT instruction specifies or points to device selection and control information. The interpretation of this 16 bits of information is dependent upon the user's system requirements.

The RDP (Read Parallel) instruction transfers a 16 bit data word from the device controller activated via the ACT to the memory location specified or pointed to by the second word of the RDP instruction. The WTP (Write Parallel) instruction transfers a 16 bit data word from the memory location specified or pointed to by the second word of the WTP instruction to the device controller activated via the ACT instruction.

The IOC (Input/Output Control) instruction controls the operation of the multiplexor and selector channels. The second word of the IOC specifies the following:

1. channel
2. device controller
3. I/O command
4. Block or character mode
5. Arm/disarm interrupt.

The format of the second word of the IOC command is:



Not Used

Device Controller Code

I/O Command

HIO	=	0000	=	Halt I/O
SIO	=	0001	=	Start I/O
ODC	=	0010	=	Output Data Character
OUS	=	0011	=	Output Unit Status
ECA	=	0100	=	Execute Command in A
TWC	=	0101	=	Terminate When Complete
SDR	=	0110	=	Skip if Device Ready
IIU	=	0111	=	Input Interrupting Unit
IDC	=	1000	=	Input Data Character
IUS	=	1001	=	Input Unit Status
		1010	=	Unassigned
		1011	=	Unassigned
		1100	=	Unassigned
		1101	=	Unassigned
		1110	=	Unassigned
		1111	=	Unassigned

Interrupt
1 = Arm
0 = Disarm

Transfer Mode
1 = Block
0 = Character

Channel
00 = Multiplexor 1
01 = Multiplexor 2
10 = Selector 1
11 = Selector 2

Indexing
1 – Device Code = XR + $INS_{9\text{-}15}$
0 – Device Code = $INS_{9\text{-}15}$

INPUT/OUTPUT CHANNELS

Input/Output in a SCC 4700 system is by means of one to four input/output channels. Two types of input/output channels are provided:

1. Multiplexor
2. Selector.

One multiplexor channel is a component of the basic 4700 system. Additional input/output channels may be either selector or multiplexor channels. The multiplexor channel is designed to service up to 64 device controllers on a "party-line" or multiplex scheme.

A selector channel provides a buffered channel for up to 64 character oriented device controllers on a block transfer scheme. The selector channel provides for direct memory access, asynchronously and independently, between an I/O device and memory.

The multiplexor and selector channels receive a character count and start address specifying either the left or right half word as the initial boundary. Data is transferred to/from contiguous characters until the character count reaches zero or may be terminated under program control prior to the character count reaching zero.

Multiplexor Channel

The multiplexor channel interfaces up to 64 device controllers with the SCC 4700 CPU on a multiplex basis. The multiplexor channel is under direct control of the CPU and utilizes the CPU data and address paths to memory.

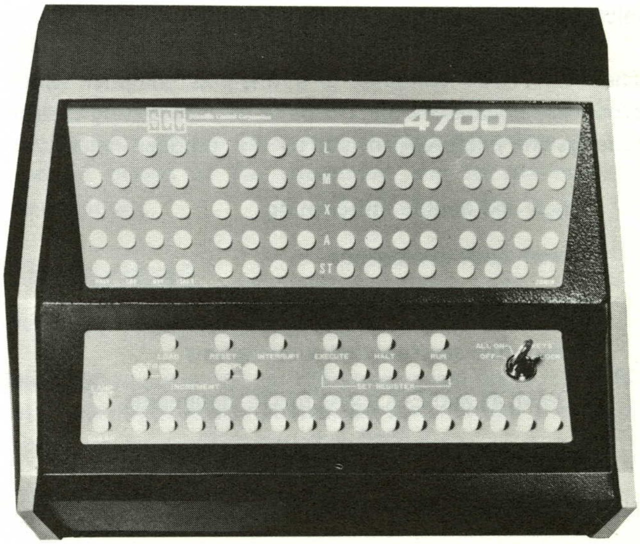
A SCC 4700 system may have one to four multiplexor channels. The multiplexor channel may operate in one of two modes:

1. character
2. block.

In the character or block mode, more than one device controller may be actively engaged in data transfer at any one time. In either case, data transfer between the channel and the device controllers is on a character oriented, multiplex basis. The active device controllers need not operate in the same mode: i.e., some device controllers may be engaged in data transfer in the character mode while other device controllers are transferring data in the block mode.

Selector Channel

The selector channel interfaces up to 64 device controllers with the SCC 4700 CPU on a block or record transfer basis. The selector channel is activated by the CPU and proceeds autonomously of the CPU to access memory for command and control information. Utilizing the start address and character count specified by the command and control information, the selector channel performs the data transfer asynchronously and independently of the CPU; the selector channel deactivates the channel at the end of the data transfer. Block transfers may be chained (that is, the selector channel can sequentially execute a series of block transfers).



SOFTWARE

SPL Assembler

The SCC 4700 is provided with a SPL assembler. This translates Symbolic Programming Language statements to machine language in two passes.

Programming features provided by the assembler are:

- Mnemonic Operation Codes
- Symbolic Addressing
- Storage Allocation
- Floating Point Constants
- Fixed Point Constants with Binary Point Specifications
- Program Communication Links
- Absolute or Relocatable Output
- Conditional Assembly Capabilities
- Editing and Update Capabilities
- Program Listings
- Diagnostics

Basic FORTRAN System

The SCC 4700 basic FORTRAN system contains four major elements: the compiler, the run time system, the loader, and the library. The system requires only a basic configuration of a SCC 4700 with 4K memory and primary teletype.

The FORTRAN compiler is a one-pass compiler which produces relocatable object code to be executed by the run time system. Features provided by the compiler include:

- Single and double subscripting
- Mixed-mode expressions
- Subscripts may be any expression
- Subroutine and function subprograms
- COMMON and EQUIVALENCE statements
- I, E, F, H, X in format specifications

The FORTRAN loader provides the following features:

- Loading, Relocating and Linking Programs and Subprograms
- Retrieval and loading of routines from the FORTRAN Library Tape

The FORTRAN Library contains the basic mathematical functions and intrinsic functions which include:

- Sine
- Cosine
- Arctangent
- Natural Logarithm
- Exponential
- Arctangent
- Square Root

FORTRAN IV

The SCC 4700 FORTRAN IV compiler conforms to the standard FORTRAN language specified by the American Standards Institute. This FORTRAN IV compiler and the SPL Assembler generate compatible code. Therefore, FORTRAN routines may call SPL sub-programs and SPL sub-programs may call FORTRAN routines without restriction. The FORTRAN IV system will require a minimum of 8K of memory. This system will also contain four major elements: the compiler, the run time system, the loader, and the library.

Double Precision/Floating Point Packages

The double precision and floating point packages include all the capabilities provided by their hardware counterparts. They are designed to be functionally identical to the hardware packages.

Debug Packages

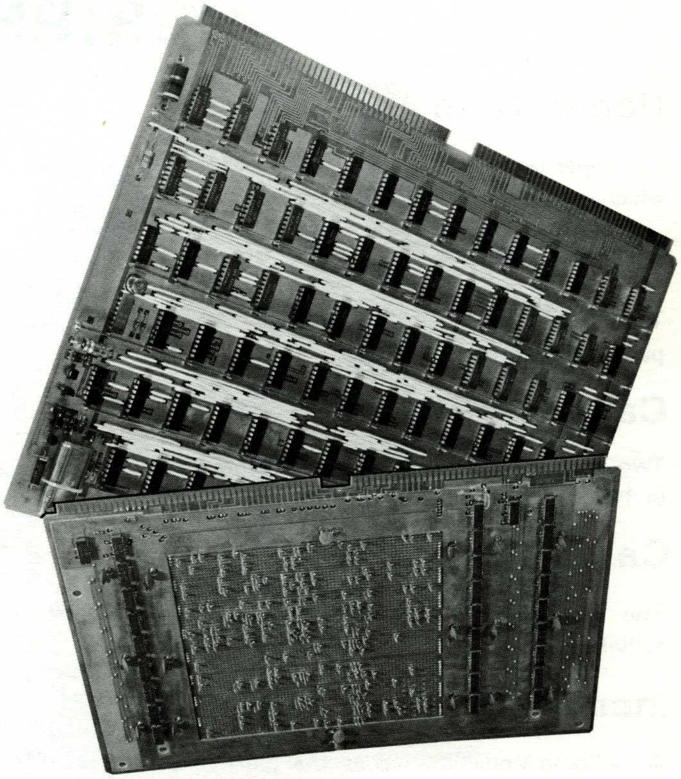
The debugging package provides the programmer with a convenient, interactive debugging system. Under keyboard control, the programmer may:

- Display principal registers
- Display and change memory registers
- List an area of memory
- Set program breakpoints
- Initialize memory
- Search memory for a specified bit pattern
- Dump an area of memory on paper tape in absolute loader format

Real Time Monitor

The SCC 4700 real time monitor (RTM) provides an efficient multiprogramming environment for real time applications. The RTM permits several co-resident programs to share the computer system. Protection is provided for all co-resident programs, not just those operating in the "foreground" mode. The general features provided by the RTM are as follows:

1. Perform user Input/Output.
2. Execute all privileged hardware functions such as handling of I/O, interrupts, and memory allocation or protection.
3. Allocate system resources such that program integrity is assured for all users.
4. Schedule processing time.



SPECIFICATIONS

Mechanical

Mounting Configurations

The CPU and power supply are contained in two separate chassis. The CPU is 12¼" X 19" X 27" and the power supply is only 8¾" X 19" X 27". Each chassis is finished with a blue front panel and decorative handles. These units are designed to be mounted in any standard RETMA 19 inch relay rack. Weight of the two units is approximately 125 pounds.

The control console and display panel are mounted together either in the front panel of the CPU unit or in a special remote control console which is connected to the CPU by cable.

The CPU, power supply, and control console may be mounted any way the customer desires: in his own equipment rack, as a special wheel mounted unit, or in a desk.

Mobile Unit

The wheel mounted cabinet model is only 2½ feet high, weighs 150 pounds and can be used with the remote control console unit.

Desk Mounted

The desk mounted model is contained on one side of the desk with space available on the other side for supplies and program storage or an input/output device. The remote control console unit can be placed on top of the desk either fixed in place or free to move around.

Other options can be arranged, depending on the space requirements of the customer, to take advantage of the modular design of the equipment and its miniaturized construction.

Electrical

Power Requirements

The 4700 requires 105 to 125 volts, single phase, 50 to 60 cycle AC at 8.5 amps nominal. By special order the power supply can be modified to utilize 220 volts.

Digital Signal Levels

False – 0 to 0.3 VDC Nominal

True – +3 to 4.0 VDC Nominal

PERIPHERALS

Paper Tape Reader

The paper tape reader is an eight channel photoelectric unit which operates at a rate of 300 characters per second.

Paper Tape Punch

Two 8-channel paper tape punches are available for punching at 50 or 120 CPS.

Card Readers

Two punched card readers are available for reading at 200 or 1100 cards per minute.

Card Punch

The card punch operates at a punching rate of 200 fully-laced 80 column cards per minute.

Incremental Plotter

A CalComp Model 565 incremental plotter and control unit is available for the high speed plotting of points, continuous curves and points connected by lines.

Teletype Input/Output Device

The Model 33 ASR Teletype set can be supplied with each computer. The ASR 33 contains both paper tape reader and paper tape punch in addition to the normal typewriter keyboard. (Optional - ASR 35 Teletype.) The teletype unit is equipped with a pin-feed platen. Input rate is 10 CPS.

Digital to Analog Converter

The SCC digital to analog converter is an addressable multichannel, binary, two's complement converter. Unit may be 7 bit plus sign bipolar or 8 bit unipolar.

Selectric Typewriter

An IBM Selectric Typewriter with a maximum speed of 15 characters per second is available. The typewriter is equipped with a pin-feed platen allowing the use of continuous form paper. Either 8 or 15 inch platen models are available.

Line Printers

Three line printer models are available at printing rates of 300, 600, or 1000 lines per minute. Each machine prints a selection of 64 characters with a 132 characters per line. The vertical format is controlled by a punched paper tape within the printer under program or operator control.

Magnetic Tape Controller

Magnetic tape controllers are available to control from one to four seven track magnetic tape transports or from one to four, nine track tape transports.

Magnetic Tape Transports

Four magnetic tape transports are available; two for seven track tape and two for nine track tape. Each tape transport can read or write in densities of 200, 556, or 800 bpi. Speeds of 45 ips or 75 ips are available.

Magnetic Disc Units

Magnetic disc units are available which provide memory in six, 29, or 54 million bit modules with high speed access.

Communications Controllers

Three communications controller models are available. One of the models provides line termination and multiplexing for one to three full duplex line terminals. A second model provides for multiplexing of one to 32 full duplex line terminals. A third model, a bit-oriented communication controller for one to sixteen teletype circuits with selectable clock rates is available.

Duplex Line Terminal

The duplex line terminal provides modem interface and control logic and line synchronization for a full duplex line.

Multiplexor - Encoder

The multiplexor – encoder allows for the sequential sampling of 64 analog channels of 0 to 5 volts DC. The number of channels can vary in groups of eight, up to the maximum of 64. The multiplexor – encoder contains a buffer amplifier and a sample and hold amplifier. The unit outputs an eleven bit plus sign digital data word. Digitizing rates up to 40 KC can be provided. The computer can address the multiplexor – encoder for sampling the analog channels sequentially.

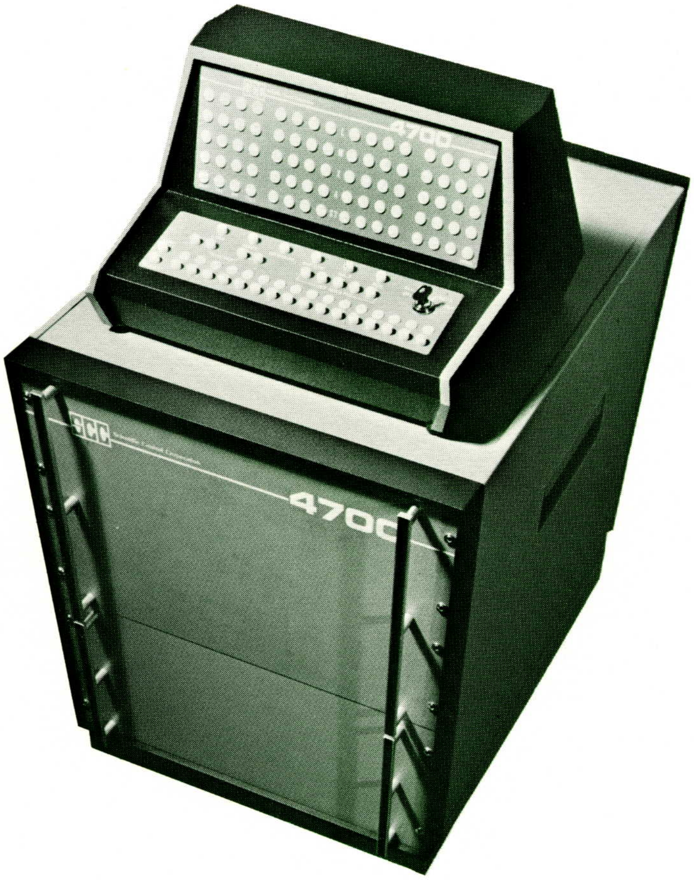
Digital Input Multiplexor

The digital input multiplexor provides for multiplexing one to four, 16-bit digital input channels.

Digital Output Multiplexor

The digital output multiplexor provides for multiplexing one to four, 16-bit, digital output channels.

Remote Control Console
and Display Panel



920 Nanosecond
Memory Cycle

Sixteen External Priority
Interrupts

TTL Integrated
Circuitry

Versatile I/O with Real Time
Capabilities

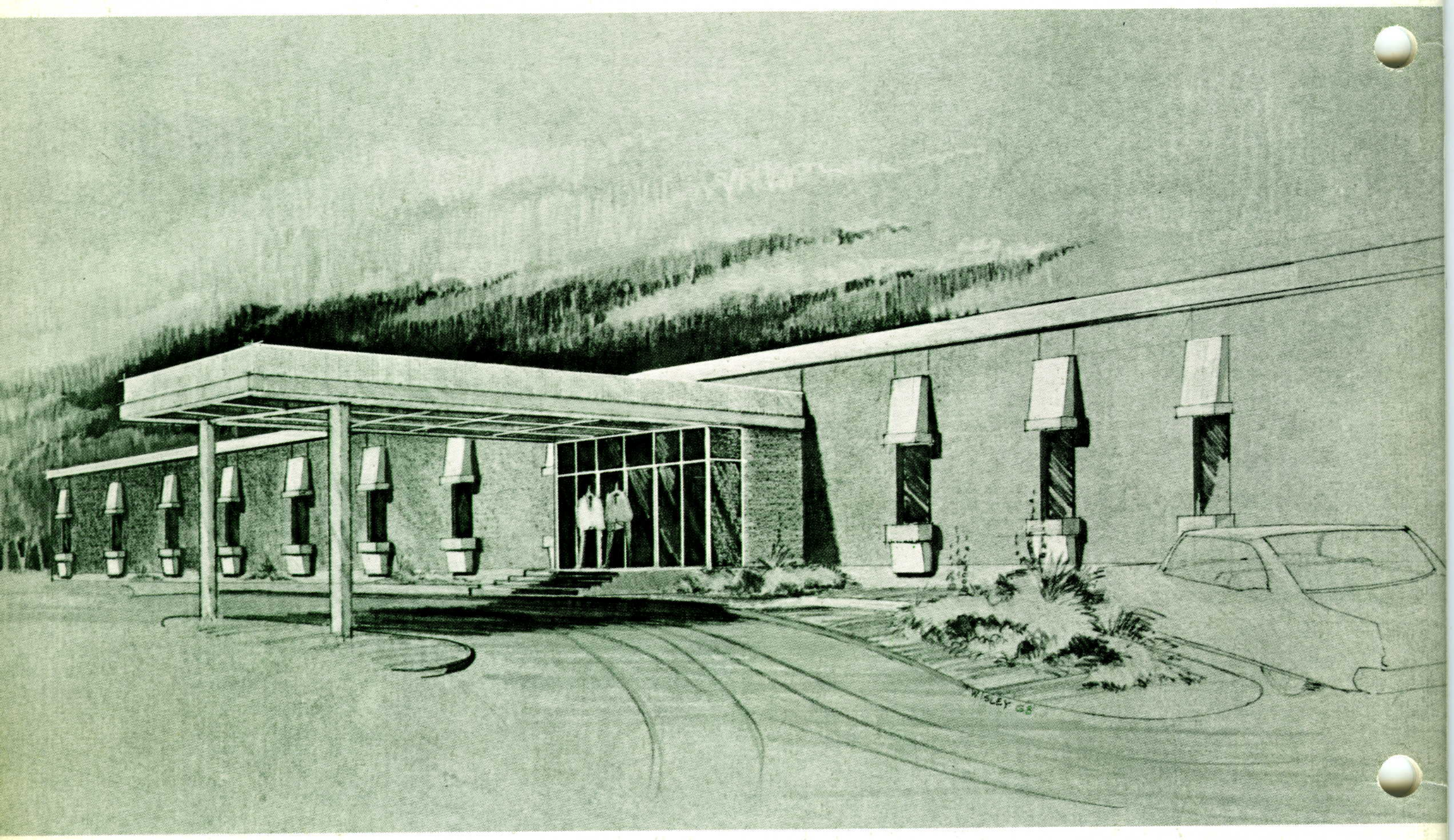
Memory Mapping
for Multiprogramming
Applications

Memory Parity
Error Detection

Power Failure
Protection

Hardware
· Multiply-Divide
· Double Precision
· Floating Point

Some of the above features are optional, depending upon application requirements for computer speed and capability.



EASTERN REGION

7100 Baltimore Ave., Suite 105
College Park, Maryland 20748
301-779-2510

Holiday Office Center, Suite 96
Huntsville, Alabama 35801
205-881-8805

1222 Route 46, Suite 217
Parsippany, New Jersey 07054
201-335-3001

2024 Riverdale Street
West Springfield, Massachusetts 01089
413-781-0063

CENTRAL REGION

612 Exchange Bank Building
Dallas, Texas 75235
214-358-1331

International Trades Building
400 Brookes Drive, Room 125
Hazelwood, Missouri 63042
314-864-6291

WESTERN REGION

780 Welch Road, Suite 208
Palo Alto, California 94304
415-328-8980

306 Hirschmann Bldg.
9550 Flair Drive
El Monte, California 91731
213-443-0143

SCC

Scientific Control Corporation

P.O. Box 34529 Dallas, Texas 75234 214-241-2111 TWX 910-860-5509

1026 66813

SCC 43-768