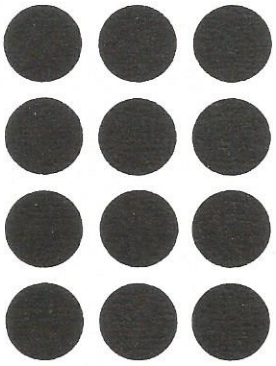
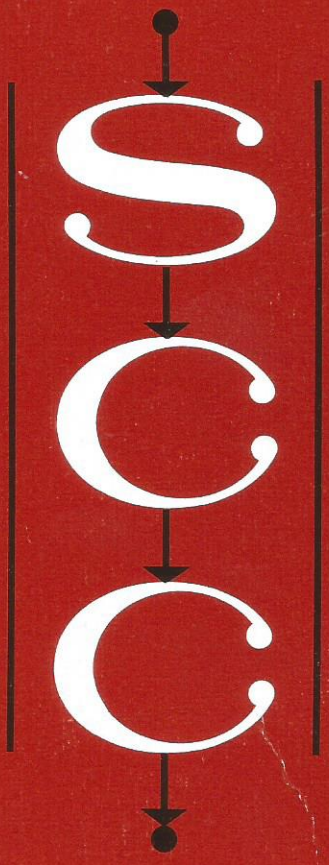
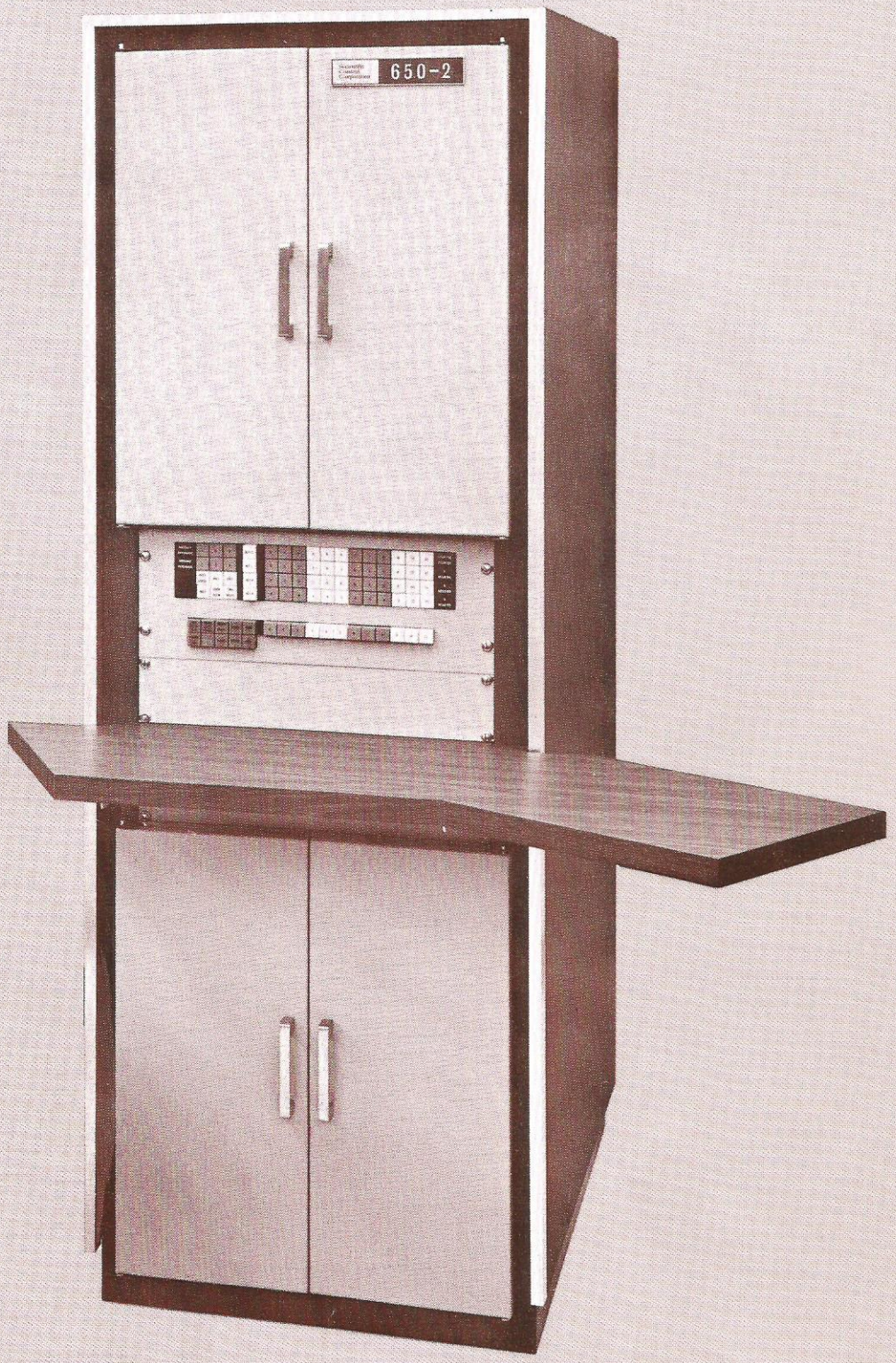
Scientific Control Corporation

Per cop. Ripoboa

Digital Computer
SCC 650






650-2

INTRODUCTION

The SCC 650 is a powerful twelve bit binary computer which offers many features found only in more expensive equipment: a high speed magnetic core memory with a cycle time of 1.75 microseconds, a versatile and complete instruction repertoire to simplify programming, an input/output interface which permits the use of an unlimited variety of peripheral equipment, and an understandable design for ease in use and maintenance.

In the SCC 650, special emphasis has been placed upon providing a computer system which can adapt to changing needs in the field.

A comprehensive software package is provided with each hardware system. It is designed to take full advantage of the hardware capabilities of the SCC 650 while providing the user with a convenient means of communication with the hardware system.

FEATURES

Large Memory Capacity

The basic memory module of the SCC 650 contains 4,096 twelve-bit words. The memory capability is expandable to 32,768 words in 4,096 word increments. In keeping with the corporate modular design philosophy, memory expansion may be made in the field.

All Silicon Semiconductors

All circuits utilize reliable, silicon semiconductors mounted on printed circuit plug-in cards. The system is highly maintainable at any modular level.

Fully Parallel Operation

All arithmetic and logical operations, together with high speed data transfers between memory and the various registers, are completed in parallel.

Hardware Index Register

In addition to providing address modification for memory reference instructions, the index register is a valuable aid in logical operation. A special set of instructions provide logical and arithmetic operations which use the index register and the accumulator as operands.

Priority Interrupt Channels

The interrupt channel, included as standard equipment, may be activated externally by an external device or internally by a memory protect error.
A priority facility is incorporated to permit external devices to command the computer to transfer to a specific subroutine. Using this facility priority channels can be added which are fully nested. That is, a higher priority channel may interrupt lower priority interrupt routines and the lower priority routine will be resumed after completion of the higher priority function.

Memory Protect Feature

Hardware memory protection is provided for lower numbered memory positions. The protect feature allows the reading of instructions or data from the protected area but inhibits any writing into the protected area of memory. Attempting to write into the protected area causes a program interrupt, allowing the programmer to take corrective action as required. The memory protect feature may be enabled or disabled from a switch on the control console.

Direct Memory Access Feature

The direct memory access feature permits high speed data transfer between memory and external devices. This access channel operates on a cycle stealing basis. As a result, the external device is not required to wait until the present instruction is complete before control of the memory is transferred to the external device. Complete isolation between memory banks allows the computer to continue undisturbed if the external device requests a memory bank other than the one being used by the processor. Also, memory overlap can occur when two or more external devices request different memory modules simultaneously. Therefore, each of eight different devices can transfer data simultaneously at a rate of 570,000 twelve-bit words per second for a total rate of 4,560,000 words/second.

Micro-Programmable Instructions

All instructions which do not reference memory may be micro-programmed. This group of instructions allows the computer to execute up to four non-memory instructions in a single computer cycle. It also provides the programmer with a variety of bit-manipulation, shift, skip, test, and I/0 instructions.

Flexible Subroutine Linkages

A single instruction executes a transfer to a subroutine in any core bank and provides the mechanism for a single instruction return to the calling program.

Four Addressing Modes

The addressing techniques employed in the SCC 650 computer permit several modes of addressing — indirect, direct, relative, indexed, or indirect indexed.

Functional Control Console

The control console is designed as a functional man-machine interface. The addressable registers are continuously displayed on the control console. These register displays are arranged for quick operator recognition of their contents. Special switches are provided for a fast, convenient means of entering information into memory or examining the contents of memory.

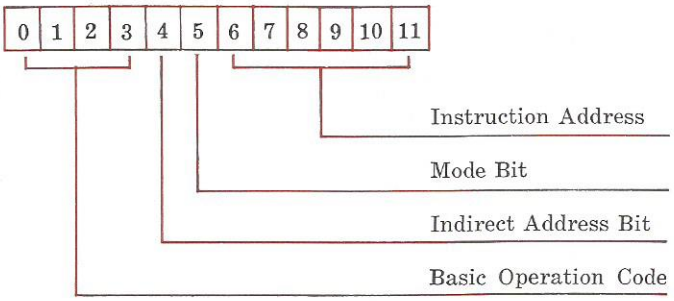
The computer may be operated either in run or single-step modes.

Literal Instructions

Several literal instructions are incorporated which use the last six bits of the instruction as the operand. This feature allows high speed, one character arithmetic or logical operations.

BASIC INSTRUCTION FORMAT

The basic instruction format is as follows:



In addition to the mode specification bit in the basic instruction, an index state or an indirect index state may be initiated through the status register. When the index state is in effect, the index register is used in lieu of the P counter for address modification.

ADDRESSING MODES

Primary Address

Each of the four possible addressing modes is based on the concept of a "primary address." The primary address of an instruction is defined to be any address which can be formed using the instruction address and the available address modification registers.

Direct Mode . . . Mode Bit = 0

Primary address is determined by the instruction address. In the direct mode, the primary address always refers to the first 64 locations of memory bank zero.

Index Mode

Primary address is the sum of the instruction address and the contents of the index register. It will always be in the same memory bank as the instruction.

Relative Mode

Primary address is the sum of the instruction address and the address from which the instruction was taken. It will always be in the same memory bank as the instruction. If the primary address is the instruction address plus one, the location counter skips the next word in the instruction sequence. In this way both operands and full addresses may be included "in-line."

Indirect Mode

The indirect address bit is always applied after the contents of the primary address has been obtained. If the indirect address bit is zero, the contents of the primary address is used as the operand of the instruction. If the indirect address bit is a one, the contents of the primary address is interpreted not as an operand but as the 12 low-order bits of an operand address. The 3 high-order bits of the operand address are specified by the indirect bank register.

Indirect Indexed Mode

In this mode, after the contents of the primary address are obtained, the contents of the index register are added to form the operand address. As in indirect mode, the 3 high-order bits of the operand address are specified by the indirect bank register.

INSTRUCTIONS

The instruction repertoire of the SCC 650 is described below. 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) The letter "Q" denotes the effective address, i.e., Q denotes the memory address used in the execution of the instruction.

Arithmetic Instructions

Instruction	Mnemonic	Cycles	Op Code
Add to Accumulator (A) + (Q) → A (Q) → Q	ADD	2	1001
Subtract from Accumulator (A) − (Q) → A (Q) → Q	SUB	2	1101
Exclusive OR with Accumulator (A) ⊕ (Q) → A (Q) → Q	XOR	2	1011

AND with Accumulator (A) • (Q) → A (Q) → Q	AND	2	1111

Memory Increment and Skip on Zero
(Q) + 1 → Q
(A) → A
If (Q) = 0, (P) + 2 → P
If (Q) ≠ 0, (P) + 1 → P
	MIN	3	1000

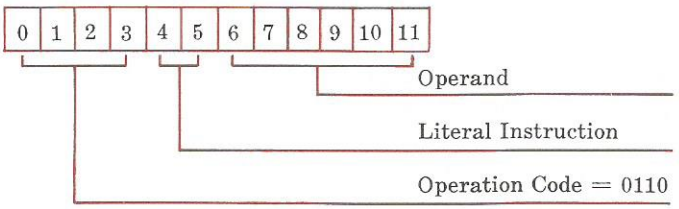
Load and Store Instructions

Instruction	Mnemonic	Cycles	Op Code
Load Accumulator (Q) → A (Q) → Q	LDA	2	1100
Store Accumulator (A) → Q (A) → A	STA	2	0111
Load Index (Q) → X (Q) → Q	LDX	2	0010
Store Index (X) → Q (X) → X	STX	2	0011

Jump Instructions

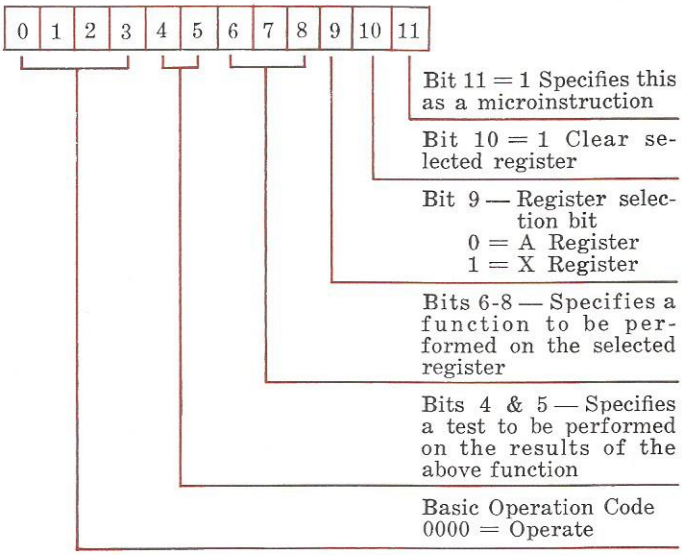
Instruction	Mnemonic	Cycles	Op Codes
Jump Forward Q → P (A) → A Note: If Bit 4 = 0, Q = (P) + (INS)$_{6-11}$	JMF	1	0100
Jump Backward Q → P (A) → A Note: If Bit 4 = 0, Q = (P) - (INS)$_{6-11}$	JMB	1	0101
Jump and Store Location (P) + 1 → (Q) Status → Q + 1 Q + 2 → P (A) → A	JSL	3	1010
Return Jump (Q) → P (Q + 1) → Status (A) → A	JRT	3	1110

Literal Instruction Format



R_4	R_5	Instruction	Mnemonic
0	0	AND Literal	ANL
0	1	EXCLUSIVE OR Literal	XOL
1	0	LOAD A Literal	LDL
1	1	ADD Literal	ADL

Instruction	Mnemonic	Cycles	R_4	R_5
AND Literal (INS)$_{6-11}$ • (A) → A	ANL	1	0	0
Exclusive OR, Literal (INS)$_{6-11}$ ⊕ (A) → A	XOL	1	0	1
Load A, Literal (INS)$_{6-11}$ → A	LDL	1	1	0
Add, Literal (INS)$_{6-11}$ + (A) → A	ADL	1	1	1



First Microinstruction

Select the register defined by Bit 9

0 = A Register

1 = X Register

Second Microinstruction

Perform one of eight functions as specified by Bits 6-8.

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 & X (A) ⊕ (X) → Sel Reg

100	One's complement of Selected Register $(\overline{\text{Sel Reg}}) \rightarrow$ Sel Reg
101	Two's complement of Selected Register $(\overline{\text{Sel Reg}}) + 1 \rightarrow$ Sel Reg
110	Complement Selected Register and Exclusive OR with Unselected Register $\overline{(A) \oplus (X)} \rightarrow$ Sel Reg
111	Subtract Selected Register from Unselected Register (Unsel Reg) – (Sel Reg) → Sel Reg

Third Microinstruction

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

R_4	R_5	
0	0	No Skip
0	1	Skip if positive (>0)
1	0	Skip if negative
1	1	Skip if Zero

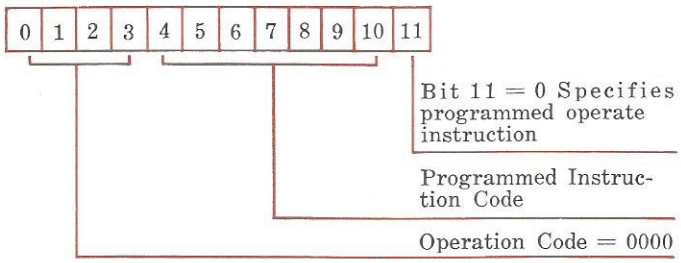
Fourth Microinstruction

Clear selected register if specified by Bit 10.
If bit 10 = 0, do not clear selected register.

Commonly Used Microinstructions — One Cycle

Instruction Code	Description
0003	Clear A Register
0007	Clear X Register
0101	Test A Register and skip if positive
0201	Test A Register and skip if negative
0301	Test A Register and skip if zero
0105	Test X Register and skip if positive
0205	Test X Register and skip if negative
0305	Test X Register and skip if zero
0011	Increment A and no skip
0111	Increment A and skip if positive
0211	Increment A and skip if negative
0311	Increment A and skip if zero
0015	Increment X and no skip
0115	Increment X and skip if positive
0215	Increment X and skip if negative
0315	Increment X and skip if zero
0021	Add A and X, results into A Register, No Skip
0121	Add A and X, results into A Register—Skip if positive
0221	Add A and X, results into A Register—Skip if negative
0321	Add A and X, results into A Register—Skip if zero
0025	Add A and X, results into X Register—No skip
0125	Add A and X, results into X Register—Skip if positive
0225	Add A and X, results into X Register—Skip if negative
0325	Add A and X, results into X Register—Skip if zero
0031	Exclusive OR A with X, Results into A Register—no skip
0131	Exclusive OR A with X, Results into A Register—Skip if positive
0231	Exclusive OR A with X, Results into A Register—Skip if negative
0331	Exclusive OR A with X, Results into A Register—Skip if zero
0035	Exclusive OR A with X, Results into X Register—No Skip
0135	Exclusive OR A with X, Results into X Register—Skip if positive
0235	Exclusive OR A with X, Results into X Register—Skip if negative
0335	Exclusive OR A with X, Results into X Register—Skip if zero
0041	One's complement of A — No skip
0141	One's complement of A — Skip if positive
0241	One's complement of A — Skip if negative
0341	One's complement of A — Skip if zero
0045	One's complement of X — No skip
0145	One's complement of X — Skip if positive
0245	One's complement of X — Skip if negative
0345	One's complement of X — Skip if zero
0051	Two's complement of A — No skip
0151	Two's complement of A — Skip if positive
0251	Two's complement of A — Skip if negative
0351	Two's complement of A — Skip if zero
0055	Two's complement of X — No skip
0155	Two's complement of X — Skip if positive
0255	Two's complement of X — Skip if negative
0355	Two's complement of X — Skip if zero
0061	Complement A and Exclusive OR with X, Place result in A register — No Skip
0161	Complement A and Exclusive OR with X, Place result in A register —Skip if positive
0261	Complement A and Exclusive OR with X, Place result in A register —Skip if negative
0361	Complement A and Exclusive OR with X, Place result in A register — Skip if zero
0065	Complement X and Exclusive OR with A, Place result in X register — No skip
0165	Complement X and Exclusive OR with A, Place result in X register —Skip if positive
0265	Complement X and Exclusive OR with A, Place result in X register —Skip if negative
0365	Complement X and Exclusive OR with A, Place result in X register — Skip if zero
0071	Subtract A from X, Place result in A register — No skip
0171	Subtract A from X, Place result in A register — Skip if positive
0271	Subtract A from X, Place result in A register—Skip if negative
0371	Subtract A from X, Place result in A register—Skip if zero
0075	Subtract X from A, Place result in X Register — No skip
0175	Subtract X from A, Place result in X Register—Skip if positive
0275	Subtract X from A, Place result in X Register—Skip if negative
0375	Subtract X from A, Place result in X Register—Skip if zero

Operate Instruction Format (Programmed)

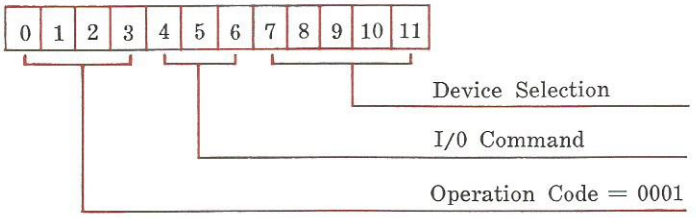


Instruction	Mnemonic	Instruction Code	Cycles
Short Arithmetic Right Shift $(A)_i \rightarrow A_{i+1}$ for $0 \leq i < 11$ $(A)_0 \rightarrow A_0$	SAR	0014	1
Short Arithmetic Left Shift $(A)_i \rightarrow A_{i-1}$ for $0 < i \leq 11$ $0 \rightarrow A_{11}$	SAL	0016	1
Long Arithmetic Right Shift $(A)_i \rightarrow A_{i+1}$, $(X)_i \rightarrow X_{i+1}$, for $0 \leq i \leq 11$ $(A)_0 \rightarrow A_0$ $(A)_{11} \rightarrow X_0$	LAR	0214	1
Long Arithmetic Left Shift $(A)_i \rightarrow A_{i-1}$, $(X)_i \rightarrow X_{i-1}$, $0 < i \leq 11$ $(X)_0 \rightarrow A_{11}$ $0 \rightarrow X_{11}$	LAL	0216	1
Short Logical Right Shift $(A)_i \rightarrow A_{i+1}$, $0 \leq i < 11$ $0 \rightarrow A_0$	SLR	0114	1
Short Logical Left Shift $(A)_i \rightarrow A_{i-1}$, $0 < i \leq 11$ $0 \rightarrow A_{11}$	SLL	0116	1
Long Logical Right Shift $(A)_i \rightarrow A_{i+1}$, $(X)_i \rightarrow X_{i+1}$, $0 \leq i < 11$ $0 \rightarrow A_0$ $(A)_{11} \rightarrow X_0$	LLR	0314	1
Long Logical Left Shift $(A)_i \rightarrow A_{i-1}$, $(X)_i \rightarrow X_{i-1}$, $0 < i \leq 11$ $(X)_0 \rightarrow A_{11}$ $0 \rightarrow X_{11}$	LLL	0316	1
Short Rotate Right $(A)_i \rightarrow A_{i+1}$ for $0 \leq i < 11$ $(A)_{11} \rightarrow A_0$	SRR	0010	1
Short Rotate Left $(A)_i \rightarrow A_{i-1}$ for $0 < i \leq 11$ $(A)_0 \rightarrow A_{11}$	SRL	0012	1
Long Rotate Right $(A)_i \rightarrow A_{i+1}$, $(X)_i \rightarrow X_{i+1}$ for $0 \leq i < 11$ $(X)_{11} \rightarrow A_0$ $(A)_{11} \rightarrow X_0$	LRR	0210	1
Long Rotate Left $(A)_i \rightarrow A_{i-1}$, $(X)_i \rightarrow X_{i-1}$, $0 < i \leq 11$ $(X)_0 \rightarrow A_{11}$ $(A)_0 \rightarrow X_{11}$	LRL	0212	1
Short Circulate Right $(A)_i \rightarrow A_{i+1}$, $0 \leq i < 11$ $(A)_{11} \rightarrow CO$ $(CO) \rightarrow A_0$	SCR	0110	1
Short Circulate Left $(A)_i \rightarrow A_{i-1}$, $0 < i \leq 11$ $(A)_0 \rightarrow CO$ $(CO) \rightarrow A_{11}$	SCL	0112	1
Long Circulate Right $(A)_i \rightarrow A_{i+1}$, $(X)_i \rightarrow X_{i+1}$, $0 \leq i < 11$ $(CO) \rightarrow A_0$ $(A)_{11} \rightarrow X_0$ $(X)_0 \rightarrow CO$	LCR	0310	1
Long Circulate Left $(A)_i \rightarrow A_{i-1}$, $(X)_i \rightarrow X_{i-1}$, $0 < i \leq 11$ $(CO) \rightarrow X_{11}$ $(X)_0 \rightarrow A_{11}$ $(A)_0 \rightarrow CO$	LCL	0312	1
Copy Accumulator into Index $(A) \rightarrow X$	CAX	0240	1
Copy Index into Accumulator $(X) \rightarrow A$	CXA	0140	1
Exchange Accumulator and Index $(A) \rightarrow X$ $(X) \rightarrow A$	XAX	0040	1
Exchange Halves of Accumulator $(A)_{0-5} \rightarrow A_{6-11}$ $(A)_{6-11} \rightarrow A_{0-5}$	XHA	0020	1
Halt	HLT	0000	1
No Operation $(P) + 1 \rightarrow P$	NOP	0002	1
Set Carry On $1 \rightarrow CO$	SCN	0122	1
Set Carry Off $0 \rightarrow CO$	SCF	0022	1
Enable Interrupts Enable Interrupt System	ENA	0322	1
Disable Interrupts Disable Interrupt System	DIS	0222	1
Clear Interrupt Clear Current Priority Interrupt	CLI	0226	1
Index State Set $1 \rightarrow$ Index State	XSS	0234	1
Index State Reset $0 \rightarrow$ Index State	XSR	0134	1

Instruction	Mnemonic	Instruction Code	Cycles
Carry Out Test (P) + 1 → P if CO = 1 (P) + 2 → P if CO = 0	COT	0032	1
Overflow Test (P) + 1 → P, 0 → OV; if OV = 1 (P) + 2 → P, if OV = 0	OFT	0024	1
Memory Protect Error Test (P) + 1 → P, 0 → MPE; if MPE = 1 (P) + 2 → P; if MPE = 0	MPT	0044	1
Input/Output Error Test (P) + 1 → P, 0 → IOE; if IOE = 1 (P) + 2 → P; if IOE = 0	IOT	0046	1
Load IB From Accumulator $(A)_{9-11} \rightarrow IB$	LDI	0036	1
And Accumulator with Index (A) · (X) → A	AAX	0242	1
Load Status Register (A) → Status	LDS	0260	1
Load Accumulator from Switches (Switches) → A	LAS	0050	1
Store Status Register (Status) → A	STS	0060	1
Load IB from Memory* $[(P) + 1]_{9-11} \rightarrow IB$ (P) + 2 → P	LDM	0236	1
Add Carry (A) + (CO) → A Carry → CO 1 → OV, if overflow	ADC	0054	1
OR Accumulator with Index (A) + (X) → A	AOX	0042	1
Load Double-Length** (Q) → A, (Q + 1) → X	LDD	0276	3
Store Double-Length** (A) → Q, (X) → Q + 1	STD	0376	3
Multiply** (A) × (Q) → A, X	MPY	0074 0274	9
Divide** (A, X) / (Q) → A, Remainder → X	DIV	0174 0374	9
Normalize**	NOR	0072	1 + Count
Store Shift Register** $0 \rightarrow A_{0-5}$ (Shift) → A_{6-11}	SSH	0066	1
Load Shift Register** $(A_{0-5}) \rightarrow A_{0-5}$ $(A_{6-11}) \rightarrow$ Shift	LSH	0266	1
Repeat** Repeat following instruction	RPT	0070	1

*Available with larger memories
**Available with extended arithmetic unit

Input-Output Instruction Format

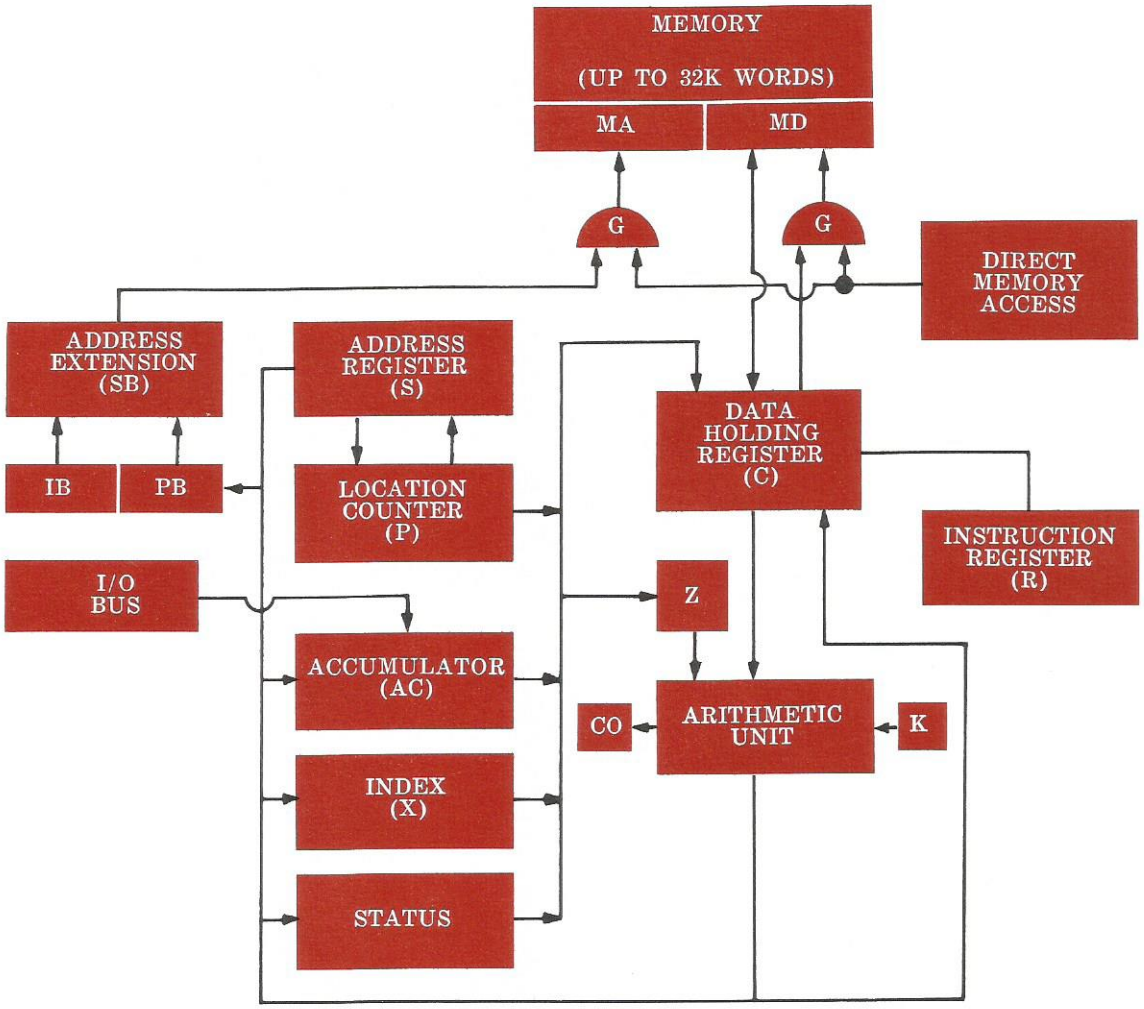


I/O	Commands	Mnemonic	Cycles
000	Transmit to A & Skip (Selected device) → A, (P) + 1 → P; if device is ready (P) + 2 → P; if device is not ready	TTA	1
001	Transmit from A & Skip (A) → Selected device, (P) + 1 → P; if device is ready (P) + 2 → P; if device is not ready	TFA	1
010	Input Device Status (Device status) → A, (P) + 1 → P; if device is ready (P) + 2 → P; if device is not ready	DST	1
011	Skip on device flag FLAG = 0 (P) + 2 → P FLAG = 1 (P) + 1 → P	SDF	1
100	Execute Command in A External device executes command in A	EXU	1
101	Terminate Inactivate device	TMR	1
110	Select Device	SEL	1
111	I/O control Select Device	IOC	1

INTERNAL MACHINE ORGANIZATION

Figure 1 shows a block diagram of the internal operation of the SCC 650 computer. In this illustration, all arrows indicate data paths and all transfers over these data paths are parallel transfers.

As can be seen from this illustration, the computer contains seven twelve-bit registers, three three-bit registers, three one-bit registers, a parallel adder, and miscellaneous gates and control registers. The operation of these registers is as follows:



DATA PATHS

Figure 1

C Register — The C Register accepts data from the memory and is used as a holding register into the parallel adder. In the halt mode, this register contains the instruction which will be executed next. Execution of the instruction contained in the C Register causes the transfer of the instruction into the R Register which is used to hold the instruction throughout its execution time.

Z Register — The Z Register is the second holding register into the parallel adder. The parallel adder uses as its input the output of both the C Register and the Z Register.

X Register — The X Register is a twelve-bit register which can be used for indexing and addressing modification. Microinstructions use the X and the A registers for internal manipulation.

The X Register also serves as the extended accumulator for multiply and divide operations.

A Register — The A Register is the computer's main accumulator. All arithmetic and logical operations performed within the computer use this register to hold the results of the operation.

R Register — The R Register holds instructions for decoding into various instruction signals.

P Register — The P Register is a twelve-bit register which is used as the location counter for the computer. When in the relative mode, the six-bit instruction address is added to the contents of the P counter to determine the effective address of the instruction to be executed.

S Register — The S Register is the memory address register consisting of twelve bits.

PB Register — The PB Register is the instruction extension register used to determine the memory bank from which instructions and all relative operands will be taken.

IB Register — The IB Register is the indirect extension register used to specify a particular memory bank for indirectly addressed operands.

SB Register — The SB Register is a three-bit register which is decoded and used to specify which memory banks will be used for a particular memory operation.

Adder — The twelve-bit parallel adder is capable of arithmetic operations (add and subtract) as well as logical operations such as AND, OR, and EXCLUSIVE OR.

K Register — The K Register is a one-bit register used to initiate a carry into the adder. This register is used for two's complement operation and for incrementing registers.

CO Register — The CO Register is a one-bit register which is set when the output of the adder has a carry-out.

Overflow Register — The Overflow Register is a one-bit register which is set to "one" when the adder has an overflow.

Status Register — The Status Register is a twelve-bit composite register consisting of:

(1) The PB register (3 bits)
(2) The IB register (3 bits)
(3) The CO register (1 bit)
(4) The Overflow register (1 bit)
(5) The Memory Protect register (1 bit)
(6) The Index State (1 bit)
(7) The Indirect Indexing State (1 bit)
(8) The I/O Error register (1 bit)

G Gates — The G Gates serve as a buffer between the central processing unit and memory. When memory is being directly addressed by external devices, the G Gates disconnect memory from the CPU allowing transfer between memory and the external device to take priority. During the time that the external device is using memory, the CPU enters an idle state for one cycle.

Memory — Memory for the computer is broken into eight 4,096 word banks. Memory isolation is provided between the CPU and each bank of memory allowing overlap between memory operations of the CPU and other external devices. If an external device requests a memory bank other than the one being requested by the CPU, both requests are honored simultaneously.

Switch Register — A switch register consisting of twelve switches is located on the front panel of the computer. Data can be transferred from the twelve switches into the accumulator under program control. Also, data from these switches may be transferred to any of the active registers during a halt mode operation. In the halt mode, the store switch may be used to store the data contained in the switch register in the memory location specified by the P counter.

INPUT/OUTPUT FACILITY

Communications with the SCC 650 computer can be accomplished by any of three basic means:

Programmed Data Transfer

This facility allows the computer to control the input/output sequence by initiating proper I/O instructions and testing to see if the external device is in a ready state. If the external device is ready, then the computer will initiate a data transfer and skip the next instruction. Programmed data transfer can be initiated by the programmer or by the external device using a program interrupt. Upon receipt of an interrupt from the external device, the computer will trap to the particular subroutine which is programmed to service the interrupting device.

Direct Memory Access

Individual data words or blocks of data can be entered into or taken from the computer memory directly through the direct memory access system. When operating in this mode, the computer will enter an idling state until the external device has completed the data transfer. If the external device is using a memory bank other than the one being requested by the computer, the computer may proceed without intervention. In this case, memory overlap occurs and the external device and the central processing unit are using different banks of memory simultaneously. After initiating a request for memory, the external device does not have to wait until the end of a current instruction to initiate a transfer. The maximum length of waiting time for such a transfer is one computer cycle or two microseconds.

Skip Testing

The computer contains a facility to allow the CPU under program control to request the status of external equipment and, based on the contents of that status, either skip or execute the following instruction. This feature allows the computer to make decisions during the program based on the condition of many external devices.

OPTIONAL EQUIPMENT

Extended Arithmetic Elements

This option allows the computer to perform a hardware multiply and divide using the X Register as an extended accumulator. This option also allows for added instructional capabilities for normalizing and scaling of data.

Memory Extension Control

The basic computer can be expanded to include eight banks of memory by using the memory extension control. The control provides all the logic elements necessary for the addition of seven memory banks.

Memory Modules

The basic memory module is a high speed, random access memory with a 1.75 microsecond cycle time consisting of 4,096 twelve bit words.

High Speed Tape Reader

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

Paper Tape Punch

Two 8-channel models are available for punching at either 50 or 120 CPS.

Card Reader

Punched card readers are available for reading 200 cards or 400 cards per minute.

Card Punch

The card punch operates at a punching rate of 100 cards per minute.

Incremental Plotter

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

Line Printer

This machine prints a selection of 64 characters at the rate of 300 lines per minute and with a 132 character per column format. The vertical format is selected by a punched paper tape within the printer.

Magnetic Tape Controller

The magnetic tape controller can control up to four magnetic tape decks simultaneously. Each magnetic tape can be controlled to read or write in densities of 200, 556, and 800 bpi and at speeds of from 75 ips to 120 ips.

Magnetic Tape Transports

The magnetic tape transports will write or read IBM compatible magnetic tape at transfer rates of 15 KC to 96 KC.

Magnetic Disc

A magnetic disc unit is available which provides memory in 3 million bit modules with high speed access.

Interrupt Clock or Real Time Clock

A clock is available which generates an interrupt within the computer after a preset period of time has elapsed interrupt. Another clock is available for real time input.

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.)

Pulse Height Adapter

A module is available which in conjunction with an A/D converter will automatically count pulse heights into memory locations.

Remote Teletype Unit

This option allows the Teletype I/O writer to be remotely located from the central processing unit. (Either 5 or 8 level codes available.)

Selectric Typewriter

An IBM Selectric Typewriter with a maximum speed of 15.5 characters per second is available. The typewriter is equipped with a pin-feed platen allowing the use of continuous form paper.

CRT Display

The CRT Display presents X-Y plots of recorded and processed signals for evaluation by the system operator. The system includes two digital to analog converters.

Analog Multiplexer

This option allows for the sequential sampling of 32 analog channels of 0 to 5 volts DC. The multiplexer contains a buffer amplifier and all the interface necessary for operation with the A-D Converter. The number of channels can vary in groups of two up to the maximum of 32. The computer can address the multiplexer on a random access basis or program the multiplexer for sampling the analog channels sequentially.

Analog to Digital Converter

The Analog to Digital Converter digitizes the output of the analog multiplexer to an eleven bit plus sign digital data word. Digitizing rates up to 50 KC can be provided. A sample and hold amplifier is included with the analog to digital converter.

Digital to Analog Converter

The SCC digital to analog converter is an addressable multichannel, binary, two's complement converter. Unit may be bipolar or unipolar with resolutions from 9 to 12 bits.

MECHANICAL SPECIFICATIONS

Standard Cabinet Model

The cabinet model is rack mounted in a standard RETMA 19 inch relay rack, 6 feet in height. The teletype unit is contained on its own stand with the cabinet model. Total weight of the cabinet is 500 pounds. Weight of the Teletype with stand is 40 pounds.

Power Requirements

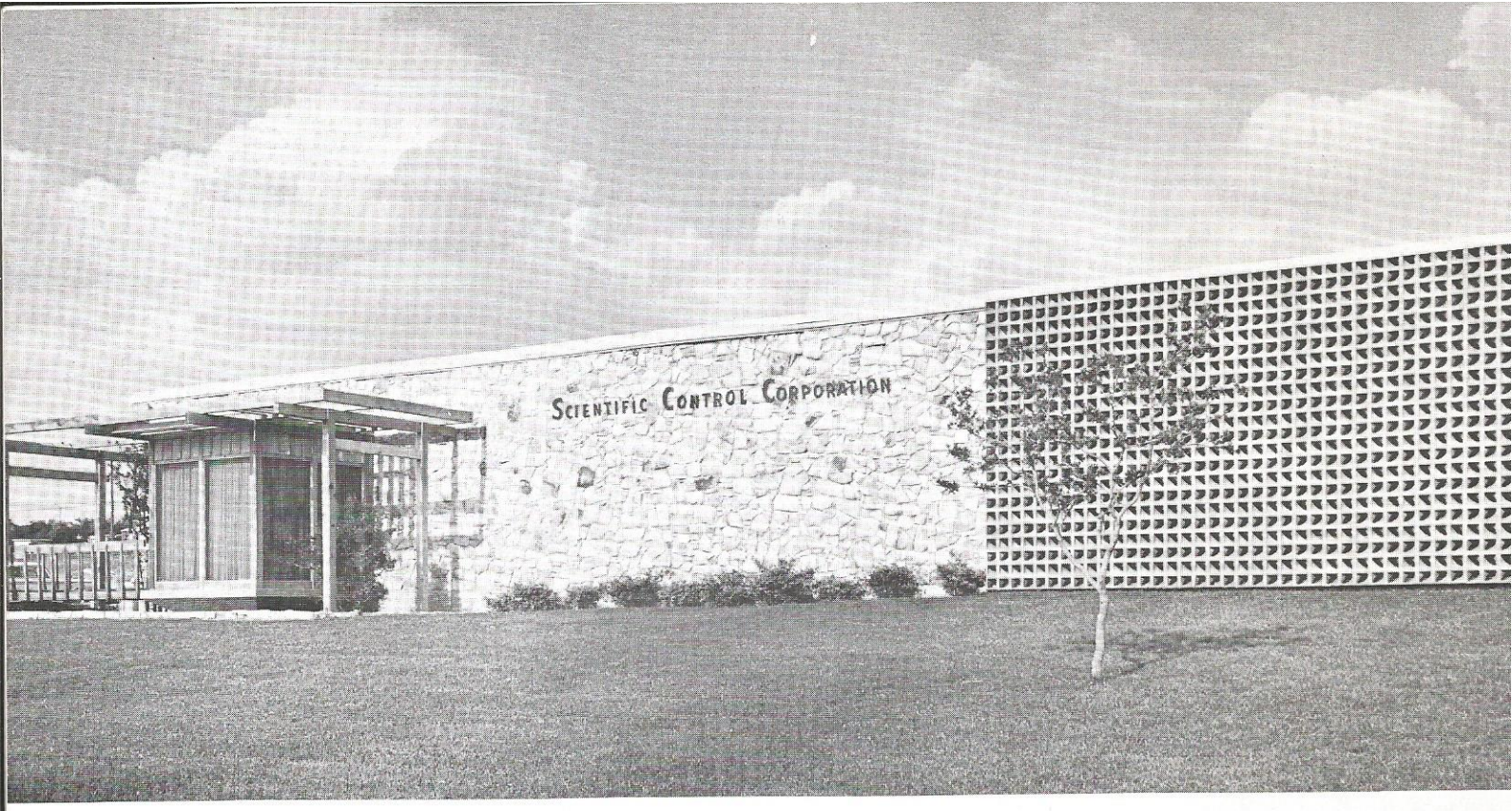
115 volts, 60 cycle, single phase, at 30 amps. 220 volts can be accommodated on special order.

Digital Signal Levels

0 Volts — False, + 8 Volts — True. (Internal)
0 Volts — True + 8 Volts — False. (Interface)

SOFTWARE

The SCC 650 includes a symbolic assembler, FORTRAN compiler, utility and math subroutines, and diagnostic routines. Other software is available from a library of programs. Custom programming is available through qualified personnel who are capable of obtaining maximum use and flexibility from the SCC 650.



SCC maintains complete support activities for its users. Installation and maintenance services are available through SCC offices strategically located throughout the United States. For pre-procurement demonstration of hardware and programs in Dallas, contact local sales office or the Marketing Department in Dallas.

BARNHILL ASSOCIATES

1170 So. Sheridan Blvd.
Denver, Colorado 80226
Phone: 303 934-5505

6101 Anderson St., S.E.
Albuquerque, N.M. 87108
Phone: 505 265-7766

3810 Westheimer, Suite 108
Houston, Texas
Phone: 713 NA 1-0040

506 N. Central Expressway
Richardson, Texas 75080
Phone: 214 AD 1-2573

2721 First Street
El Paso, Texas 79925
Phone: 915 598-2442

COZZENS AND CUDAHY, INC.

9501 W. Devon Avenue
Rosemont, Illinois 60018
Phone: 312 825-1144

3712 W. Elm Street
Milwaukee, Wis. 53209
Phone: 414 332-0700

7710 Computer Avenue
Minneapolis, Minn. 55435
Phone: 612 920-1022

1160 S. Florissant Road
Florissant, Missouri 63031
Phone: 314 831-0010

44 Kirk Drive
Indianapolis, Indiana 46234
Phone: 317 244-2456-7

MOXON ELECTRONICS CORPORATION

36 Central Avenue
Midland Park, New Jersey
Phone: 201 652-6750

40 Coles Avenue
Cherry Hill, New Jersey
Phone: 609 665-5445

2311 Pontius Avenue
Los Angeles, Calif. 90064
Phone: 213 272-9311

15 West 41st Avenue
San Mateo, Calif. 94403
Phone: 415 345-7961

#15 No. 40th Place
Phoenix, Arizona 85034
Phone: 602 273-7363

ELECTRONIC SALES ASSOCIATES, INC.

Village Green
Crofton, Maryland 21113
Phone: 301 723-0620

TECHNICAL ASSOCIATES

2319 East South Street
Orlando, Florida 32803
Phone: 305 424-5681

P. O. Box 1443
Huntsville, Alabama 35807
Phone: 205 881-8494

Scientific Control Corporation

P.O. Box 34529 • 14008 Distribution Way • Dallas, Texas 75234 • 214 — 241-2111