SINGLE-CHIP 8-BIT N-CHANNEL MICROPROCESSOR (SC/MP-II) ISP-8A/600

DESCRIPTION

SC/MP (Simple Cost-effective MicroProcessor) is a single-chip 8-bit microprocessor packaged in a standard, 40-pin, dual-in-line package.

N-channel, silicon gate, depletion mode standard-process technology ensures high performance, high reliability, and high producibility.

SC/MP is intended for use in general-purpose applications where cost per function is a most significant criterion. But cost efficiency is only a part of SC/MP's story. It goes on to include a variety of useful functions that are not even provided by some of the expensive microprocessors, like self-contained timing circuitry, 16-bit (65K) addressing capability, serial or parallel data-transfer capability and common memory/peripheral instructions. The built-in features in conjunction with the low initial cost describe what SC/MP really is—a microprocessor specifically designed to provide the simplest and most efficient solution to many application requirements.

APPLICATIONS

- **Test systems and instrumentation**
- **Machine tool control**
- **Small business machines**
- **Word processing systems**
- **Educational systems**
- **Multiprocessor systems**
- **Process controllers**
- **Terminals**
- **Traffic controls**
- **Laboratory controllers**
- **Sophisticated games**
- **Automotive**

ABSOLUTE MAXIMUM RATINGS*

PARAMETER		RATING	UNIT
	Voltage at any pin	-0.5 to +7.0	V
T_A	Operating temperature range	0 to +70	°C
T_{STG}	Storage temperature range	-65 to +150	°C
	Lead temperature (solder, 10 sec)	+300	°C

*NOTE

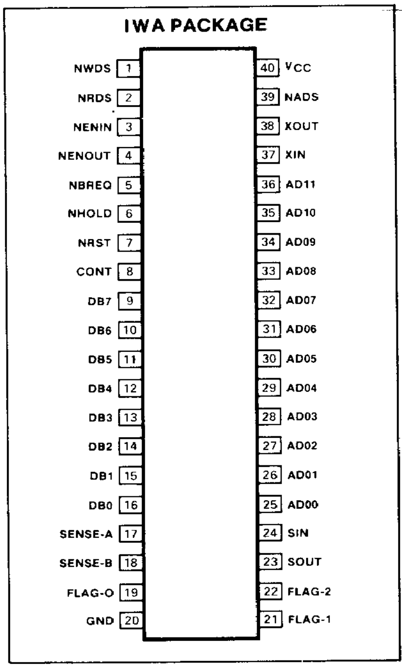
Maximum ratings indicate limits beyond which damage may occur. Continuous operation at these limits is not intended and should be limited to those conditions specified under electrical characteristics.

George Wong

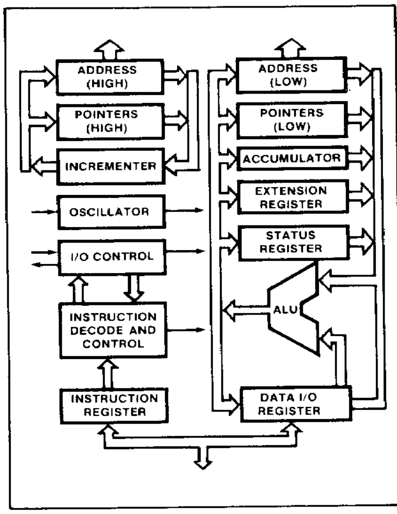
FEATURES

- **Simpler interfacing**
  - **Bidirectional Tri-state 8-bit data bus**
  - **TTL-compatible input/output interface**
- **Si-gate N-channel ion-implant process**
- **Direct Memory Access (DMA) and multiprocessor capabilities**
  - **Handshake bus-access control on chip**
- **Simplified programming**
  - **Multiple addressing modes—program-counter-relative, immediate data, indexed, auto-indexed, and implied**
- **Direct control output**
  - **Three user-accessible control-flag outputs**
- **Simpler I/O hardware**
  - **Separate serial-data input and output ports**
  - **Two sense inputs**
  - **Direct interfacing to standard memory parts**
- **Simplified timing hardware**
  - **On-chip clock generator**
- **Interface flexibility**
  - **Capability to interface with memories or peripherals of any speed**
- **Large system capability**
  - **Address capability to 65K bytes of memory**
- **Simplified power requirements**
  - **Single 5-volt supply**
  - **Low power**

PIN CONFIGURATION



BLOCK DIAGRAM



DC ELECTRICAL CHARACTERISTICS T_A = 0° C to +70° C, V_{CC} = +5V ±5%

PARAMETER	TEST CONDITIONS	LIMITS			UNIT
		Min	Typ	Max	
INPUT SPECIFICATIONS All input pins except V_{CC} and GND					
Logic "1" input		2.0		V_{CC}	V
Logic "0" input		-0.5		0.8	V
Capacitance (All pins except V_{CC} and GND)				10	pF
Supply current I_{CC}	T_A = 25°C outputs unloaded			45	mA
	T_A = 0°C outputs unloaded			50	mA
OUTPUT SPECIFICATIONS TRI-STATE pins (NWDS, NRDS, DB0-DB7, AD00-AD11)					
Logic "1" output	I_{OUT} = -100μA	2.4			V
Logic "0" output	I_{OUT} = 2.0mA			0.4	V
NADS, FLAG 0-2, SOUT, NENOUT					
Logic "1" output	I_{OUT} = -100μA	V_{CC} - 1			V
Logic "1" output	I_{OUT} = -1mA	1.5			V
Logic "0" output	I_{OUT} = 2.0mA			0.4	V
NBREQ[1]					
Logic "0" output	I_{OUT} = 2.0mA			0.4	V
Logic "1" output	$0 \leq V_{OUT} \leq V_{CC}$			±10	μA
XOUT					
Logic "1" output	I_{OUT} = -100μA	2.4			V
Logic "0" output	I_{OUT} = 1.6mA			0.4	V

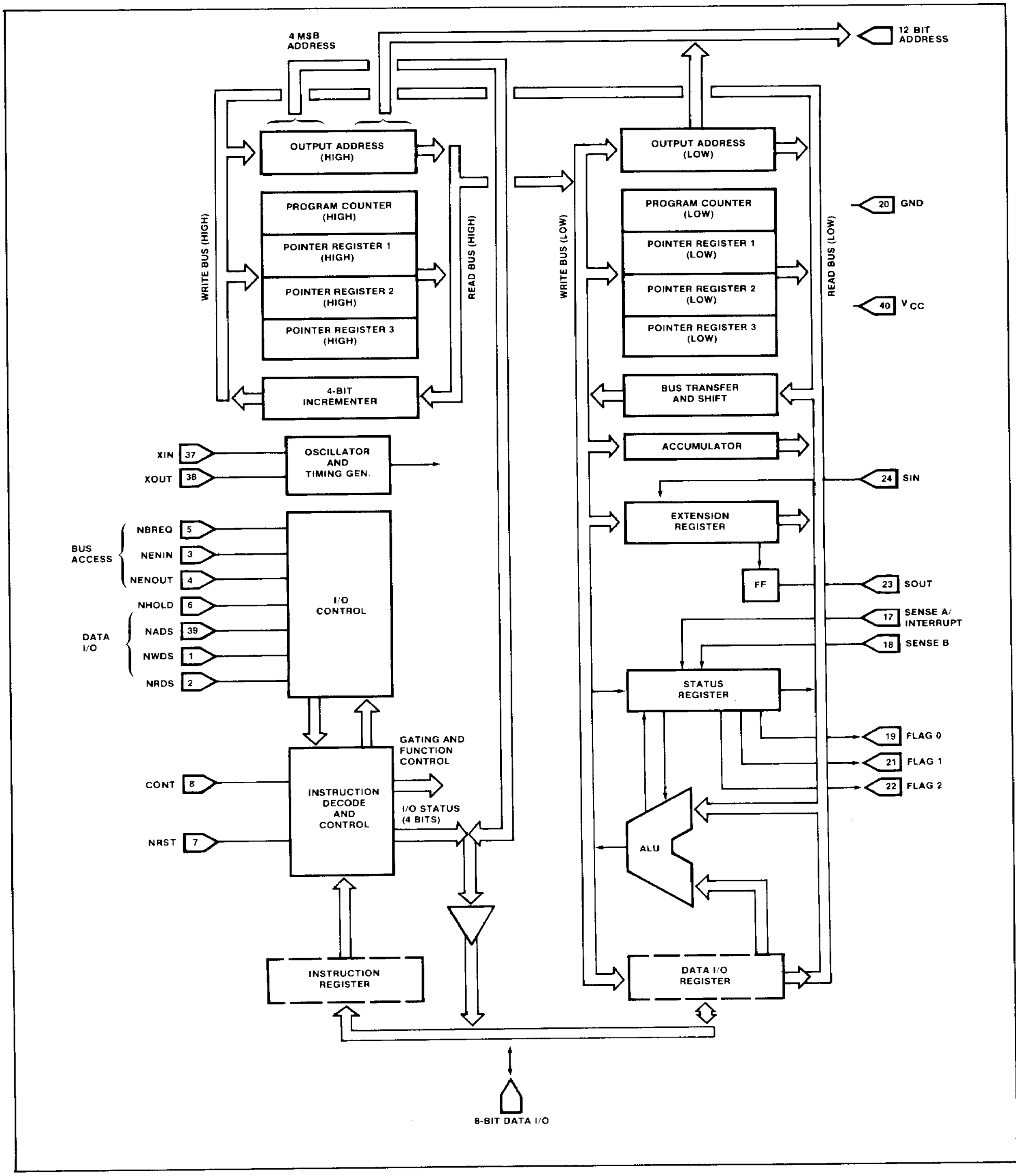
FUNCTIONAL DESCRIPTION

SC/MP is a self-contained general-purpose microprocessor designed for ease of implementation in stand-alone, DMA (Direct Memory Access), and multiprocessor applications. Communications between SC/MP and external memory/peripheral devices are effected via a 12-bit dedicated address bus and an 8-bit bidirectional data bus. During the address interval of each input/output cycle, SC/MP employs both buses to provide a 16-bit address output: the 12 least significant address bits are sent out over the 12-bit address bus and the 4 most significant address bits are sent out over the 8-bit data bus along with 4 status bits. Separate strobe outputs from SC/MP (NADS, NWDS, NRDS) indicate when valid address information is present on the two buses, and when valid input/output memory or peripheral data are present on the 8-bit bus. To further extend flexibility of application, serial data input/output ports are also provided so that serial data transfers can be effected under program control. The remaining input/output signals shown in the Detailed Block Diagram are dedicated to general-purpose control and status functions, including initialization, bus management, microprocessor halt, interrupt request, input/output cycle extension, and user-specified hardware/software interface functions. A detailed description of each input/output signal is provided in Table 1.

*NOTE

[1] NBREQ is an input/output signal that requires an external resistor to V_{CC}.

DETAILED BLOCK DIAGRAM



AC ELECTRICAL CHARACTERISTICS $T_A = 0°C$ to $+70°C$, $V_{CC} = +5V \pm 5\%$, 1 TTL Load.[1,3]

PARAMETER	TEST CONDITIONS	LIMITS			UNIT
		Min	Typ	Max	
f_X		0.1		4.0	MHz
	R = 240Ω ± 5% (Figure 2B)	2.0		4.0	MHz
	C = 300pF ±10%				
T_C[2]		500			ns
Microcycle		1			μs
External clock input	See Figure 2A				ns
T_{W0}		120			
T_{W1}		120			
XOUT/ADS timing relationship	See Figure 3				ns
T_H(ADS)		100		225	
Address and input/output status	See Figures 5 and 6				ns
T_{D1}(ADS)				$3T_C/2$	
T_W(ADS)		$(T_C/2) - 50$			
T_S(ADDR)		$(T_C/2) - 165$			
T_H(ADDR)		50			
T_S(STAT)		$(T_C/2) - 150$			
T_H(STAT)		50			
T_H(NBREQ)		0			
Data input cycle	See Figure 5				ns
T_D(RDS)		0			
T_W(RDS)		$T_C + 50$			
T_S(RD)		175			
T_H(RD)		0			
T_{ACC}(RD)		$2T_C - 200$			
Data output cycle	See Figure 6				ns
T_D(WDS)		$T_C - 50$			
T_W(WDS)		$T_C - 75$			
T_S(WD)		$(T_C/2) - 200$			
T_H(WD)		100			
Input/output cycle extend	See Figure 7				ns
T_S(HOLD)		200			
T_{D1}(HOLD)		130		275	
T_{D2}(HOLD)				350	
T_W(HOLD)				∞	
T_H(HOLD)		0			
Bus access	See Figure 4				ns
T_D(NENOUT)				150	
T_{D2}(ADS)		$T_C/2$		$3T_C/2$	
T_H(NENIN)		0			
Output load capacitance					pF
XOUT				30	
All other output pins				75	

NOTES

1. All times measured from valid Logic "0" level = 0.8V or valid Logic "1" level = 2.0V.
2. T_C is the time period for two clock cycles of the on-chip or external oscillator ($T_C = 2/f_X$). Refer to paragraph titled Timing Control for detailed definition.
3. All times measured with a 50% duty cycle on the external clock.

PIN DESIGNATIONS*

SIGNAL MNEMONIC	FUNCTIONAL NAME	DESCRIPTION
NRST	Reset input	Set high for normal operation. When set low, aborts in-process operations. When returned high, internal control circuit zeroes all programmer-accessible registers; then, first instruction is fetched from memory location 0001_{16}.
CONT	Continue input	When set high, enables normal execution of program stored in external memory. When set low, SC/MP operation is suspended (after completion of current instruction) without loss of internal status.
NBREQ	Bus request input/output	Associated with SC/MP internal allocation logic for system bus. Can be used as bus request output or bus busy input. Requires external load resistor to V_{CC}.
NENIN	Enable input	Associated with SC/MP internal allocation logic for system bus. When set low, SC/MP is granted access to system buses. When set high, places system buses in high-impedance (Tri-state) mode.
NENOUT	Enable output	Associated with SC/MP internal allocation logic for system bus. Set low when NENIN is low and SC/MP is not using system buses (NBREQ-high). Set high at all other times.
NADS	Address strobe output	Active-Low strobe. While low, indicates that valid address and status output are present on system buses.
NRDS	Read strobe output	Active-Low strobe. On trailing edge, data are input to SC/MP from 8-bit bidirectional data bus. High-impedance (Tri-state) output when input/output cycle not in progress.
NWDS	Write strobe output	Active-Low strobe. On trailing edge, data are input to SC/MP from 8-bit bidirectional data bus. High-impedance (Tri-state) output when input/output cycle not in progress.
NHOLD	Input/ouput cycle extend input	When set low prior to trailing edge of NRDS or NWDS strobe, stretches strobe to extend input/output cycle; that is, strobe is held low until NHOLD signal is returned high.
SENSE A	Sense/interrupt request input	Serves as interrupt request input when SC/MP internal IE (Interrupt Enable) flag is set. When IE flag is reset, serves as user-designated sense condition input. Sense condition testing is effected by copying status register to accumulator.
SENSE B	Sense input	User-designated sense-condition input. Sense-condition testing is effected by copying status register to accumulator.
SIN	Serial input to E register	Under software control, data on this line are right-shifted into E register by execution of SIO instruction.
SOUT	Serial output from E register	Under software control, data are right-shifted onto this line from E register by execution of SIO instruction. Each data bit remains latched until execution of next SIO instruction.
FLAGS 0, 1, 2	Flag outputs	User-designated general-purpose flag outputs of status register. Under program control, flags can be set and reset by copying accumulator to status register.
AD00-AD11	Address bit 00 through address bit 11	Twelve Tri-state address output lines. SC/MP outputs 12 least significant address bits on this bus when NADS strobe is low. Address bits are then held valid until trailing edge of read (NRDS) or write (NWDS) strobes. After trailing edge of NRDS or NWDS strobe, bus is set to high-impedance (Tri-state) mode until next NADS strobe.

PIN DESIGNATIONS[1] (Cont'd)

MNEMONIC	FUNCTIONAL NAME Output at NADS Time [2,3,4]	DESCRIPTION
DB0	Address Bit 12	Fourth most significant bit of 16-bit address.
DB1	Address Bit 13	Third most significant bit of 16-bit address.
DB2	Address Bit 14	Second most significant bit of 16-bit address
DB3	Address Bit 15	Most significant bit of 16-bit address.
DB4	R-Flag	When high, data input cycle is starting; when low, data output cycle is starting.
DB5	I-Flag	When high, first byte of instruction is being fetched.
DB6	D-Flag	When high, indicates delay cycle is starting; that is, second byte of DLY instruction is being fetched.
DB7	H-Flag	When high, indicates that Halt instruction has been executed. (In some system configurations, the H-Flag output is latched and, in conjunction with the Continue input, provides a programmed Halt.)

NOTES

[1] The 8-bit bidirectional data bus is set to the high-impedance (tri-state) mode except when it is actually in use by SC/MP (NADS, NRDS, or NWDS low). During the addressing interval of each input/output cycle (NADS low), SC/MP provides address and status outputs over the bus. During the ensuing data-transfer interval (NRDS or NWDS low), 8-bit input or output data bytes are routed over the bus.

[2] The DB0 through DB7 (AD12-HFLG) lines are a high-impedance (open circuit) load when SC/MP does not have access to the input/output bus.

[3] Input at NRDS time: Input data are expected on the eight (DB0-DB7) lines.

[4] Output at NWDS time: Output data are valid on the eight (DB0-DB7) lines.

DRIVERS AND RECEIVERS

Equivalent circuits for SC/MP drivers and receivers are shown in Figure 2. All inputs have static charge protection circuits consisting of an RC filter and voltage clamp. These devices still should be handled with care, as the protection circuits can be destroyed by excessive static charge.

SC/MP-II DRIVER AND RECEIVER EQUIVALENT CIRCUITS

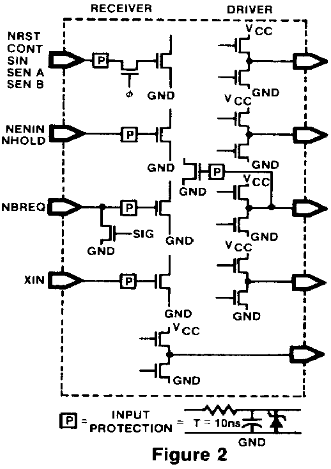


Figure 2

TIMING CONTROL

All necessary timing signals are provided by a three-stage inverter ring oscillator contained on the SC/MP chip. Two control pins, XIN and XOUT, permit the frequency of the oscillator to be controlled by any of the following methods:

1. By leaving the XOUT pin unterminated and driving the XIN pin with an externally generated TTL clock that conforms to the parameters shown in Figure 3A. For this method, the frequency of the oscillator is equalt to the frequency of the external clock input.
2. By connecting a resistor-capacitor feedback network between the XIN and XOUT pins and GND as shown in Figure 3B.
3. By connecting a crystal with low-pass filter network between the XIN and XOUT pins and GND as shown in Figure 3C (for above 1MHz or Figure 3D (for 1MHz or below). For this method, the frequency of the oscillator is equal to the resonant frequency of the crystal and the low-pass filter prevents unwanted harmonic oscillations.

In addition to illustrating appropriate frequency-control networks for the on-chip oscillator, Figures 3A through 3D also show how an optional driver may be used to derive a system clock from the oscillator signal present at the XOUT pin. For reference purposes, the timing relationship between the XOUT signal and the NADS strobe is shown in Figure 5.

In the discussions that follow, instruction execution and input/output timing are described in terms of microcycles.

The time interval of a microcycle is four times the period of the oscillator, that is:

$$\text{period of one microcycle} = 2T_C$$

$$T_C = 2\left(\frac{1}{f_{osc}}\right) = 2\left(\frac{1}{f_{res}}\right) = 2\left(\frac{1}{f_{XIN}}\right)$$

Where:

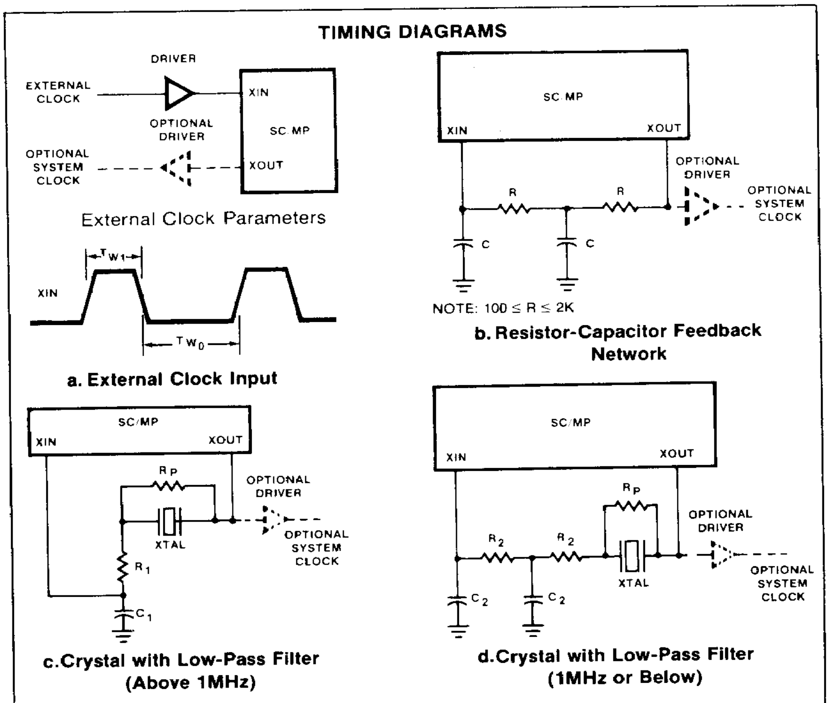
T_C = time period for two cycles of on-chip or external oscillator

f_{osc} = frequency of on-chip oscillator

f_{res} = resonant frequency of crystal connected between XIN and XOUT pins

f_{XIN} = frequency of external clock applied to XIN pin

TIMING DIAGRAMS



a. External Clock Input

b. Resistor-Capacitor Feedback Network

c.Crystal with Low-Pass Filter (Above 1MHz)

d.Crystal with Low-Pass Filter (1MHz or Below)

SUGGESTED VALUES FOR CRYSTAL WITH LOW-PASS FILTER NETWORK.

Crystal	Rp	C1	R1
2MHz	100kΩ	56pF	1kΩ
3.58MHz	100kΩ	27pF	1kΩ
4MHz	100kΩ	27pF	1kΩ

NOTE XTAL is parallel resonant with maximum series resonance equal to 1kΩ

Figure 3

FREQUENCY CONTROL NETWORKS FOR ON-CHIP OSCILLATOR

Typical Oscillator Frequency vs RC Time Constant

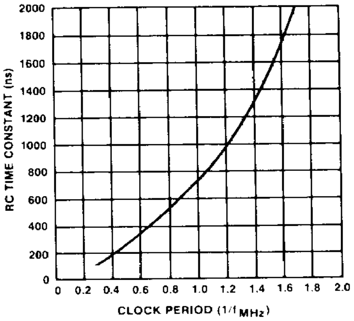


Figure 4

XOUT/NADS TIMING RELATIONSHIP

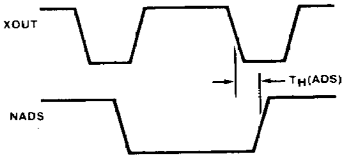


Figure 5

INSTRUCTION FORMAT

The SC/MP instruction repertoire includes both single-byte and double-byte instructions. A single-byte instruction consists of an 8-bit operation code that specifies an operation that SC/MP can execute without further reference to memory. A double-byte instruction consists of an 8-bit operation code and an 8-bit data or displacement field. When the second byte represents a data field, the data are processed by SC/MP during execution of the instruction, thereby eliminating the need for further memory references. When the second byte represents a displacement value, it is used to calculate a memory address that will be accessed (written into or read from) during execution of the instruction (refer to Addressing).

DATA STORAGE

As shown in the Detailed Block Diagram, SC/MP provides ten internal registers, seven of whivh are accessible to the programmer. The purpose and function of these registers are described below.

Program Counter—The program counter is a 16-bit register that contains the address of the instruction being executed. The contents of this register are automatically incremented by one just before each instruction is fetched from memory to enable sequential execution of the stored instructions. Under program control, the contents of this register also may be modified or exchanged with the contents of a pointer register to effect subroutine calls and program branches.

NOTE

The 16-bit address output of the program counter consists of a 4-bit high-order address and a 12-bit low-order address. When the program counter is incremented at the start of each instruction fetch input/output cycle, only the 12 low-order bits are affected; no carry is provided to the 4 high-order bits. For systems employing memories of 4K or less, the high-order bits can be ignored as they are set to 0000_{16} following initialization. For systems employing larger memories, the contents of a pointer register can be modified to select the desired 4K block of memory.

Pointer Registers—The pointer registers are 16-bit general-purpose registers that normally are loaded under program control with reference addresses that serve as page pointers, stack pointers, and subroutine pointers. In applications having minimal memory addressing requirements, these registers may be used alternately as data storage registers.

NOTE

When interrupt requests are enabled, pointer register 3 is automatically referenced by the internal microprogram for formation of the starting address of the user-generated interrupt service routine. (See Figure 10.) In this case, the contents of pointer register 3 must be set to one less thar the memory location of the first instruction in the interrup service routine.

Accumulator—The 8-bit accumulator (AC is the primary working register of SC/MP. I is used for performing and storing the results of arithmetic and logic operations as well as for data transfers, shifts, rotates, anc data exchanges with the program counter the pointer registers, and the status register

Extension Register—The extension registe is used both for serial input/output data transfers and with the accumulator to effec arithmetic, logic, and data-transfer operations. If the second byte of an indexed o auto-indexed memory reference instructio (refer to Addressing) equals -128_{10}, the contents of the extension register are used as the displacement value for address formation.

Status Register—The status register provides storage for arithmetic, control, and software status flags. For more detailed information on the function of this register, refer to Status Register under the description of the Arithmetic and Logic Unit.

Instruction Register—The 8-bit instruction register is not accessible to the programmer. During the fetch phase of each instruction cycle, this register is loaded with the 8-bit instruction operation code retrieved from memory (for a single-byte instruction or the first byte of a double-byte instruction).

Data Input/Output Register—The data input/output register is not accessible to the programmer. It is used for temporary storage of all input/output data received via or transmitted over the 8-bit bidirectional data bus during the data-transfer interval of each input/output cycle (NRDS or NWDS low).

Address Register—The 16-bit address register is not accessible to the programmer. It is used for temporary storage of the 16-bit address transmitted during an input/output cycle.

ARITHMETIC AND LOGIC UNIT

The Arithmetic and Logic Unit (ALU) provides the data-manipulation capability that is an essential feature of any microprocessor. The operations provided by the ALU include OR, XOR, increment, decrement, binary addition, and decimal addition. For decimal addition, the data inputs to the ALU are treated as two 4-bit BCD digits, thereby eliminating the program-storage and execution time required to perform BCD to binary conversion.

BUS TRANSFER LOGIC

The bus transfer logic processes the gating and function control outputs of the instruction-decode logic to provide the shift-right (with link, without link, or with serial input data), rotate (with or without link), and bus-exchange functions necessary for data movement between the SC/MP internal read and write buses. A general summary of the data-manipulation capabilities available to the programmer follows.

1. Either the low-order or the high-order byte of any pointer register can be exchanged with the contents of the 8-bit accumulator. Thus, data exchanges between the pointer registers can be effected one byte at a time via the accumulator.

2. The contents of the program counter can be directly exchanged with the contents of any pointer register.

3. The contents of the extension register can be loaded into the accumulator or can be exchanged with the contents of the accumulator. When the accumulator is loaded from the extension register, the original contents of the accumulator are lost.

4. The contents of the status register can be copied into the accumulator to enable status modification or conditional branch testing. When the status register is copied into the accumulator, the contents of the status register are not altered but the original contents of the accumulator are lost.

5. The contents of the accumulator can be copied into the status register to change the outputs of the status register, except for status bits 4 and 5 (Sense A and B inputs to SC/MP). Since these are read-only bits, they are not affected by data movements internal to SC/MP. Copying the accumulator into the status register does not alter the contents of the accumulator.

NOTE

The flag 0, 1, and 2 outputs of the status register serve as latched flags, in other words, they are set to the specified state when the contents of the accumulator are copied into the status register, and they remain in the specified state until the contents of the status register are modified again under program control.

STATUS REGISTER

The function of each bit in the status register is described briefly below.

7	6	5	4	3	2	1	0
CY/L	OV	S_B	S_A	IE	F_2	F_1	F_0

User Flag 0—User-assigned general-purpose status bit for implementation as software status bit or in system control applications. This status bit is available as an external output from SC/MP.

User Flag 1—Same as User Flag 0.

User Flag 2—Same as User Flag 0.

Interrupt Enable Flag—Internal status bit that is set and reset under program control. When set, SC/MP recognizes external interrupt requests received via Sense A input. When reset, inhibits SC/MP from recognizing interrupt requests.

Sense A—General purpose status input for sensing external conditions. When IE flag is reset, this bit can be tested by copying status register to accumulator. When IE flag is set, this bit serves as interrupt request input causing SC/MP to automatically branch to user-generated interrupt-service routine in response to high input.

Sense B—Same as Sense A except that it is not tested for interrupt status.

NOTE

Sense A and B inputs are read-only bits. Thus, they are not affected when the contents of the accumulator are copied into the status register.

Overflow (OV)—This bit is set if an arithmetic overflow occurs during an add (ADD, ADI, or ADE) or a complement-and-add instruction (CAD, CAI, or CAE). It is not affected by the decimal-add instructions (DAD, DAI, or DAE).

Carry/Link (CY/L)—This bit is set if a carry from the most significant bit occurs during an add, complement-and-add, or decimal-add instruction. Thus, it serves as a carry input to the next add instruction. In addition, it is included in the Shift Right with Link (SRL) and Rotate Right with Link (RRL) instructions.

CONTROL

The operation of the SC/MP microprocessor consists of repeatedly accessing or fetching instructions from the program stored in external memory and executing the operations specified by the instructions. These two steps are carried out under the control of an internal microprogram. (SC/MP is not user-microprogrammable.) The microprogram is similar to a state table specifying the series of states of system control signals necessary to carry out each instruction. Microprogram storage is provided in the instruction decode and control logic, and microprogram routines are implemented to fetch and execute instructions. The fetch routine first increments the program counter, and then causes the instruction address to be transferred from the program counter to the system buses via the output address register. The microprogram next initiates an input data transfer. When the instruction operation code is subsequently placed on the 8-bit data bus (single-byte instruction or first byte of double-byte instruction), the operation code is loaded into the instruction register. The operation code is then partially decoded to determine whether the instruction contains a second byte. If it does, a second input data transfer is effected to load the next byte in the data input/output register.

After the complete instruction is stored in the instruction and/or data input/output register(s), the instruction decoder transforms the instruction operation code into the address of the appropriate instruction-execution routine contained in the internal microprogram. The microprogram then branches to the specified internal address to initiate execution of the instruction. The resulting execution routine comprises one or more microinstructions that implement the required functions. For example, the first microcycle of an Extension Register Add Instruction (ADE) causes the contents of the extension register to be gated onto the read bus, transferred to the write bus via the bus control logic, and then written into the data input/output register. The next microcycle causes the contents of the accumulator to be gated onto the read bus, the contents of the read bus to be added to the contents of the data input/output register via the ALU, and the resultant output of the ALU to be written into the accumulator via the write bus. The final step of the execution routine is a jump back to the fetch routine to access the next instruction.

INITIALIZATION

Since SC/MP may power up in a random condition, the following power-up and initialization procedure is recommended.

1. Apply power (GND and V_{CC}) and set NRST low.

NOTE

Allow ample time (typically, 250ms) for the oscillator and the internal clocks to stabilize. In systems where NRST is set low after turning on power, NRST must remain low for a minimum of $4T_C$. While NRST is low, any in-process operations are aborted automatically. When NRST is low, strobes and address and data buses are in the Non-I/O state (high-Z state).

2. Set NRST high. If the rise time of this input is too slow, the processor, first, will initialize and execute a few instructions and, then, will reinitialize. If the application is such that multiple initialization is undesireable, NRST should be brought high at a minimum rate of 2 volts per microcycle.

NOTE

This causes the SC/MP internal control circuit to set the contents of all programmer-accessible registers to zero. Thus, when SC/MP is granted access to the system buses following initialization, the first instruction is fetched *always* from memory location 0001_{16}. The NBREQ output goes low, indicating the start of this input/output cycle; this occurs at a time within $13T_C$ after NRST is set high. Normal execution of the program continues as long as NRST remains high.

PARALLEL DATA TRANSFERS

Parallel data transfers occur during each instruction fetch and during the ensuing read/write cycle associated with execution of the memory-reference instructions. This class of instruction could perhaps more properly be called the "Input/Output Reference Class" in the case of the SC/MP microprocessor, since all data tranfers, whether with memory, peripheral devices, or a central processor data bus, occur through the execution of these instructions. This unified bus structure is in contrast with many other microprocessors and minicomputers that have one instruction type (input/output class) for communication with peripheral devices and another instruction type (memory reference class) for communication with memories. The advantage of the approach taken by SC/MP is that a wider variety of instructions (the entire memory-reference class) is available for communications with peripherals. Thus, the LD and ST (Load and Store) instructions can be used for basic transfers, the ILD and DLD (increment/decrement and load) instructions can be used for indexing peripheral registers, and the remaining memory reference instructions can be used, as required, for "one-step" retrieval and processing of peripheral input data.

BUS UTILIZATION

The bus utilization of SC/MP is shown in Table 2.

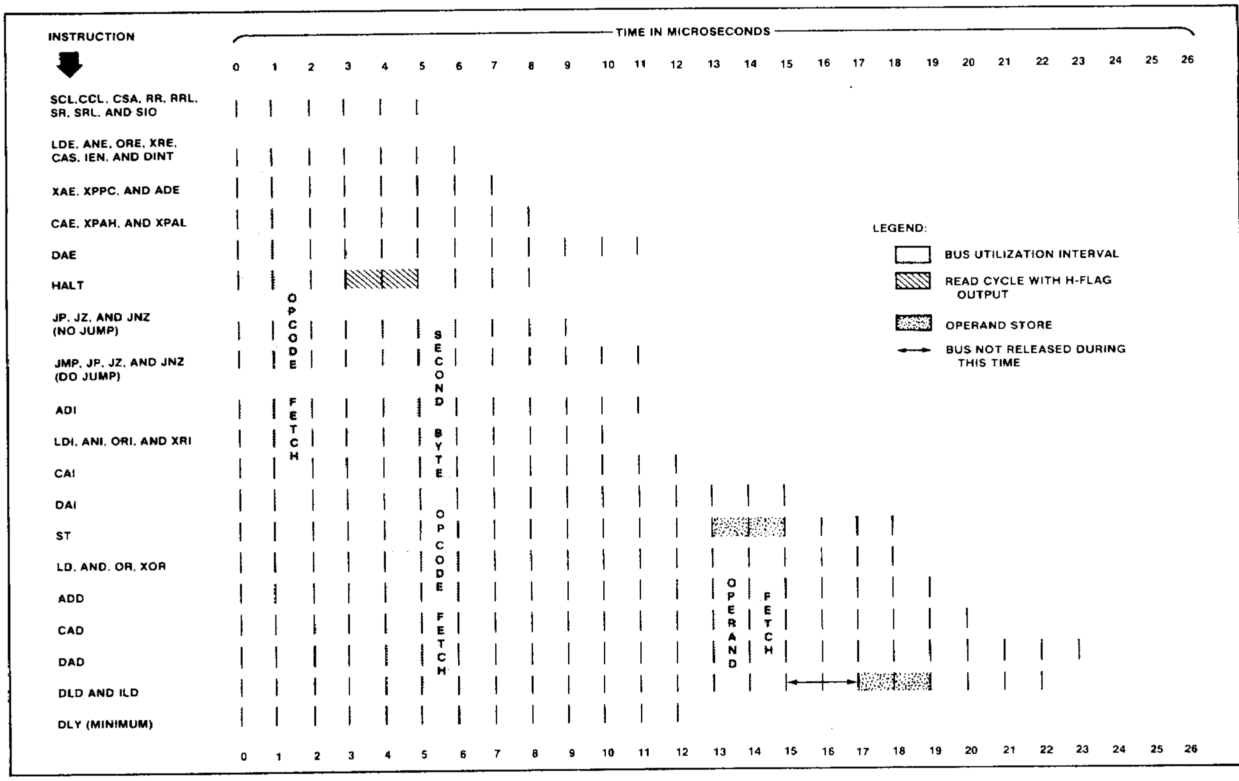


Table 2 BUS UTILIZATION OF EACH INSTRUCTION

NBREQ, NENIN, and NENOUT are active and bus access is controlled as shown in Figure 5. If NENIN is returned high during an input/output cycle, the input/output cycle is repeated when NENIN is again returned low.

During an ILD or DLD instruction, SC/MP does not relinquish the bus between the loading of the data and the storing of the modified data. If NENIN is brought high after the data have been loaded, the load portion of the cycle is not repeated when NENIN is returned low.

BUS ACCESS

Before SC/MP can initiate parallel data transfers with memory or peripheral devices, it must have access to the system address and data buses. Three of the SC/MP input/output signals are associated with bus control: NBREQ, NENIN, and NENOUT. For simple stand-alone applications, the NENOUT signal can be ignored and the NENIN signal can be tied to GND to allow the SC/MP microprocessor to have continual access to the system buses. The NBREQ input/output line then goes low during each input/output cycle as shown in Figures 6 and 7 to indicate when SC/MP is actually using the system buses.

NOTE

The NBREQ input/output line must be tied to V_{CC} via an external load resistor to allow normal operation of the SC/MP microprocessor.

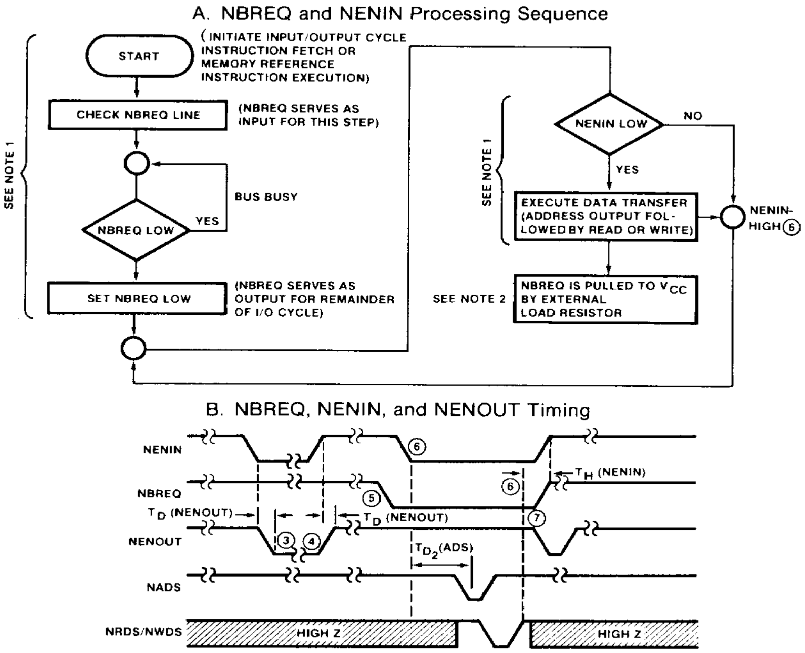
For DMA and multiprocessor applications, the NBREQ, NENIN, and NENOUT signals can be interconnected in various configurations to allow bus access to be granted to requesting devices according to user-specified priorities. Figure 5 illustrates the general sequence in which these signals are processed by SC/MP to gain access to the system buses and to indicate when the buses are actually being used.

INPUT/OUTPUT CYCLE

Once SC/MP has control of the system buses, the actual input/output cycle begins. As shown in Figures 6 and 7, the functions of memory addressing, data reading, and data writing are implemented, respectively, by the address strobe (NADS), the read strobe (NRDS), and the write strobe (NWDS). Note that the NBREQ signal is reset high at the end of the input/output cycle to indicate that the system buses are now free for use by the highest-priority requesting device.

The first operation that SC/MP performs for each input/output cycle is to load the 12 least significant address bits onto the 12-bit address bus, and the 4 most significant

BUS ACCESS CONTROL


A. NBREQ and NENIN Processing Sequence

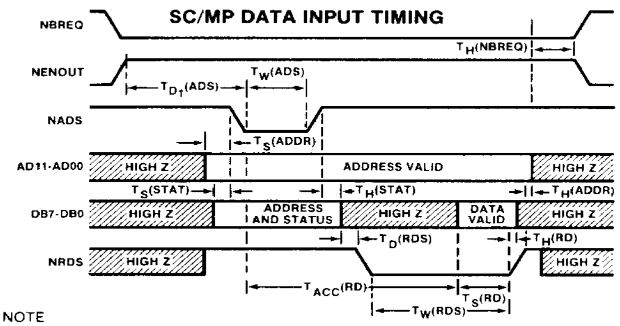
B. NBREQ, NENIN, and NENOUT Timing

START
(INITIATE INPUT/OUTPUT CYCLE INSTRUCTION FETCH OR MEMORY REFERENCE INSTRUCTION EXECUTION)
CHECK NBREQ LINE
(NBREQ SERVES AS INPUT FOR THIS STEP)
BUS BUSY
NBREQ LOW
YES
SET NBREQ LOW
(NBREQ SERVES AS OUTPUT FOR REMAINDER OF I/O CYCLE)
SEE NOTE 1
NENIN LOW
NO
YES
EXECUTE DATA TRANSFER (ADDRESS OUTPUT FOLLOWED BY READ OR WRITE)
NENIN-HIGH⑥
SEE NOTE 2
NBREQ IS PULLED TO VCC BY EXTERNAL LOAD RESISTOR
NENIN
NBREQ
NENOUT
NADS
NRDS/NWDS
TD (NENOUT)
TD (NENOUT)
TH (NENIN)
TD2(ADS)
HIGH Z
HIGH Z
③ ④ ⑤ ⑥ ⑦

NOTES

1. NENOUT is always high while SC/MP is actually using bus; that is, NENIN input and NBREQ output are low.
2. When SC/MP is not using bus (NBREQ output or NENIN input high), NENOUT is held in same state as NENIN input.
3. NENOUT goes low to indicate that SC/MP was granted access to bus (NENIN low) but is not using bus.
4. NENOUT goes high in response to high NENIN input.
5. SC/MP generates bus request; bus access not granted because NENIN high.
6. NENIN goes low. Bus access now granted and input/output cycle actually initiated. If NENIN is set high while SC/MP has access to the bus, the address and data ports will go to the high-impedance (Tri-State) state, but NBREQ will remain low. When NENIN is subsequently set low, the input/output cycle will begin again.
7. Input/output cycle completed. NENOUT goes low to indicate that SC/MP granted access to bus but not using bus. If NENIN had been set high before completion of input/output cycle. NENOUT would have remained high.

Figure 5



NOTE

Timing is valid when NENIN is low before NBREQ is set low by SC/MP; see Figure 5 for NADS timing when NENIN is set low after NBREQ.

Figure 6

address bits along with 4 status bits onto the 8-bit data bus. At the same time, SC/MP sets the NADS output low to indicate that the address and the status information are valid. The low-order address on the 12-bit bus is then held valid for the duration of the input/output cycle; the high-order address and the status information on the 8-bit bus remain valid only while NADS is low. While valid, the status bits have the following significance:

RFLG—When high, indicates that input/output cycle is read cycle; when low, indicates that input/output cycle is write cycle.

IFLG—Set high to indicate that instruction operation code (single-byte instruction or first byte of double-byte instruction) will be output from memory following NADS.

DFLG—Set high only when second byte of Delay Instruction is to be read from memory following NADS. Execution of the Delay Instruction then starts at trailing edge of NRDS. Upon completion, SC/MP provides NADS output to initiate next input/output cycle if bus access is granted. Time in microcycles from leading edge of delay flag to leading edge of subsequent NADS output is computed from the following formula:

Delay = [9 + 2(AC) + 2 disp + 2^9 disp] microcycles

where:

(AC) = unsigned contents of accumulator

disp = unsigned displacement value contained in second byte of Delay Instruction

The time derived from the above formula does not include the four microcycles required to fetch the first byte of the Delay Instruction. Thus, when the Delay Instruction is used for software timing, total instruction execution time equals [13 + 2(AC) + 2 disp + 2^9 disp] microcycles.

NOTE

When Halt Instruction is executed, instruction decode and control logic inhibits incrementing of program counter for one input/output cycle. Thus, Halt Instruction is read from memory a second time to enable generation of HFLG output, but no further processing of Halt Instruction occurs. In effect, this procedure ensures HFLG is output in advance of the next instruction to be fetched from memory.

HFLG—Set high only during addressing interval of read cycle that follows Halt Instruction. HFLG may be used to cause user-provided external logic to set the CONT input low, and thereby to effect a programmed halt. Since HFLG read cycle precedes the next instruction fetch, termination of programmed halt enables fetch of first instruction that follows Halt Instruction.

DATA OUTPUT TIMING

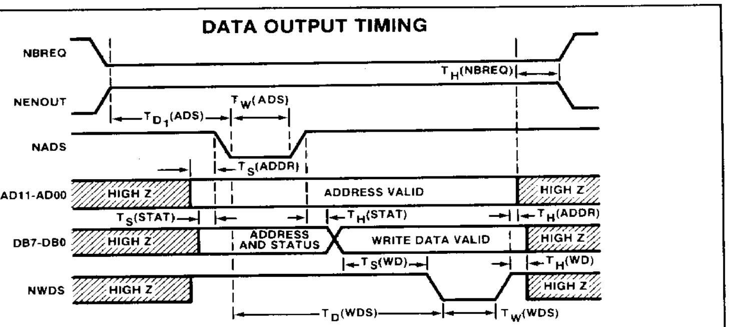
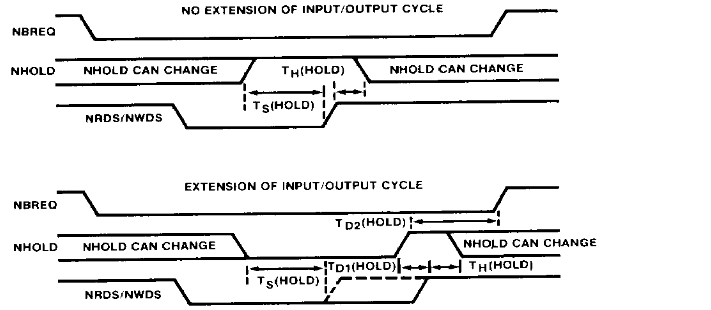


Figure 7

NHOLD TIMING



NOTE

In order to extend the input/output cycle, NHOLD must remain low until the point where NRDS/NWDS would have made a low-to-high transition with NHOLD inactive. Dashed line indicates the trailing edge of NRDS/NWDS when NHOLD is not active.

Figure 8

After resetting the NADS output, SC/MP generates an NRDS or NWDS strobe, respectively, to initiate a data-input (read) or data-output (write) operation. For a read operation, input data are strobed into SC/MP from the 8-bit bus on the trailing edge of the NRDS strobe. For a write operation, SC/MP places valid output data on the 8-bit bus on the leading edge of the NWDS strobe. After resetting the NRDS or NWDS strobe to complete the data transfer, SC/MP then resets the NBREQ signal to indicate that the system buses are free for use by another controller.

INPUT/OUTPUT CYCLE EXTENSION

As shown in Figure 8, the NHOLD signal may be set low prior to the trailing edge of the NRDS or NWDS strobe to cause SC/MP to lengthen the input/output cycle by holding the strobe active until after the NHOLD signal is returned high. Since there is no restriction on the maximum duration of the NHOLD signal, it can be used in a variety of applications ranging from accommodation of memories/peripherals with long access times to single-cycle control of the operating program for software debug purposes.

Figure 9 illustrates a typical circuit that may be used to generate an NHOLD signal of repeatable duration. The circuit shown employs an N74165 8-Bit Parallel In/Serial Out Shift Register to allow selection of an input/output cycle extend time that ranges from $T_C/2$ to $2T_C$ in increments of $T_C/2$. Functional operation of the circuit is controlled by the NADS strobe and XOUT signals. Each time that the NADS strobe goes low, the data present at the A through H terminals are loaded into the shift register in parallel. When the NADS strobe subsequently returns high, the data are then shifted out serially on the positive-to-negative transitions of XOUT. Thus, the NHOLD output of the circuit is set low on the leading edge of each NADS strobe and, as shown in the chart that accompanies the circuit diagram, it remains low for a time period ranging from three clock cycles minimum (B, C

D, and E inputs = Logic "1") to seven clock cycles maximum (B, C, D, and E inputs = Logic "0").

It is important to note that instruction execution time is increased whenever an input/output cycle is extended via the NHOLD signal. For purposes of computing the increase in instruction execution time, it is necessary to distinguish between the terms Input/Output Cycle Delay Period and Input/Output Cycle Extend Time. The term Input/Output Cycle Delay Period refers to the time that the NRDS/NWDS strobe is actually "stretched" to provide the required memory or peripheral access time. The term Input/Output Cycle Extend Time refers to the additional number of microcycles required by the internal SC/MP microprogram to complete the extended input/output cycle; that is:

INPUT/OUTPUT CYCLE	
Delay Period	**Extend Time**
$T_C/2$ through $2T_C$ $(> 0 \leq 1\mu$cycle)	1μcycle
$5T_C/2$ through $4T_C$ $(> 1 \leq 2\mu$cycles)	2μcycles
$9T_C/2$ through $6T_C$ $(> 2 \leq 3\mu$cycles)	3μcycles
etc.	etc.

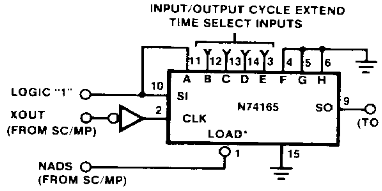
The total increase in instruction execution time, therefore, is equal to the Input/Output Cycle Extend Time multiplied by the total number of input/output cycles associated with the instruction. For example, a DLD Instruction is normally executed in 22 microcycles. Since this instruction employs three read input/output cycles and one write input/output cycle, an Input/Output Cycle Extend Time of one microcycle would increase total DLD Instruction execution time to 26 microcycles.

SERIAL DATA TRANSFERS

Serial input/output data transfers can be used efficiently with very slow input/output peripherals such as X-Y plotters, teletypewriters, slow-speed printers, and so forth. Such transfers can be effected in any of the following manners:

1. By assigning serial input/output functions to the extension register via the SIO (Serial Input/Output) Instruction. When this instruction is executed, the contents of the extension register are shifted right one bit. At the same time, data present on the SIN line are shifted into bit position 7 of the extension register and the original contents of bit position 0 are shifted into a flip-flop to provide a latched output of the SOUT line. The SOUT data are then held latched until the next SIO instruction is executed.

TYPICAL NHOLD CONTROL CIRCUIT



DATA INPUTS B C D E	NHOLD DURATION (in clock cycles)	INPUT/OUTPUT CYCLE		REQUIRED MEMORY ACCESS TIME [T_{ACC} (RD)]
		DELAY PERIOD	EXTEND TIME IN MICROCYCLES	
1 1 1 1	3	0	0	$2T_C$ - 200
1 1 1 0	4	$T_C/2$	1	$(5T_C/2)$ - 200
1 1 0 0	5	T_C	1	$3T_C$ - 200
1 0 0 0	6	$3T_C/2$	1	$(7T_C/2)$ - 200
0 0 0 0	7	$2T_C$	1	$4T_C$ - 200

Figure 9

2. By using one of the status flags as an output data bit and one of the sense lines as an input data bit.

3. By implementing external logic such that only one line of the 8-bit data input/output bus is used.

For synchronous systems, serial data input/output timing may be provided by program loops that employ the delay instruction, or by using one or more of the transfer instructions (see Table 2) to test the output of an external timing circuit. For asynchronous systems, one of the sense inputs can be used for testing bit-received/ready status and a pulsed flag output can be provided, under program control, for peripheral indexing each time that a data bit is actually shifted in or out.

Systems that have several inpu̇t/outout devices must be multiplexed, device selection can then be accomplished using the status flag outputs of SC/MP, or by using parallel input/output commands to load an external latch. Systems that do not require serial input/output capability can employ the SIN and SOUT lines as a sense input and flag output, respectively.

INTERRUPTS

When the internal interrupt enable (IE) flag is set under program control, the Sense A line is enabled to serve as an interrupt request input; when the IE flag is reset, SC/MP is inhibited from detecting interrupts. Thus, while the IE flag is set, the Sense A input is tested prior to the fetch phase of each instruction as shown in Figure 10. Upon detection of an interrupt request (Sense A high), the following events occur automatically.

MICROPROCESSOR HALT AND INTERRUPT REQUEST INPUT PROCESSING

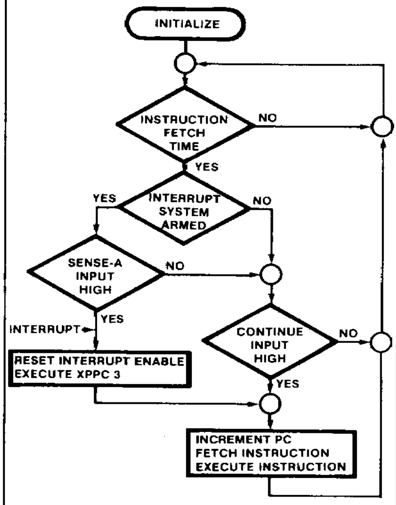


Figure 10

1. The status register IE flag is reset to prevent SC/MP from responding to any further interrupt requests. Interrupt request capability can then be reenabled during or at the end of the ensuing user-generated interrupt service routine via the IEN (Enable Interrupt) Instruction or by copying the accumulator into the status register.

2. The contents of the program counter are exchanged with the contents of the pointer register 3.

3. The contents of the program counter are incremented by one to address the first instruction of the user-generated interrupt service routine.

The interrupt system must be armed before interrupts are enabled. This is accomplished as follows:

1. First, the Interrupt Enable Bit in the Status Register is set true by executing either an Enable Interrupt Instruction (IEN) or a Copy Accumulator to Status Register Instruction (CAS).
2. Second, one additional instruction is fetched and executed.

A return from interrupt is accomplished by executing two instructions: Enable Interrupt (IEN) immediately followed by Exchange Pointer 3 with Program Counter (XPPC 3).

MICROPROCESSOR HALT

The CONT input to SC/MP is provided to enable suspension of operation without loss of internal status. Processing of the CONT input is shown in Figure 9. Since this is an asynchronous input, it can be controlled by external timing logic, or as stated previously, the HALT flag output that appears on the 8-bit data bus (during the read cycle that follows execution of a Halt Instruction) can be used with an external circuit to effect a programmed halt condition. Note that when an interrupt request is detected while the CONT input is low, the first instruction of the user-generated interrupt service routine is automatically executed. Thus, the first instruction of the interrupt service routine can be used to reset the external CONT input logic and, thereby, to terminate the microprocessor halt condition if so desired.

After execution of an instruction, the CONT input must be high for a minimum time of $2T_C$ (1 microcycle) in order to fetch and execute the next instruction.

INSTRUCTION SET

The SC/MP instruction set provides the general-purpose user of microprocessors a powerful programming capability along with above-average flexibility and speed. The instruction set consists of 46 instructions, which comprise eight general categories. A listing of the complete instruction set is provided in Table 2; typical instruction execution times are given in Table 3, and notations and symbols used as shorthand expressions of instruction capability are defined in Table 4.

DOUBLE-BYTE INSTRUCTIONS

MNEMONIC	DESCRIPTION	OBJECT FORMAT	OPERATION	MICRO-CYCLES
	Memory reference instructions	7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0		
LD	Load	1 1 0 0 0 | m | ptr | disp	(AC)←(EA)	18
ST	Store	1 1 0 0 1	(EA)←(AC)	18
AND	AND	1 1 0 1 0	(AC)←(AC) Λ (EA)	18
OR	OR	1 1 0 1 1	(AC)←(AC) V (EA)	18
XOR	Exclusive-OR	1 1 1 0 0	(AC)←(AC) ⊻ (EA)	18
DAD	Decimal add	1 1 1 0 1	$(AC)←(AC)_{10}+(EA)_{10}+(CY/L)$; (CY/L)	23
ADD	Add	1 1 1 1 0	(AC)←(AC)+(EA)+(CY/L); (CY/L), (OV)	19
CAD	Complement and add	1 1 1 1 1	(AC)←(AC) + ~(EA) + ((CY/L); (CY/L), (OV)	20
	Memory increment/ decrement instructions	7 6 5 4 3 2 | 1 0 | 7 6 5 4 3 2 1 0		
ILD	Increment and load	1 0 1 0 1 0 | ptr | disp	(AC), (EA)←(EA) + 1	22
DLD	Decrement and load	1 0 1 1 1 0	(AC), (EA)←(EA) - 1	22
	Immediate instructions	7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0		
LDI	Load immediate	1 1 0 0 0 1 0 0 | data	(AC)←data	10
ANI	AND immediate	1 1 0 1 0 1 0 0	(AC)←(AC) Λ data	10
ORI	OR immediate	1 1 0 1 1 1 0 0	(AC)←(AC) V data	10
XRI	Exclusive-OR immediate	1 1 1 0 0 1 0 0	(AC)←(AC) ⊻ data	10
DAI	Decimal add immediate	1 1 1 0 1 1 0 0	$(AC)←(AC)_{10}+data_{10}+(CY/L)$; (CY/L)	15
ADI	Add immediate	1 1 1 1 0 1 0 0	(AC)←(AC)+data+(CY/L); (CY/L), (OV)	11
CAI	Complement and add immediate	1 1 1 1 1 1 0 0	(AC)←(AC) + ~data + (CY/L); (CY/L), (OV)	12
	Transfer instructions	7 6 5 4 3 2 | 1 0 | 7 6 5 4 3 2 1 0		
JMP	Jump	1 0 0 1 0 0 | ptr | disp	(PC)←EA	11
JP	Jump if positive	1 0 0 1 0 1	If (AC) ≥ 0, (PC)←EA	9, 11
JZ	Jump if zero	1 0 0 1 1 0	If (AC) = 0, (PC)←EA	9, 11
JNZ	Jump if not zero	1 0 0 1 1 1	If (AC) ≠ 0, (PC)←EA	9, 11
	Double-byte miscellaneous instructions	7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0		
DLY	Delay	1 0 0 0 1 1 1 1 | disp	count AC to -1. delay = 13 + 2(AC) + 2 disp + 2^9 disp microcycles	13 to 131, 593

Table 3 SC/MP INSTRUCTION SUMMARY

SINGLE-BYTE INSTRUCTIONS

MNEMONIC	DESCRIPTION	OBJECT FORMAT	OPERATION	MICRO-CYCLES
	Extension register instructions	7 6 5 4 3 2 1 0		
LDE	Load AC from extension	0 1 0 0 0 0 0 0	$(AC)\leftarrow(E)$	6
XAE	Exchange AC and extension	0 0 0 0 0 0 0 1	$(AC)\leftarrow(E)$	7
ANE	AND extension	0 1 0 1 0 0 0 0	$(AC)\leftarrow(AC)\ \Lambda\ (E)$	6
ORE	OR extension	0 1 0 1 1 0 0 0	$(AC)\leftarrow(AC)\ V\ (E)$	6
XRE	Exclusive-OR extension	0 1 1 0 0 0 0 0	$(AC)\leftarrow(AC)\ \forall\ (E)$	6
DAE	Decimal add extension	0 1 1 0 1 0 0 0	$(AC)\leftarrow(AC)_{10}+(E)_{10}+(CY/L); (CY/L)$	11
ADE	Add extension	0 1 1 1 0 0 0 0	$(AC)\leftarrow(AC)+(E)+(CY/L); (CY/L), (OV)$	7
CAE	Complement and add extension	0 1 1 1 1 0 0 0	$(AC)\leftarrow(AC)+\$(E)+(CY/L); (CY/L), (OV)$	8
	Pointer register move instructions	7 6 5 4 3 2 1 0		
XPAL	Exchange pointer low	0 0 1 1 0 0 ptr	$(AC)\leftarrow(PTR_{7:0})$	8
XPAH	Exchange pointer high	0 0 1 1 0 1	$(AC)\leftarrow(PTR_{15:8})$	8
XPPC	Exchange pointer with PC	0 0 1 1 1 1	$(PC)\leftarrow(PTR)$	7
	Shift, rotate, serial I/O instructions	7 6 5 4 3 2 1 0		
SIO	Serial input/output	0 0 0 1 1 0 0 1	$(E_i)\rightarrow(E_{i-1})$, $SIN\rightarrow(E_7)$, $(E_0)\rightarrow SOUT$	5
SR	Shift right	0 0 0 1 1 1 0 0	$(AC_i)\rightarrow(AC_{i-1})$, $0\rightarrow(AC_7)$	5
SRL	Shift right with link	0 0 0 1 1 1 0 1	$(AC_i)\rightarrow(AC_{i-1})$, $(CY/L)\rightarrow(AC_7)$	5
RR	Rotate right	0 0 0 1 1 1 1 0	$(AC_i)\rightarrow(AC_{i-1})$, $(AC_0)\rightarrow(AC_7)$	5
RRL	Rotate right with link	0 0 0 1 1 1 1 1	$(AC_i)\rightarrow(AC_{i-1})$, $(AC_0)\rightarrow(CY/L)\rightarrow(AC_7)$	5
	Single-byte miscellaneous instructions	7 6 5 4 3 2 1 0		
HALT	Halt	0 0 0 0 0 0 0 0	Pulse H-flag	8
CCL	Clear carry/link	0 0 0 0 0 0 1 0	$(CY/L)\leftarrow0$	5
SCL	Set carry/link	0 0 0 0 0 0 1 1	$(CY/L)\leftarrow1$	5
DINT	Disable interrupt	0 0 0 0 0 1 0 0	$(IE)\leftarrow0$	6
IEN	Enable interrupt	0 0 0 0 0 1 0 1	$(IE)\leftarrow1$	6
CSA	Copy status to AC	0 0 0 0 0 1 1 0	$(AC)\leftarrow(SR)$	5
CAS	Copy AC to status	0 0 0 0 0 1 1 1	$(SR)\leftarrow(AC)$	6
NOP	No operation	0 0 0 0 1 0 0 0	None	5

Table 3 SC/MP INSTRUCTION SUMMARY (Cont'd)

INSTRUCTION	READ CYCLES	WRITE CYCLES	TOTAL MICROCYCLES
ADD	3	0	19
ADE	1	0	7
ADI	2	0	11
AND	3	0	18
ANE	1	0	6
ANI	2	0	10
CAD	3	0	20
CAE	1	0	8
CAI	2	0	12
CAS	1	0	6
CCL	1	0	5
CSA	1	0	5
DAD	3	0	23
DAE	1	0	11
DAI	2	0	15
DINT	1	0	6
DLD	3	1	22
DLY	2	0	13 - 131593
HALT	2	0	8
IEN	1	0	6
ILD	3	1	22
JMP	2	0	11
JNZ	2	0	9,11 for Jump
JP	2	0	9,11 for Jump
JZ	2	0	9,11 for Jump
LD	3	0	18
LDE	1	0	6
LDI	2	0	10
NOP	1	0	5
OR	3	0	18
ORE	1	0	6
ORI	2	0	10
RR	1	0	5
RRL	1	0	5
SCL	1	0	5
SIO	1	0	5
SR	1	0	5
SRL	1	0	5
ST	2	1	18
XAE	1	0	7
XOR	3	0	18
XPAH	1	0	8
XPAL	1	0	8
XPPC	1	0	7
XRE	1	0	6
XRI	2	0	10

NOTE

If slow memory is being used, the appropriate delay should be added for each read or write cycle.

Table 4 INSTRUCTION EXECUTION TIME

ADDRESSING

During execution, instructions and data defined in a program are stored into and loaded from specific memory locations, the accumulator, or selected registers. Because SC/MP, memory (read/write and read-only), and peripherals are on a common data bus, any instruction used to address memory may be used to address the peripherals. The formats of the instruction groups that reference memory are shown below.

7 3	2	1,0	7 0	
opcode	m	ptr	disp	Memory Reference Instructions
opcode		ptr	disp	Memory Increment/ Decrement Instructions and Transfer Instructions

Memory-reference instructions use the PC-relative, indexed, or auto-indexed methods of addressing memory. The memory-increment/decrement instructions and the transfer instructions use the PC-relative or indexed methods of addressing.

The various methods of addressing memory and peripherals are shown below.

Immediate addressing is an addressing format specific to the immediate instruction group.

TYPE OF ADDRESSING	OPERAND FORMATS		
	m	ptr	disp
PC-relative	0	0	-128 to +127
Indexed	0	1,2, or 3	-128 to +127
Immediate	1	0	-128 to +127
Auto-indexed	1	1,2, or 3	-128 to +127

For PC-relative, indexed, and auto-indexec memory-reference instructions, anothe feature of the addressing architecture is tha the contents of the extension register are substituted for the displacement if the instruction displacement equals -128 (-X'80)

NOTE

All arithmetic operations associated with addres formation affect only the 12 low-order address bits; n carry is provided to the 4 high-order bits. For system employing memories of 4K or less, the high-order bits ca be ignored as they are set to 0000 following initializatior For systems employing larger memories, the high-orde bits must be set to the starting address of the desired 4 block of memory. For example:

0001_2 enables memoory locations 1000_{16} - $1FFF_{16}$ to b addressed.

0010_2 enables memory locations 2000_{16} - $2FFF_{16}$ to b addressed and so forth.

PC-Relative Addressing—A PC-relative ad dress is formed by adding the displacemen value specified in the operand field of th instruction to the current contents of th program counter. The displacement is an 8 bit twos-complement number, so the rang of the PC-relative addressing format i -128_{10} to $+127_{10}$ locations from the curren contents of the program counter.

Immediate Addressing—Immediate ad dressing uses the value in the second byt of a double-byte instruction as the operan for the operation to be performed (see below).

For example, compare a Load (LD) instruc tion to a Load Immediate (LDI) instructio The Load instruction uses the contents the second byte of the instruction in computing the effective address of the data to be loaded. The Load Immediate instruction uses the contents of the second byte as the data to be loaded.

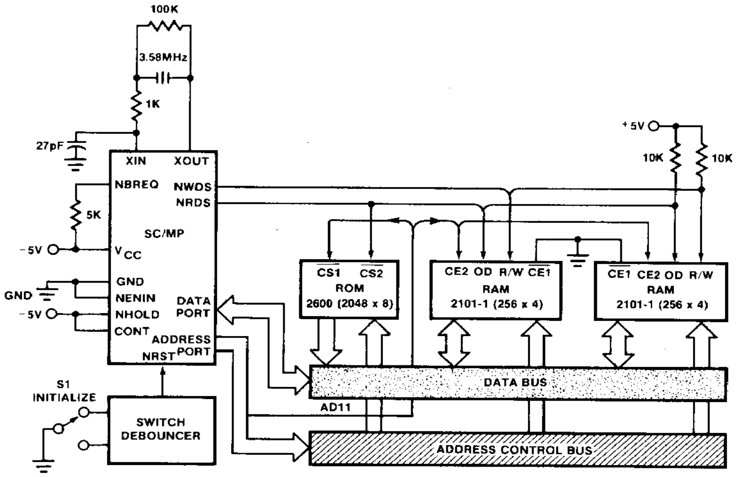
Indexed Addressing—Indexed addressing enables the programmer to address any location in memory through the use of the pointer register and the displacement. When indexed addressing is specified in an instruction, the contents of the designated pointer register are added to the displacement to form the effective address. The contents of the pointer register are not modified by indexed addressing.

Auto-Indexed Addressing—Auto-indexed addressing provides the same capabilities as indexed addressing along with the ability to increment or decrement the designated pointer register by the value of the displacement. If the displacement is less than zero, the pointer register is decremented by the displacement before the contents of the effective address are fetched or stored. If the displacement is equal to or greater than zero, the pointer register is used as the effective address, and the pointer register is incremented by the displacement after the contents of the effective address are fetched or stored.

SYSTEM IMPLEMENTATION

Figures 11 and through 13 illustrate typical SC/MP system configurations. In Figure 10, SC/MP is shown interconnected to three memory devices to form a stand-alone 4-device system that provides 256 words of read/write memory and 2,048 words for program storage. Figure 12 shows SC/MP interconnected to an external controller for Direct Memory Access (DMA) operation, and Figure 13 illustrates a multiprocessor application using SC/MP's built-in logic to control bus access.

SC/MP-II FOUR-CHIP SYSTEM



NOTE

Part numbers are shown only for information purposes. Other memory components with suitable characteristics can be used.

Figure 11

SC/MP-II INTERCONNECTED FOR DIRECT MEMORY ACCESS (DMA) OPERATION

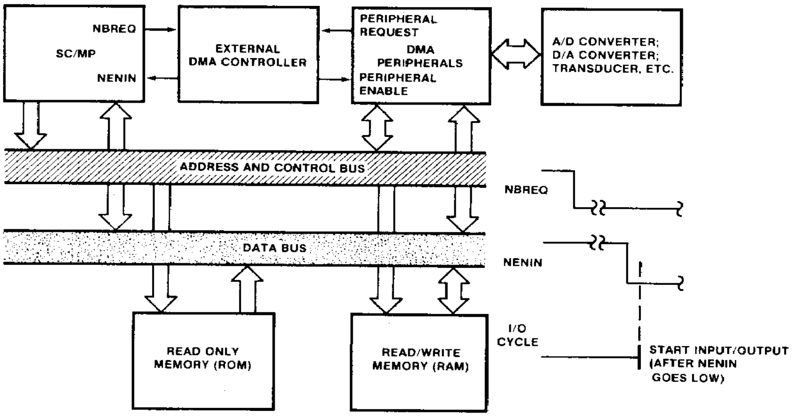


Figure 12

MULTIPROCESSOR SYSTEM USING SC/MP-II BUILT-IN LOGIC FOR BUS CONTROL

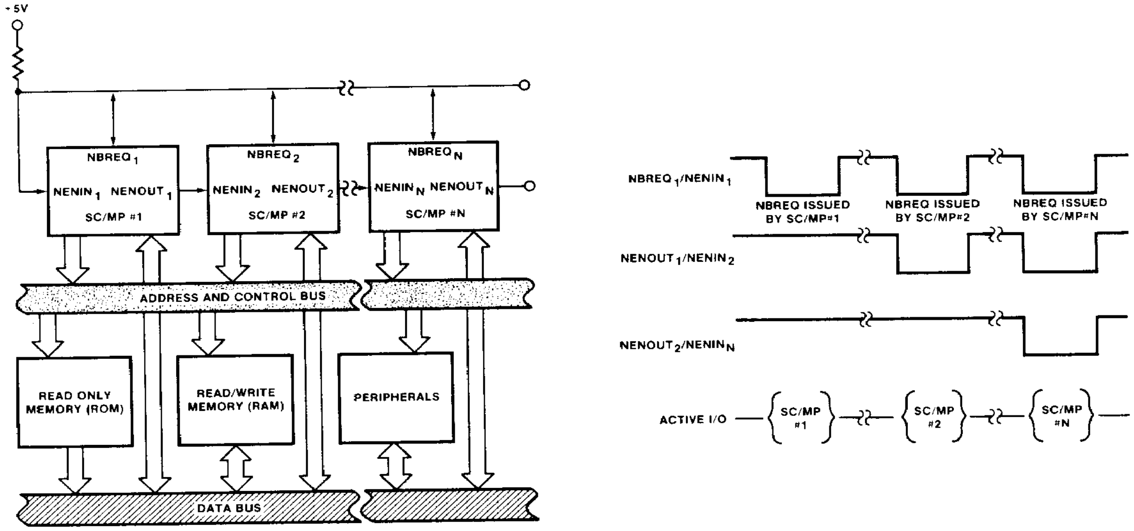


Figure 13

SYMBOL AND NOTATION	MEANING
AC	8-bit Accumulator.
CY/L	Carry/Link Flag in the Status Register.
data	Signed, 8-bit immediate data field.
disp	Displacement; represents an operand in a nonmemory reference instruction or an address modifier field in a memory reference instruction. It is a signed twos-complement number.
EA	Effective Address as specified by the instruction.
E	Extension Register; provides for temporary storage, variable displacements and separate serial input/output port.
i	Unspecified bit of a register.
IE	Interrupt Enable Flag.
m	Mode bit, used in memory reference instructions. Blank parameter sets m = 0, at sets m = 1.
OV	Overflow Flag in the Status Register.
PC	Program Counter (Pointer Register 0); during address formation, PC points to the last byte of the instruction being executed.
ptr	Pointer Register (ptr = 0 through 3). The register specified in byte 1 of the instruction.
$\text{ptr}_{n:m}$	Pointer register bits; n:m = 7 through 0 or 15 through 8.
SIN	Serial Input pin.
SOUT	Serial Output pin.
SR	8-bit Status Register.
( )	Means "contents of." For example, (EA) is contents of Effective Address.
[ ]	Means optional field in the assembler instruction format.
~	Ones complement of value to right of ~.
→	Means "replaces."
←	Means "is replaced by."
↔	Means "exchange."
@	When used in the operand field of the instruction, sets the mode bit (m) to 1 for auto-incrementing/auto-decrementing indexing.
10+	Modulo 10 addition.
∧	AND operation.
∨	Inclusive-OR operation.
∀	Exclusive-OR operation.
≥	Greater than or equal to.
=	Equals.
≠	Does not equal.

Table 5 GLOSSARY