Order this document
by BR701/D

MOTOROLA
SEMICONDUCTOR
TECHNICAL DATA

MC68HC05B6

Technical Summary

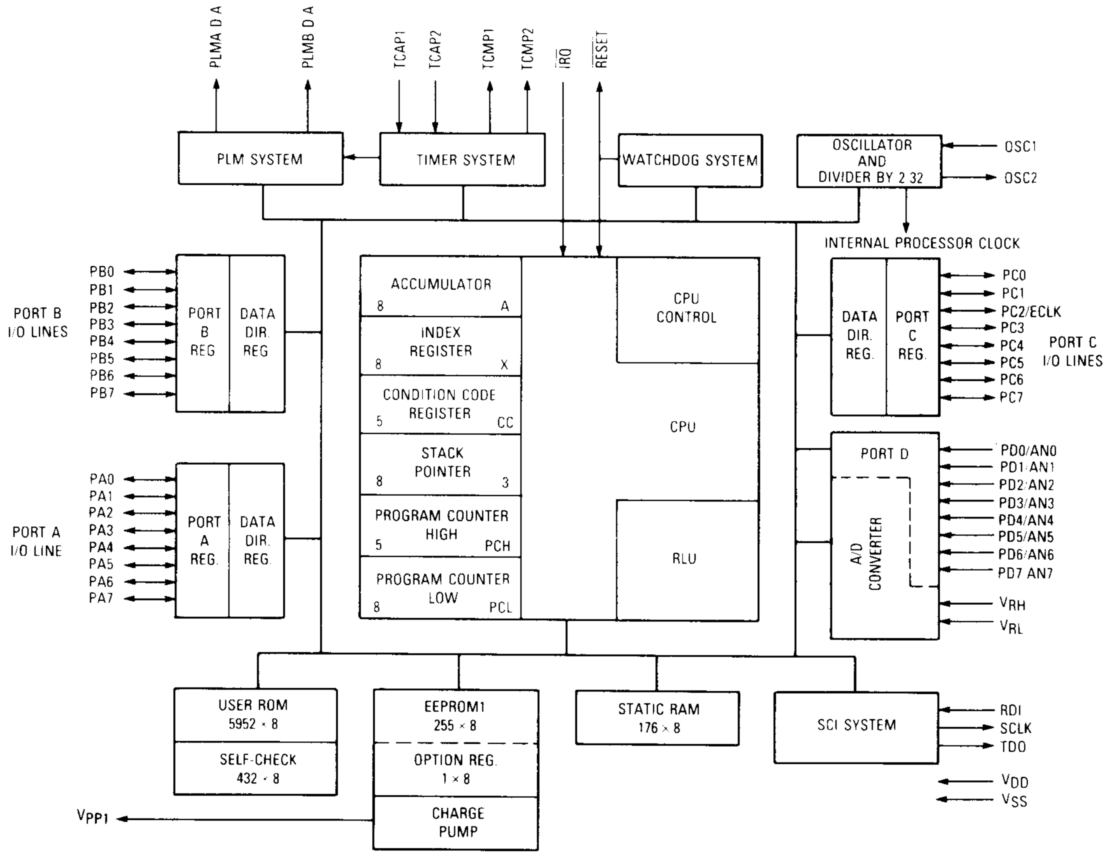
8-Bit Microcontroller Unit

The MC68HC05B6 (HCMOS) microcontroller unit (MCU) is a member of the M68HC05 Family of microcontrollers. This high-performance, low-power MCU has parallel I/O capability with pins programmable as input or output. This publication contains condensed information on the MCU; for more detailed information, contact your local Motorola sales office.

The following block diagram depicts the hardware features; additional features available on the MCU are shown below and at the top of page 2.

- On-Chip Oscillator with Crystal/Ceramic Resonator
- Memory-Mapped I/O
- 176 Bytes of On-Chip RAM
- 256 Bytes of On-Chip EEPROM

BLOCK DIAGRAM



This document contains information on a new product. Specifications and information herein are subject to change without notice.

MOTOROLA

FEATURES (Continued)

- 5952 Bytes of User ROM
- 24 Bidirectional I/O Lines and 8 Input-Only Lines
- Serial Communications Interface (SCI) System
- 8-Channel A/D Converter
- Watchdog System
- Self-Check Mode
- Power-Saving STOP and WAIT Modes
- Single 3.0- to 6.0-Volt Supply
- Fully Static Operation
- Two Pulse-Length Modulation Systems (D/A)
- 16-Bit Timer with Two Input Capture and Two Output Functions
- Slow Mode Option Divides the Basic Clock Frequency by 16

SIGNAL DESCRIPTION

The signal descriptions of the MCU are discussed in the following paragraphs.

V_{DD} AND V_{SS}

Power is supplied to the microcontroller using these two pins. V_{DD} is the positive supply, and V_{SS} is ground.

$\overline{\text{IRQ}}$

This pin is a programmable option that provides four different choices of interrupt triggering sensitivity. Refer to **INTERRUPTS** for more detail. Note that the voltage level on this pin affects the mode of operation.

OSC1, OSC2

These pins provide control input for an on-chip clock oscillator circuit. A crystal, a ceramic resonator, or an external signal connects to these pins providing a system clock. The oscillator frequency is two times the internal bus rate (or 32 times as a software option).

Crystal

The circuit shown in Figure 1(b) is recommended when using a crystal. Using an external CMOS oscillator is recommended when crystals outside the specified ranges are to be used. The crystal and components should be mounted as close as possible to the input pins to minimize output distortion and start-up stabilization time. Refer to **ELECTRICAL SPECIFICATIONS** for V_{DD} specifications.

Ceramic Resonator

A ceramic resonator may be used in place of the crystal in cost-sensitive applications. The circuit in Figure 1(b) is recommended when using a ceramic resonator. Figure 1(a) lists the recommended capacitance and resistance values. The manufacturer of the resonator considered should be consulted for specific information on resonator operation.

External Clock

An external clock should be applied to the OSC1 input with the OSC2 input not connected, as shown in Figure 1(d).

INPUT CAPTURE (TCAP1)

This pin controls the input capture 1 feature for the on-chip programmable timer (see Table 2).

INPUT CAPTURE (TCAP2)

This pin controls the input capture 2 feature for the on-chip programmable timer.

OUTPUT COMPARE (TCMP1)

This pin provides an output for the output compare 1 feature of the on-chip timer.

OUTPUT COMPARE (TCMP2)

This pin provides an output for the output compare 2 feature on the on-chip timer.

$\overline{\text{RESET}}$

This pin is used to reset the MCU and provide an orderly start-up procedure by pulling $\overline{\text{RESET}}$ low. The voltage on this pin affects the mode of operation (see Table 2, **Mode of Operation Selection**).

INPUT/OUTPUT PORTS (PA7–PA0, PB7–PB0, PC7–PC0)

These 24 lines are arranged into three 8-bit ports (A, B, and C). These ports are programmable as either inputs or outputs under software control of the data direction registers. Refer to **PROGRAMMING** for additional information.

FIXED INPUT PORT (PD7/AN7–PD0/AN0)

These eight lines comprise port D, a fixed input port. Enabling the A/D function affects this port. Port D accepts the eight analog inputs when the A/D is enabled. Port D can be used for digital input during a conversion sequence, but this may inject noise on the analog signals, reducing the conversion accuracy. Also, a digital read of

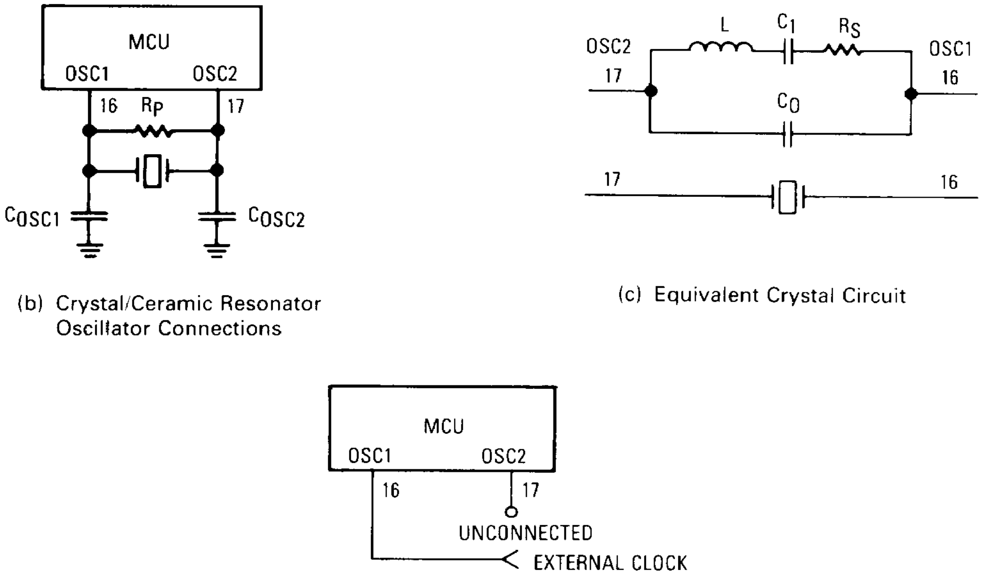
Crystal

	2 MHz	4 MHz	Units
R_{SMAX}	400	75	Ω
C_0	5	7	pF
C_1	0.008	0.012	μF
C_{OSC1}	15-40	15-30	pF
C_{OSC2}	15-30	15-25	pF
R_P	10	10	MΩ
Q	30	40	K

Ceramic Resonator

	2-4 MHz	Units
R_S (typical)	10	Ω
C_0	40	pF
C_1	4, 3	μF
C_{OSC1}	30	pF
C_{OSC2}	30	pF
R_P	1-10	MΩ
Q	1250	—

(a) Crystal/Ceramic Resonator Parameters



(b) Crystal/Ceramic Resonator Oscillator Connections

(c) Equivalent Crystal Circuit

(d) External Clock Source Connections

Figure 1. Oscillator Connections

port D with levels other than V_{DD} or V_{SS} on the pins results in greater power dissipation during the read cycle. Refer to **PROGRAMMING** for additional information.

NOTE

In the 48-pin dual-in-line package, the fixed input port (D) of the MC68HC05B6 is reduced to six pins (PD5–PD0, AN5–AN0). This change has no effect on either programming or operation of port D or the A/D converter.

PLMA

This pin is the output of the pulse-length modulation converter A. See **PULSE-LENGTH D/A CONVERTERS** for further information.

PLMB

This pin is the output of the pulse-length modulation converter B. See **PULSE-LENGTH D/A CONVERTERS** for further information.

RDI (Receive Data In)

This pin is the input of the SCI. See **Serial Communications Interface** for more information.

TDO (Transmit Data Out)

This pin is the output of the SCI. See **Serial Communications Interface** for more information.

SCLK

This pin is the clock output pin of the SCI transmitter. See **Serial Communications Interface** for more information.

V_{PP1}

This pin is the EEPROM programming voltage output. See **EEPROM** for further information.

V_{RH}

This pin is the positive reference voltage for the A/D converter.

V_{RL}

This pin is the negative reference voltage for the A/D converter.

INPUT/OUTPUT PROGRAMMING

Input/output port programming, fixed input port programming, and serial port programming are discussed in the following paragraphs.

INPUT/OUTPUT PORT PROGRAMMING

Any port pin is programmable as either an input or an output under software control of the corresponding data direction register (DDR). Each port bit can be selected as output or input by writing the corresponding bit in the port DDR to a logic one for output and logic zero for input. On reset, all DDRs are initialized to logic zero to put the ports in the input mode. The port output registers are not initialized on reset but may be written to before setting the DDR bits to avoid undefined levels.

When programmed as outputs, the latched output data is readable as input data regardless of the logic levels at the output pin due to output loading. The latched output data bit may always be written. Therefore, any write to a port writes all of its data bits, even though the port DDR is set to input. This port write may be used to initialize the data registers and avoid undefined outputs. Refer to Figure 2 for typical port circuitry and to Table 1 for a list of the I/O pin functions.

Table 1. I/O Pin Functions

R/W̄*	DDR	I/O Pin Functions
0	0	The I/O pin is in input mode. Data is written into the output data latch.
0	1	Data is written into the output data latch and output to the I/O pin.
1	0	The state of the I/O pin is read.
1	1	The I/O pin is in an output mode. The output data latch is read.

*R/W̄ is an internal signal.

Under software control, the PC2 pin can become the CPU clock output. If this option is selected, the corresponding DDR bit is automatically set, and bit 2 of port C always reads the output data latch. The other port C pins are not affected by this feature.

Control Register (CTL/ECLK) $07

7	6	5	4	3	2	1	0
0	0	0	0	ECLK	—	—	—
RESET:							
0	0	0	0	0	0	0	0

ECLK — ECLK Control

1 = I/O port function of PC2 is forced to output mode, and PC2 outputs the ECLK CPU clock.

0 = PC2 functions as a regular I/O pin.

FIXED INPUT PORT PROGRAMMING

Port D is a fixed input port that monitors the external pins whenever the A/D is disabled. After reset, all eight bits become digital inputs because all special function drivers are disabled. Port D is always at digital input, whether the A/D is on or off.

NOTE

Any unused inputs and I/O ports should be tied to an appropriate logic level (e.g., either V_{DD} or V_{SS}).

SERIAL PORT (SCI) PROGRAMMING

The SCI uses two or three pins for its functions: RDI for its receive data input, TDO for its transmit data output, and SCLK to output the transmitter clock, if needed.

MEMORY

The MCU is capable of addressing 8192 bytes of memory and I/O registers, as shown in Figure 3. The locations consist of user ROM, user RAM, EEPROM, self-check ROM, control registers, and I/O. The user-defined reset and interrupt vectors are located from $1FF0 to $1FFF.

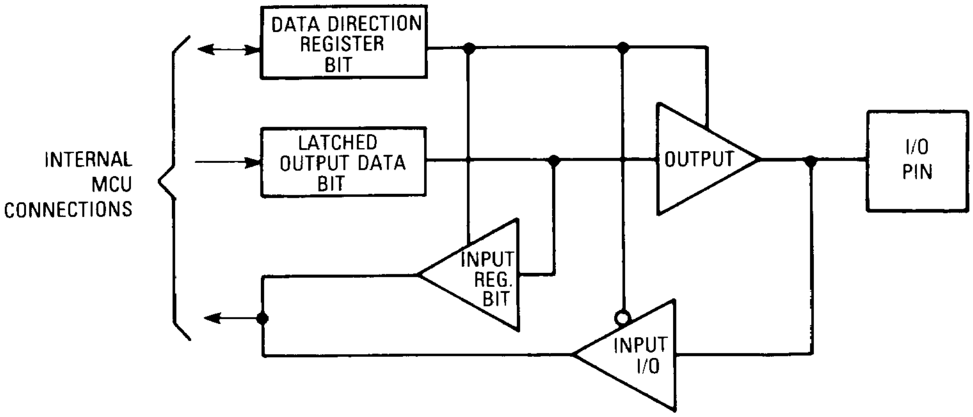


Figure 2. Typical Port I/O Circuit

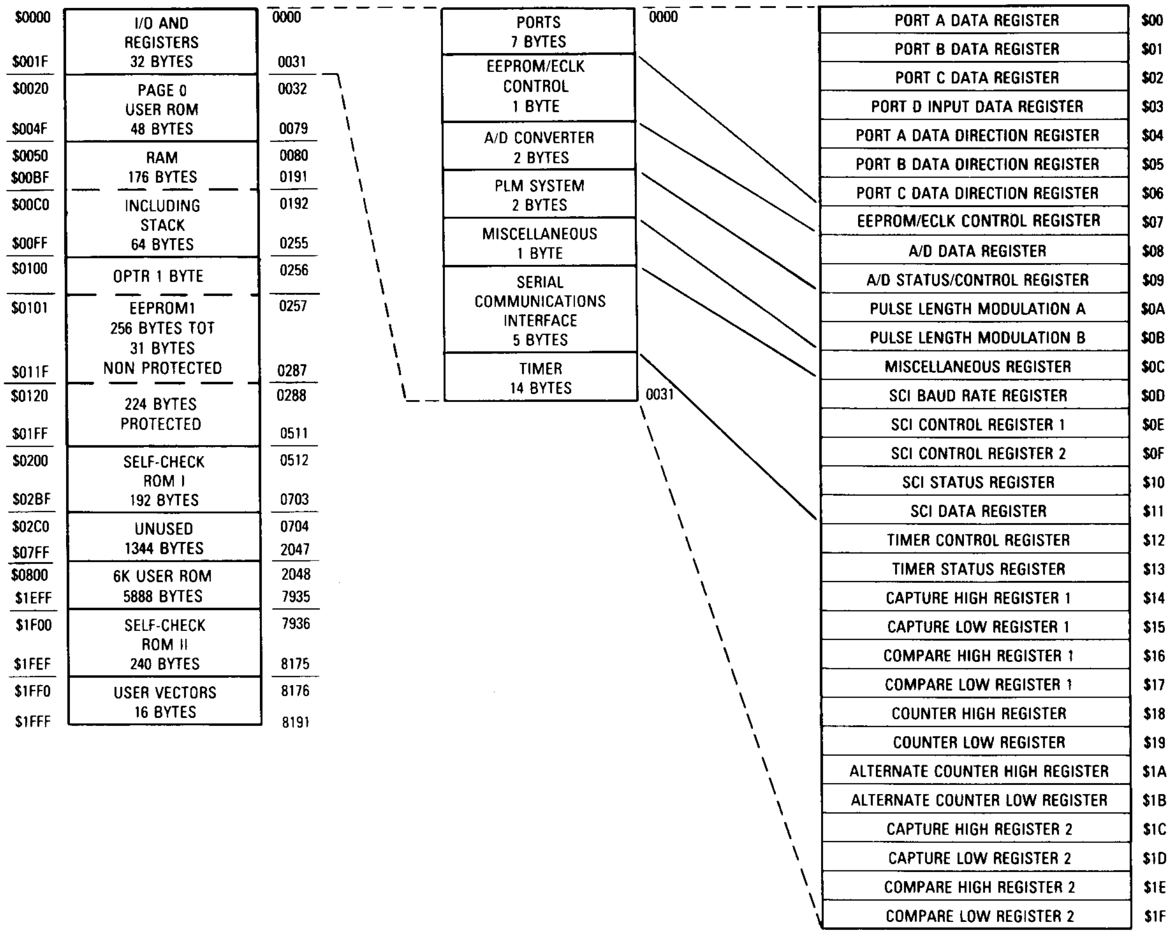

$0000
$001F
$0020
$004F
$0050
$00BF
$00C0
$00FF
$0100
$0101
$011F
$0120
$01FF
$0200
$02BF
$02C0
$07FF
$0800
$1EFF
$1F00
$1FEF
$1FF0
$1FFF
I/O AND REGISTERS 32 BYTES
PAGE 0 USER ROM 48 BYTES
RAM 176 BYTES
INCLUDING STACK 64 BYTES
OPTR 1 BYTE
EEPROM1 256 BYTES TOT 31 BYTES NON PROTECTED
224 BYTES PROTECTED
SELF-CHECK ROM I 192 BYTES
UNUSED 1344 BYTES
6K USER ROM 5888 BYTES
SELF-CHECK ROM II 240 BYTES
USER VECTORS 16 BYTES
0000
0031
0032
0079
0080
0191
0192
0255
0256
0257
0287
0288
0511
0512
0703
0704
2047
2048
7935
7936
8175
8176
8191
PORTS 7 BYTES
EEPROM/ECLK CONTROL 1 BYTE
A/D CONVERTER 2 BYTES
PLM SYSTEM 2 BYTES
MISCELLANEOUS 1 BYTE
SERIAL COMMUNICATIONS INTERFACE 5 BYTES
TIMER 14 BYTES
0000
0031
PORT A DATA REGISTER $00
PORT B DATA REGISTER $01
PORT C DATA REGISTER $02
PORT D INPUT DATA REGISTER $03
PORT A DATA DIRECTION REGISTER $04
PORT B DATA DIRECTION REGISTER $05
PORT C DATA DIRECTION REGISTER $06
EEPROM/ECLK CONTROL REGISTER $07
A/D DATA REGISTER $08
A/D STATUS/CONTROL REGISTER $09
PULSE LENGTH MODULATION A $0A
PULSE LENGTH MODULATION B $0B
MISCELLANEOUS REGISTER $0C
SCI BAUD RATE REGISTER $0D
SCI CONTROL REGISTER 1 $0E
SCI CONTROL REGISTER 2 $0F
SCI STATUS REGISTER $10
SCI DATA REGISTER $11
TIMER CONTROL REGISTER $12
TIMER STATUS REGISTER $13
CAPTURE HIGH REGISTER 1 $14
CAPTURE LOW REGISTER 1 $15
COMPARE HIGH REGISTER 1 $16
COMPARE LOW REGISTER 1 $17
COUNTER HIGH REGISTER $18
COUNTER LOW REGISTER $19
ALTERNATE COUNTER HIGH REGISTER $1A
ALTERNATE COUNTER LOW REGISTER $1B
CAPTURE HIGH REGISTER 2 $1C
CAPTURE LOW REGISTER 2 $1D
COMPARE HIGH REGISTER 2 $1E
COMPARE LOW REGISTER 2 $1F

Figure 3. Memory Map

The shared stack area is used during processing of an interrupt or subroutine call to save the CPU state. The stack pointer decrements during pushes and increments during pulls. Refer to **INTERRUPTS** for additional information.

NOTE

Using the stack area for data storage or temporary work locations requires care to prevent it from being overwritten due to stacking from an interrupt or subroutine call.

EEPROM

The MCU has 256 bytes of byte-erasable EEPROM (255 bytes general purpose and 1-byte option register), located at addresses $0100–$01FF. An internal charge pump, connected to the V_{PP} pin, avoids the necessity of supplying a high voltage for erase and programming. The V_{PP} pin should be left open.

CAUTION

An external high voltage should **not** be applied to this pin.

To provide a higher degree of security for stored data, there is no bulk or row erase.

EEPROM Read Operation

To read data from EEPROM, the E1LAT bit must be zero. When E1LAT is zero, the E1PGM and E1ERA bits are forced to zero, and the 256-byte EEPROM is read as if it were a normal ROM. The V_{PP} charge pump generator is off since E1PGM is zero. If a read is performed while E1LAT is set, data will be read as $FF.

NOTE

When not performing a programming or erase operation on the EEPROM, remain in read mode (E1LAT = 0).

EEPROM Erase Operation

To erase a byte of EEPROM, set E1LAT and E1ERA to one, write to the address to be erased, and set E1PGM for a time t_{ERA1}. After the required erase time, E1LAT must be cleared, which resets E1ERA and E1PGM. To erase a second word, E1LAT must be cleared before it is set, or the erase will have not effect. This procedure is done to increase the security of the stored data. While an erase is being performed, any access to the EEPROM will not be successful. Data written in an erase operation is not used; therefore, its value is not significant. User programs must be running from ROM or RAM since the EEPROM has its address and data buses latched.

EEPROM Programming Operation

To program a byte of EEPROM, set the E1LAT bit, write data to the desired address, and set the E1PGM bit for a time t_{PROG}. After the required programming delay, E1LAT must be cleared, which also resets E1PGM. While a programming operation is being performed, any access to the EEPROM will not be successful.

NOTE

To program a byte correctly, the byte must have been previously erased.

To program a second word, E1LAT must be cleared before it is set, or the programming will have no effect. This procedure is done to increase the security of the stored data. User programs must be running from RAM or ROM since the EEPROM will have its data buses latched.

Control Register (CTL/ECLK) $07

7	6	5	4	3	2	1	0
—	—	—	—	—	E1ERA	E1LAT	E1PGM
RESET:							
0	0	0	0	0	U	0	U

E1ERA — EEPROM Erase
- 1 = An erase will take place if E1LAT and E1PGM are both one.
- 0 = A programming operation will take place if E1LAT and E1PGM are both one.

If E1LAT = 0, E1ERA is held to zero. Once an EEPROM address is selected, E1ERA cannot be changed.

E1LAT — EEPROM Latch Enable
- 1 = Address and data can be latched into the EEPROM for programming or erase operation if E1PGM = 0.
- 0 = Data can be read from the EEPROM, and the E1ERA and E1PGM bits are cleared.

After the programming or erase time, the E1LAT bit must be reset in order to reset the E1ERA and E1PGM bits.

E1PGM — EEPROM Program Mode
- 1 = Charge pump generator is on, and the resulting high voltage is applied to the EEPROM array.
- 0 = Charge pump generator is off.

E1PGM cannot be set before the data is selected; it can only be reset by resetting E1LAT.

The charge pump is not affected by the WAIT mode; thus, WAIT can be used for the erase or programming delay time. If STOP mode is entered, the EEPROM is set to read mode.

The V_{PP1} charge pump generator is normally supplied by the CPU clock, but for very low clocking frequencies, the A/D RC oscillator should be used. See **A/D CONVERTER** for more information.

Options Register (OPTR) $0100

7	6	5	4	3	2	1	0
—	—	—	—	—	—	EE1P	SEC
RESET:							
U	U	U	U	U	U	U	U

EE1P — EEPROM Protect
- 1 = EEPROM not protected.
- 0 = EEPROM addresses from $0120 to $01FF are read only, and attempts to write to this area will be unsuccessful.

When this bit is erased to one, protection remains until the next external or power-up reset occurs.

SEC — High-Security Bit
- 1 = Security not active.
- 0 = EEPROM contents protected because access to test mode is inhibited.

The SEC bit can only be erased to one externally by entering self-check mode, which erases the entire EEPROM. When SEC is changed, the new value has no effect until the next external or power-on reset.

REGISTERS

The MCU contains the registers described in the following paragraphs.

ACCUMULATOR (A)

The accumulator is a general-purpose 8-bit register used to hold operands and results of arithmetic calculations or data manipulations.

7		0
	A	

INDEX REGISTER (X)

The index register is an 8-bit register used for the indexed addressing mode. It contains an 8-bit value that may be added to an 8- or 16-bit immediate value to create an effective address. The index register may also be used as a temporary storage area.

7		0
	X	

PROGRAM COUNTER (PC)

The program counter is a 13-bit register that contains the address of the next instruction to be executed.

12		0
	PC	

STACK POINTER (SP)

The stack pointer is a 13-bit register that contains the address of the next free location on the stack. During an MCU reset or the reset stack pointer (RSP) instruction, the stack pointer is set to location $00FF. The stack pointer is then decremented as data is pushed onto the stack and incremented as data is pulled from the stack.

When accessing memory, the seven most significant bits are permanently set to 0000011. These seven bits are appended to the six least significant register bits to produce an address within the range of $00FF to $00C0. Subroutines and interrupts may use up to 64 (decimal) locations. If 64 locations are exceeded, the stack pointer wraps around and loses the previously stored information. A subroutine call occupies two locations on the stack; an interrupt uses five locations.

12					7		0
0	0	0	0	0	1	1	SP

CONDITION CODE REGISTER (CCR)

The CCR is a 5-bit register in which four bits are used to indicate the results of the instruction just executed. These bits can be individually tested by a program, and specific actions can be taken as a result of their state. Each bit is explained in the following paragraphs.

H	I	N	Z	C

Half Carry (H)

This bit is set during ADD and ADC operations to indicate that a carry occurred between bits 3 and 4.

Interrupt (I)

When this bit is set, the timer and external interrupt is masked (disabled). If an interrupt occurs while this bit is set, the interrupt is latched and processed as soon as the interrupt bit is cleared.

Negative (N)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logic one).

Zero (Z)

When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

Carry/Borrow (C)

When set, this bit indicates that a carry or borrow out of the arithmetic logical unit (ALU) occurred during the last arithmetic operation. This bit is also affected during bit test and branch instructions and during shifts and rotates.

SELF-CHECK

The self-check capability provides the ability to determine if the device is functional. Table 2 shows how self-check mode is entered. Self-check is performed using the circuit shown in Figure 4. Port C pins PC3–PC0 are monitored for the self-check results. After reset, the following tests are performed automatically:

I/O — Exercise of ports A, B, C, and D
RAM — Counter test for each RAM byte
ROM — Exclusive OR with odd ones parity result
Timer — Tracks counter register and checks ICF1, ICF2, OCF1, OCF2, and TOV flag
Interrupts — Tests external, timer, and SCI interrupts
SCI — Transmission test; checks RDRF, TDRE, TC, and FE flags
A/D — Checks A/D on internal channels: V_{RL}, V_{RH}, and $(V_{RL}+V_{RH})/2$
EEPROM — Optional. Performs write/erase of the 256-byte EEPROM and then deactivates the security bit.
PLM — Checks basic PLM function
Watchdog System — Checks watchdog function

Self-check results (using the LEDs as monitors) are shown in Table 3. The following subroutines are available to the user and do not require any external hardware.

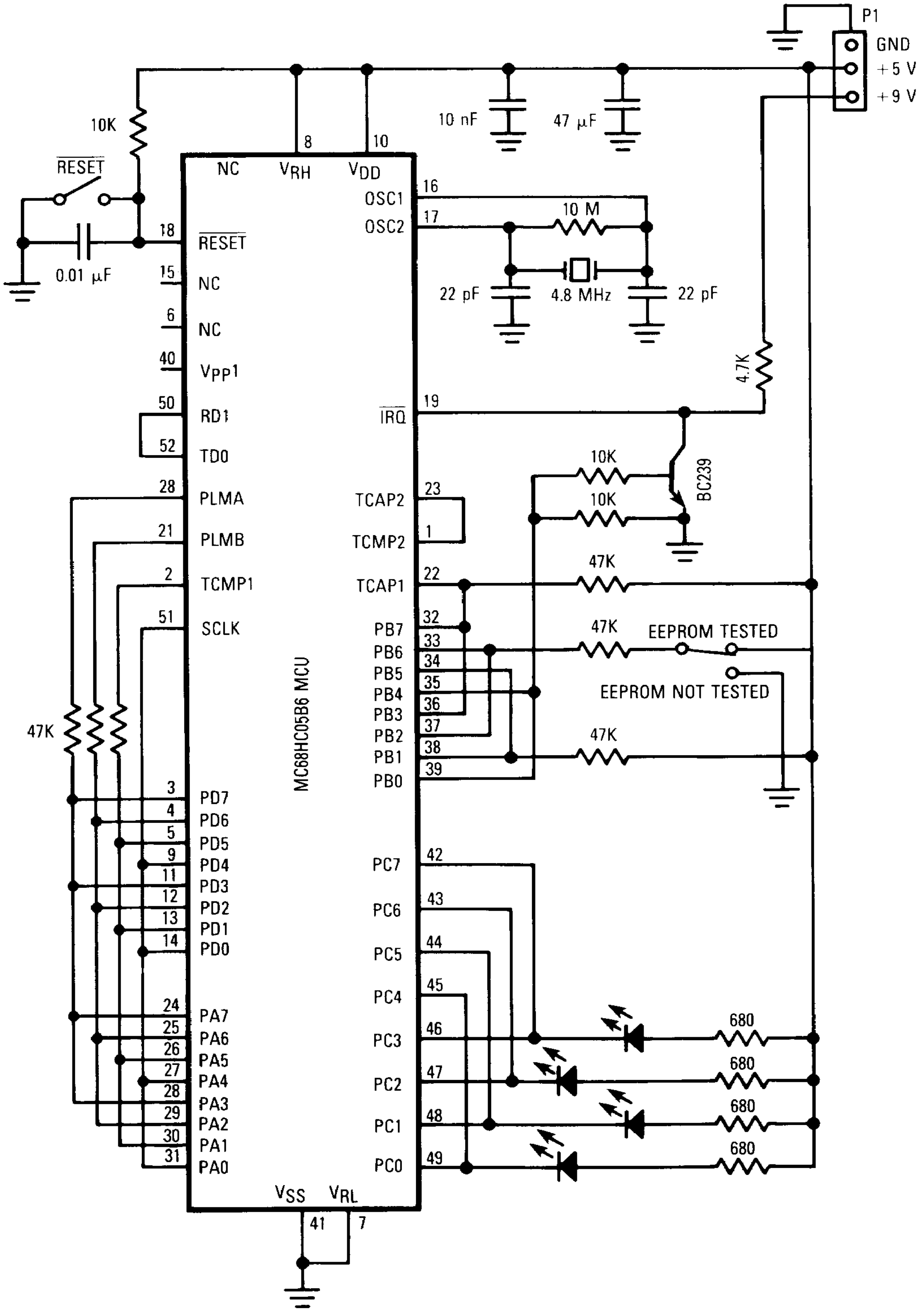
Table 2. Mode of Operation Selection

$\overline{\text{RESET}}$ Pin	$\overline{\text{IRQ}}$ Pin	TCAP1 Pin	Mode
_/¯	V_{SS} to V_{DD}	V_{SS} to V_{DD}	Normal
_/¯	+9 Volts	V_{DD}	Self-Check
V_{SS}	V_{SS} to V_{DD}	V_{SS} to V_{DD}	Reset Condition

Table 3. Self-Check Results

PC3	PC2	PC1	PC0	Remarks
1	0	0	1	Bad Port
0	1	1	0	Bad Port
1	0	1	0	Bad RAM
1	0	1	1	Bad ROM
1	1	0	0	Bad Timer
1	1	0	1	Bad SCI
1	1	1	0	Bad A/D
0	0	0	0	Bad EEPROM
0	0	0	1	Bad PLM
0	0	1	0	Bad Interrupts
0	0	1	1	Bad Watchdog
Flashing				Good Device
All Others				Bad Device, Bad Port, etc.

0 indicates LED is on; 1 indicates LED is off.



NOTE: Pin numbers are valid for 52-pin PLCC package only.

Figure 4. Self-Check Circuit Schematic Diagram

RAM CHECK SUBROUTINE

This subroutine returns with the Z bit cleared if any error is detected; otherwise, the Z bit is set. The stack pointer must be set to \$FF. The stack pointer must be set to \$FF. The RAM check subroutine is called at location \$021E. A counter test is done on each location from address \$50 to \$FD. Each location is made to count from \$00 to \$00 again. Locations \$FE and \$FF are assumed to contain the return address. Upon return to the user's program, if the test passed, X = \$00, A = \$00, and RAM locations \$0050 and \$00FD contain \$01.

NOTE

The watchdog system is turned on when calling this subroutine.

A/D CONVERTER CHECK SUBROUTINE

This subroutine returns with the Z bit cleared if any error is detected; otherwise, the Z bit is set. The subroutine is called at location \$1FAA with X = \$00 and A/D STAT/CTRL (address \$09) = \$20 (ADON = 1 for more than 100 µs and channel PD0 selected). Conversion is done on three of the internal channels: V_{RH}, V_{RL}, and $(V_{RL} + V_{RH})/2$. The result of these conversions is verified at ±1 LSB. Upon return to the user's program, if the test passed, X = \$09, A = \$00 or \$01.

ROM CHECKSUM SUBROUTINE

This subroutine returns with the Z bit cleared if any error is detected; otherwise, the Z bit is set. The ROM checksum subroutine is called at location \$0232 with RAM location \$0053 equal to \$01 and A = 0. A short routine is set up and executed in RAM to compute a checksum of the entire ROM pattern. RAM locations \$0050 through \$0053 are overwritten. Upon return to the user's program, if the test passed, X = 0, A = 0.

NOTE

The A/D and the watchdog system are turned on when calling this subroutine.

RESETS

The MCU can be reset two ways: by initial power-up (POR) and by the external reset input ($\overline{\text{RESET}}$). The $\overline{\text{RESET}}$ input consists mainly of a Schmitt trigger that senses the $\overline{\text{RESET}}$ line logic level.

POWER-ON RESET

An internal reset is generated on power-up to allow the internal clock generator to stabilize. The power-on reset is strictly for power turn-on conditions and should not be used to detect a drop in the power supply voltage. There is a delay (t_{PORL}) after the oscillator becomes active. If the $\overline{\text{RESET}}$ pin is low at the end of t_{PORL}, the MCU will remain in the reset condition until $\overline{\text{RESET}}$ goes high. A mask option allows t_{PORL} to be either 16 or 4064 internal processor clock cycles (t_{cyc}.

EXTERNAL RESET INPUT

The MCU is reset when a logic zero is applied to the $\overline{\text{RESET}}$ input for a period of one and one-half machine cycles (t_{cyc}).

Miscellaneous Register (0C)

	7	6	5	4	3	2	1	0
	POR	INTP	INTN	INTE	SFA	SFB	SM	WDOG
RESET:	U	0	0	1	0	0	0	0

POR — Power-On Reset

1 = The reset occurring is a power-on, not external, reset

0 = Power-on reset not in progress

INTP — External Interrupt Positive

Allows a choice of $\overline{\text{IRQ}}$ sensitivity, with INTN. See Table 4.

INTN — External Interrupt Negative

Allows a choice of $\overline{\text{IRQ}}$ sensitivity, with INTP. See Table 4.

INTE — External Interrupt Enable

Allows the user to enable or disable the external interrupt function.

SFA — Slow/Fast Selection for PLMA

1 = Slow speed used for PLMA (4096 times the timer clock period)

0 = Fast speed used for PLMA (256 times the timer clock period). See **PULSE-LENGTH D/A CONVERTERS**

SFB — Slow/Fast Selection for PLMB

1 = Slow speed used for PLMB (4096 times the timer clock period)

0 = Fast speed used for PLMB (256 times the timer clock period). See **PULSE-LENGTH D/A CONVERTERS**

SM — Slow Mode

1 = System runs at 1/16th the normal clock rate ($f_{OSC}/32$)

0 = System runs at normal clock rate ($f_{OSC}/2$)

WDOG — Watchdog Counter System

1 = Watchdog counter system enabled

0 = Watchdog counter system disabled

NOTE

The reset generated by the watchdog timer is a system reset; thus, the watchdog is disabled after a watchdog reset.

Table 4. External Interrupt Options

INTP	INTN	External Interrupt Options
0	0	Negative Edge and Low-Level Sensitive
0	1	Negative Edge Only
1	0	Positive Edge Only
1	1	Positive and Negative Edge Sensitive

Slow Mode

The slow mode function is controlled by the SM bit in the miscellaneous register (0C). In slow mode (SM = 1), an extra divide-by-sixteen circuit is added between the oscillator and the internal clock driver. This slows all functions by a factor of 16 (including SCI, A/D, and timer), which is particularly useful in WAIT mode. SM is cleared by external or power-on reset and by STOP mode.

NOTE

> If slow mode is enabled while using the A/D, the internal A/D RC oscillator should be turned on.

Watchdog System

The watchdog counter is driven by the 1024 prescaler in the timer and, unless the counter is reset, generates a system reset when it reaches its maximum count (1024 × 8).

A mask option is available that provides two methods of enabling the watchdog timer. In the first option, the watchdog system is controlled by the WDOG bit in the miscellaneous register (0C). Writing a one to the bit starts the watchdog or, if it is already started, resets the counter to zero. Writing a zero has no effect; the WDOG bit can only be cleared by external or power-on reset. In the second option, the watchdog timer is always enabled following reset.

A second mask option determines the watchdog timer function during WAIT. The watchdog timer can remain active during WAIT, and can cause a reset if the device remains in WAIT longer than the watchdog timeout period. Alternatively, the watchdog timer suspends operation during WAIT and resets its count, resuming normal operation following reset.

INTERRUPTS

The MCU can be interrupted four different ways: the three maskable hardware interrupts (IRQ, SCI, and timer) and the nonmaskable software interrupt instruction (SWI).

Interrupts cause the processor to save register contents on the stack and to set the interrupt mask (I bit) to prevent additional interrupts. The RTI instruction causes the register contents to be recovered from the stack and normal processing to resume. The stacking order is shown in Figure 5.

Unlike RESET, hardware interrupts do not cause the current instruction execution to be halted but are considered pending until the current instruction is complete.

NOTE

> The current instruction is the one already fetched and being operated on.

When the current instruction is complete, the processor checks all pending hardware interrupts. If unmasked (I bit clear) and if the corresponding interrupt enable bit is set, the processor proceeds with interrupt processing; otherwise, the next instruction is fetched and executed.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed the same as any other instruction, regardless of the I-bit state. Refer to Figure 6 for the reset and interrupt instruction processing sequence.

TIMER INTERRUPT

There are three different timer interrupt flags that cause a timer interrupt whenever they are set and enabled. The interrupt flags are in the timer status register (TSR), and the enable bits are in the timer control register (TCR). Refer to **TIMER** for more information.

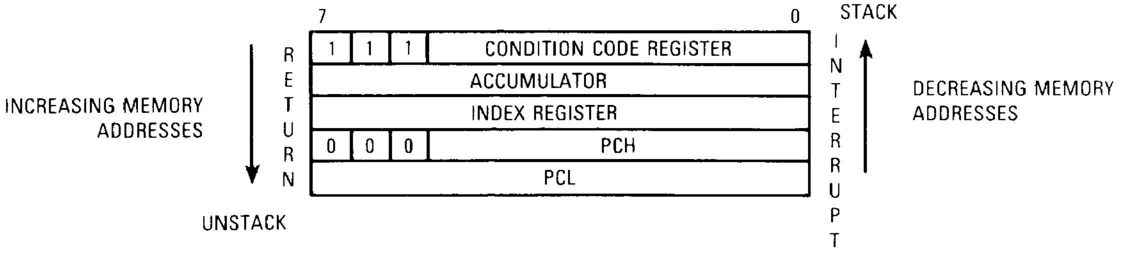
EXTERNAL INTERRUPT

If the interrupt mask bit (I bit) of the CCR is set, all interrupts are disabled. Clearing the I bit enables the external interrupt. The external interrupt is internally synchronized and then latched on the falling edge of IRQ. The action of the external interrupt is identical to the timer interrupt with the exception that the interrupt request input at IRQ is latched internally and the service routine address is specified by the contents of $1FFA and $1FFB.

Four options are available for interrupt triggering sensitivity:

- Negative edge and low level
- Negative edge only
- Positive edge only
- Positive and negative edge

See **Miscellaneous Register (0C)** for further information.



NOTE: Since the Stack Pointer decrements during pushes, the PCL is stacked first, followed by PCH, etc. Pulling from the stack is in the reverse order.

Figure 5. Interrupt Stacking Order

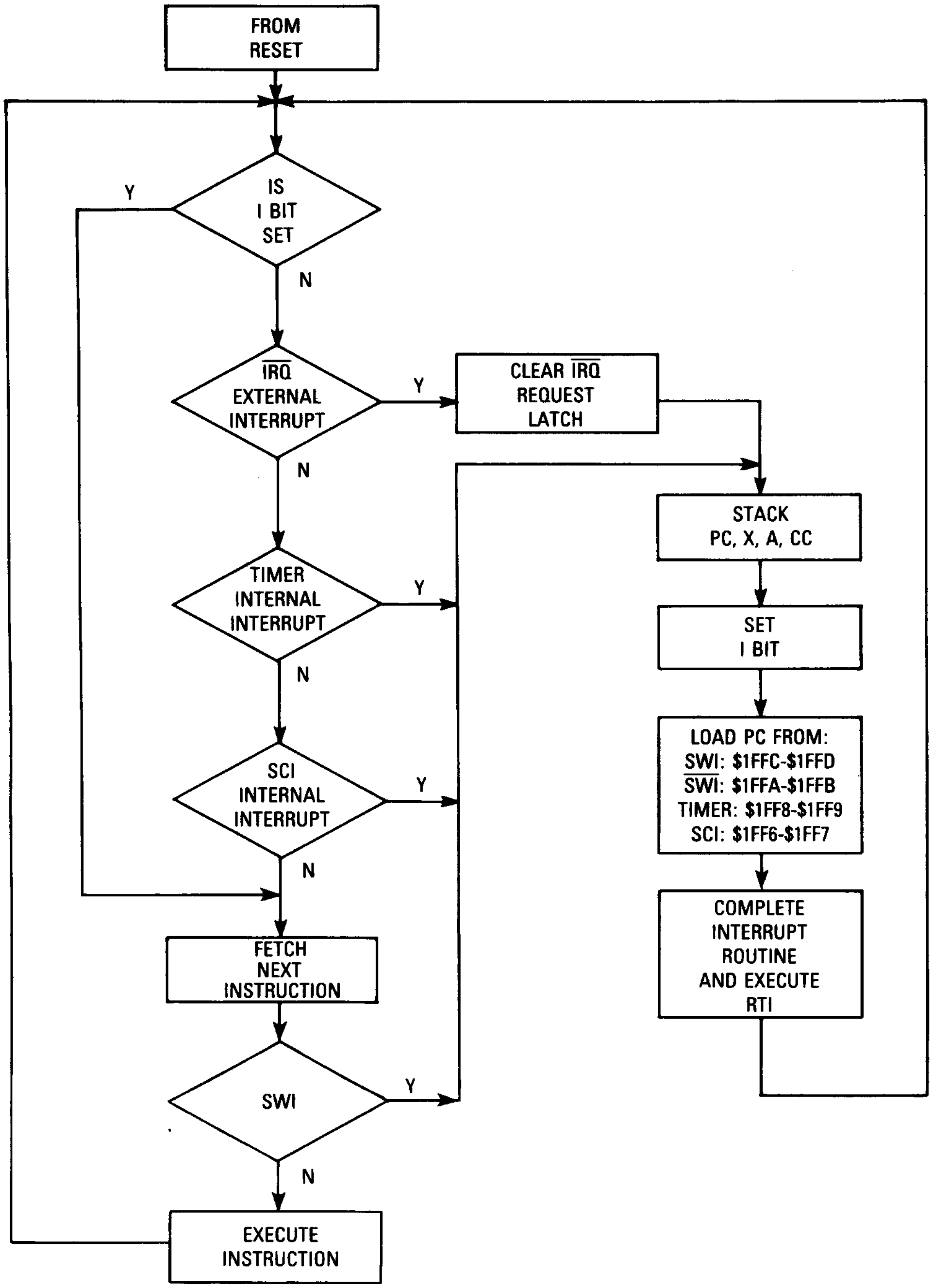
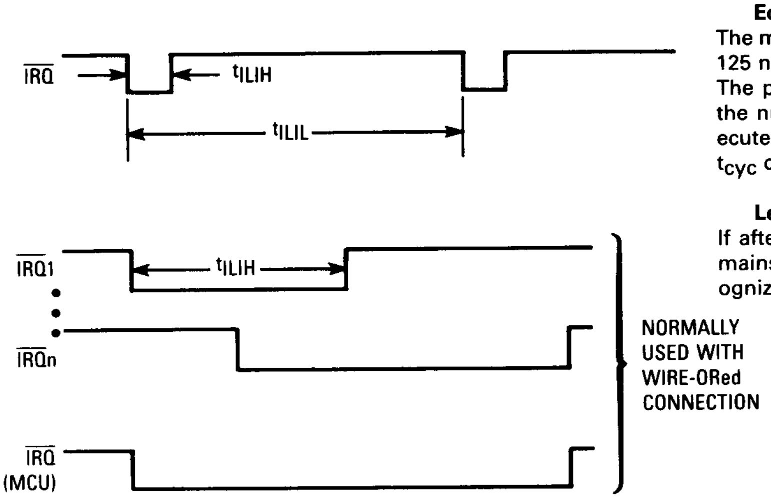


Figure 6. Reset and Interrupt Processing Flowchart

Figure 7 shows a mode timing diagram for the interrupt line. The timing diagram shows two treatments of the interrupt line to the processor. The first method shows a single pulse on the interrupt line spaced far enough apart to be serviced. The minimum time between pulses is a function of the length of the interrupt service. Once a pulse occurs, the next pulse should not occur until an RTI occurs. This time (t_{ILIL}) is obtained by adding 21 instruction cycles to the total number of cycles it takes to complete the service routine (not including the RTI instruction). The second method shows many interrupt lines "wire-ORed" to form the interrupts at the processor. If the interrupt line remains low after servicing an interrupt, then the next interrupt is recognized.

NOTE

The internal interrupt latch is cleared in the first part of the interrupt service routine; therefore, one external interrupt pulse could be latched and serviced as soon as the I bit is cleared.



Edge-Sensitive Trigger Condition

The minimum pulse width (t_{ILIH}) is either 125 ns (V_{DD}=5 V) or 250 ns (V_{DD}=3 V). The period t_{ILIL} should not be less than the number of t_{cyc} cycles it takes to execute the interrupt service routine plus 21 t_{cyc} cycles.

Level-Sensitive Trigger Condition

If after servicing an interrupt the $\overline{IRQ}$ remains low, then the next interrupt is recognized.

Figure 7. External Interrupt Mode Diagram

SOFTWARE INTERRUPT (SWI)

The SWI is an executable instruction that is executed regardless of the state of the I bit in the CCR. If the I bit is zero, SWI executes after the other interrupts. The SWI operation is similar to the hardware interrupts. The interrupt service routine address is specified by the contents of memory locations $1FFC and $1FFD.

SCI INTERRUPTS

An interrupt in the SCI occurs when one of the interrupt flag bits in the serial communications status register is set, provided the I bit in the CCR is clear and the enable bit in the serial communications control register 2 is set. Software in the serial interrupt service routine must determine the cause and priority of the SCI interrupt by examining the interrupt flags and status bits in the SCI status register.

LOW-POWER MODES

STOP

The STOP instruction places the MCU in its lowest power consumption mode. In the STOP mode, the internal oscillator is turned off, halting all internal processing including timer, SCI, and A/D operation (refer to Figure 8).

During the STOP mode, the TCR bits are altered to remove any pending timer interrupt request and to disable any further timer interrupts. The timer prescaler is cleared. The I bit in the CCR is cleared to enable external interrupts. All other registers and memory remain unaltered. All input/output lines remain unchanged. The processor can be brought out of the STOP mode only by an external interrupt or reset.

SCI during STOP Mode

When the MCU enters the STOP mode, the baud rate generator stops, halting all SCI activity. If the STOP instruction is executed during a transmitter transfer, that

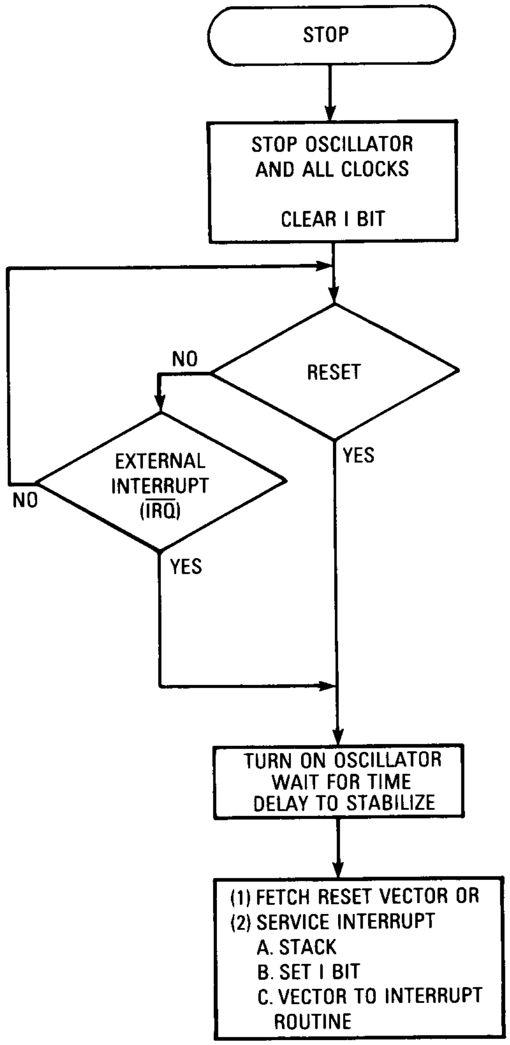

STOP
STOP OSCILLATOR AND ALL CLOCKS
CLEAR I BIT
RESET
NO
YES
EXTERNAL INTERRUPT ($\overline{IRQ}$)
NO
YES
TURN ON OSCILLATOR WAIT FOR TIME DELAY TO STABILIZE
(1) FETCH RESET VECTOR OR
(2) SERVICE INTERRUPT
A. STACK
B. SET I BIT
C. VECTOR TO INTERRUPT ROUTINE

Figure 8. STOP Function Flowchart

transfer is halted. If a low input to the $\overline{IRQ}$ pin is used to exit STOP mode, the transfer resumes. If the SCI receiver is receiving data and the STOP mode is entered, received data sampling stops because the baud rate generator stops, and all subsequent data is lost. For these reasons, all SCI transfers should be in the idle state when the STOP instruction is executed.

Watchdog during STOP Mode

The STOP instruction is inhibited when the watchdog system is enabled. If a STOP instruction is executed while the watchdog is enabled, a reset occurs that resets the entire MCU.

EEPROM during STOP Mode

The EEPROM is set to read, and the V_{PP1} high-voltage charge pump generator is disabled when stop mode is entered.

PLM during STOP Mode

When the MCU enters stop mode, the PLM outputs remain at their particular level. If power-on or external reset causes the exit from stop mode, the register values are forced to $00.

A/D Converter during STOP Mode

When stop mode is entered with the A/D converter turned on, the A/D clocks are stopped and the A/D converter is disabled for the duration of stop mode, including the t_{PORL} startup time. If the A/D RC oscillator is used, it will also be disabled.

When leaving STOP mode, after the t_{PORL} startup time, the A/D converter and A/D RC oscillator resume regular operation. However, a time t_{ADON} is required for the current sources to stabilize. During t_{ADON}, A/D conversion results may be inaccurate.

WAIT

The WAIT instruction places the MCU in a low-power consumption mode, but the WAIT mode consumes more power than the STOP mode. All CPU action and the watchdog system are suspended, but the timer, SCI, PLM, and A/D remain active (refer to Figure 9). An interrupt from the timer, SCI, or an IRQ can cause the MCU to exit the WAIT mode.

During the WAIT mode, the I bit in the CCR is cleared to enable interrupts. All other registers, memory, and input/output lines remain in their previous state. The timer may be enabled to allow a periodic exit from the WAIT mode.

To achieve proper operation and reduce power consumption, the following points should be set as desired before entering wait mode:

- Timer interrupt enable bits
- A/D control bits
- EEPROM control bits
- SCI enable bits and interrupt enable bits

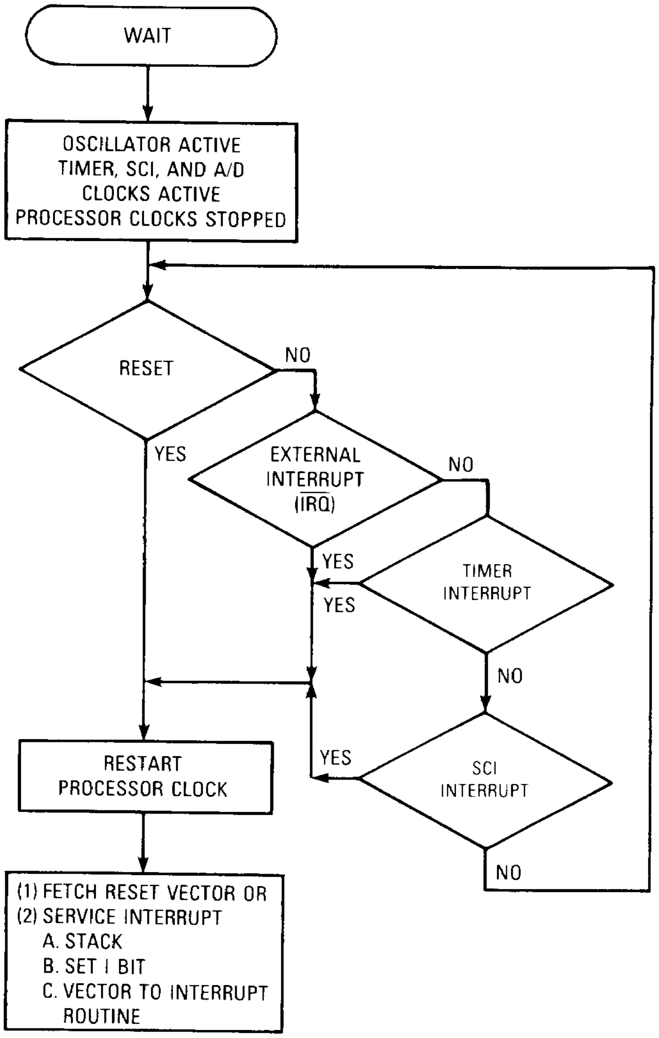


Figure 9. WAIT Function Flowchart

TIMER

The timer consists of a 16-bit, software-programmable counter driven by a fixed divide-by-four prescaler. This timer can be used for many purposes, including input waveform measurements of two input signals while simultaneously generating two output waveforms. Pulse widths can vary from several microseconds to many seconds. The programmable timer works in conjunction with the PLM system to execute two 8-bit D/A PLM conversions, with a choice of two repetition rates. Refer to Figure 10 for a timer block diagram.

Because the timer has a 16-bit architecture, each specific functional segment (capability) is represented by two registers. These registers contain the high and low byte of that functional segment. Generally, accessing the low byte of a specific timer function allows full control of that function; however, an access of the high byte inhibits that specific timer function until the low byte is also accessed.

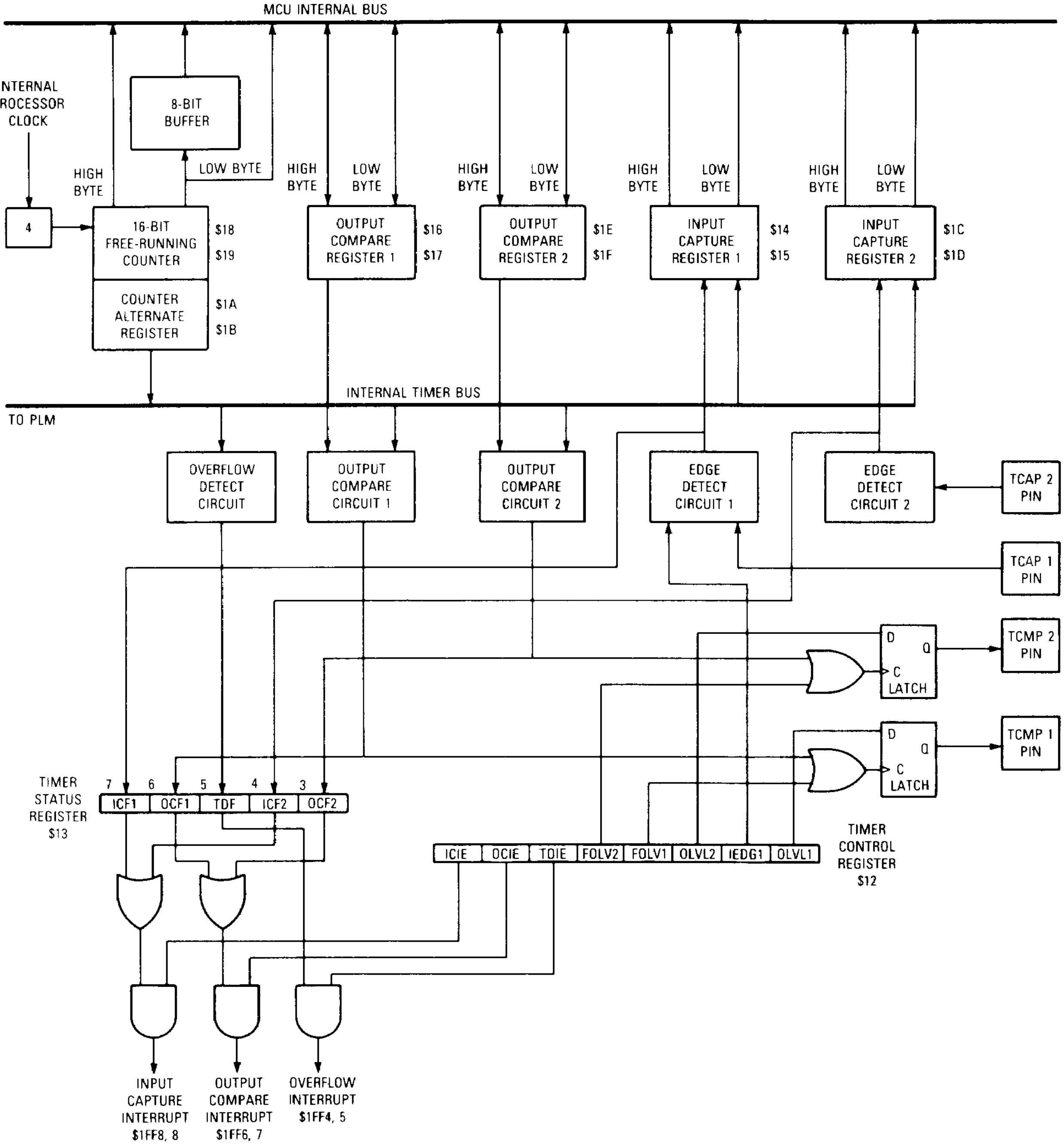


Figure 10. Timer Block Diagram

NOTE

The I bit in the CCR should be set while manipulating both the high and low byte register of a specific timer function to ensure that an interrupt does not occur.

COUNTER

The key element in the programmable timer is a 16-bit, free-running counter or counter register, preceded by a prescaler that divides the internal processor clock by four. The prescaler gives the timer a resolution of 2.0 microseconds if the internal bus clock is 2.0 MHz. The counter is incremented during the low portion of the internal bus clock. Software can read the counter at any time without affecting its value.

The double-byte, free-running counter can be read from either of two locations, $18–$19 (counter register) or $1A–$1B (counter alternate register). A read from only the least significant byte (LSB) of the free-running counter

($19, $1B) receives the count value at the time of the read. If a read of the free-running counter or counter alternate register first addresses the most significant byte (MSB) ($18, $1A), the LSB ($19, $1B) is transferred to a buffer. This buffer value remains fixed after the first MSB read, even if the user reads the MSB several times. This buffer is accessed when reading the free-running counter or counter alternate register LSB ($19 or $1B) and, thus, completes a read sequence of the total counter value. In reading either the free-running counter or counter alternate register, if the MSB is read, the LSB must also be read to complete the sequence.

The counter alternate register differs from the counter register in one respect: a read of the counter register LSB can clear the timer overflow flag (TOF). Therefore, the counter alternate register can be read at any time without the possibility of missing timer overflow interrupts due to clearing of the TOF.

The free-running counter is configured to $FFFC during reset and is always a read-only register. During a power-on reset, the counter is also preset to $FFFC and begins running after the oscillator start-up delay. Because the free-running counter is 16 bits preceded by a fixed divide-by-four prescaler, the value in the free-running counter repeats every 262,144 internal bus clock cycles. When the counter rolls over from $FFFF to $0000, the TOF bit is set. An interrupt can also be enabled when counter rollover occurs by setting its interrupt enable bit (TOIE).

NOTE

Since the PLM system uses the timer counter, PLM results will be affected when resetting this counter.

OUTPUT COMPARE REGISTERS

There are two output compare registers: output compare register 1 (OCR1) and output compare register 2 (OCR2). The output compare registers can be used for several purposes, such as controlling an output waveform or indicating when a period of time has elapsed. All bits are readable and writable and are not altered by the timer hardware or reset. If the compare function is not needed, the four bytes of the output compare registers can be used as storage locations.

NOTE

The same output compare interrupt enable bit is used for the two output compares.

Output Compare Register 1

The output compare register 1 (OCR1) is a 16-bit register, which is made up of two 8-bit registers at locations $16 (most significant byte) and $17 (least significant byte).

The output compare register contents are continually compared with the contents of the free-running counter and, if a match is found, the corresponding output compare flag (OCF1, bit 6 of timer status register $13) is set, and the corresponding output level (OLVL1) bit is clocked to pin TCMP1. The output compare register values and the output level bit should be changed after each successful comparison to control an output waveform or establish a new elapsed timeout. An interrupt can also accompany a successful output compare, provided the corresponding interrupt enable bit, OCIE, is set.

After a processor write cycle to the output compare register 1 containing the most significant byte ($16), the output compare 1 function is inhibited until the least significant byte ($17) is also written. The user must write both bytes (locations) if the most significant byte is written first. A write made only to the least significant byte ($17) will not inhibit the compare function. The free-running counter is updated every four internal bus clock cycles. The minimum time required to update the output compare register 1 is a function of the program rather than the internal hardware.

The processor can write to either byte of the output compare register 1 without affecting the other byte. The output level (OLVL1) bit is clocked to the corresponding output level register and then to the TCMP1 pin, regardless of whether the output compare flag (OCF1) is set or clear.

Output Compare Register 2

The output compare register 2 (OCR2) is a 16-bit register, which is made up of two 8-bit registers at locations $1E (most significant byte) and $1F (least significant byte). The function of OCR2 is identical to OCR1, requiring only changes of the register locations and control bits in the timer status register ($13) to make the OCR1 description apply to OCR2.

SOFTWARE FORCE COMPARE

The MCU provides a force compare capability to facilitate fixed frequency generation as well as other applications. Bit 3 (FOLV1 for OCR1) and bit 4 (FOLV2 for OCR2) in the timer control register ($12) implement this force compare. Writing a one to these bits causes the OLVL1 or OLVL2 values to be copied to the respective output registers (TCMP1 or TCMP2 pins). Internal logic allows a single instruction to change OLVL1 and OLVL2 and cause a forced compare with the new values of OLVL1 and OLVL2.

NOTE

A software force compare, which affects the corresponding output pin TCMP1 or TCMP2, does not affect the compare flag; thus, it does not generate an interrupt.

INPUT CAPTURE REGISTERS

There are two input capture registers: input capture register 1 (ICR1) and input capture register 2 (ICR2).

NOTE

The same input capture interrupt enable bit (ICIE) is used for the two input capture registers.

Input Capture Register 1

Two 8-bit registers that make up the 16-bit input capture register 1 (ICR1) are read-only and are used to latch the value of the free-running counter after the corresponding input capture edge detector senses a defined transition. The level transition that triggers the counter transfer is defined by the corresponding input edge bit (IEDG1). Reset does not affect the contents of the input capture register.

The result obtained by an input capture will be one more than the value of the free-running counter on the rising edge of the internal bus clock preceding the external transition. This delay is required for internal synchronization. Resolution is one count of the free-running counter, which is four internal-bus clock cycles.

The free-running counter contents are transferred to the input capture register on each proper signal transition, regardless of whether the input capture flag (ICF1) is set or clear. The input capture register always contains the free-running counter value, which corresponds to the most recent input capture.

After a read of the input capture register 1 ($14) most significant byte, the counter transfer is inhibited until the least significant byte ($15) is also read. This characteristic causes the time used in the input capture software routine and its interaction with the main program to determine the minimum pulse period.

A read of the input capture register 1 least significant byte ($15) does not inhibit the free-running counter transfer since they occur on opposite edges of the internal bus clock.

Input Capture Register 2

The input capture register 2 (ICR2) is a 16-bit register that is composed of two 8-bit registers at locations $1C (most significant byte) and $1D (least significant byte). Input capture register 2 functions identically to input capture register 1, except that only negative edge sensitivity is available. By substituting the appropriate bits in the timer status register ($13) and substituting register locations, the ICR1 description applies to ICR2.

TIMER CONTROL REGISTER (TCR) $12

The TCR is an 8-bit read/write register, illustrated below with a definition of each bit.

7	6	5	4	3	2	1	0
ICIE	OCIE	TOIE	FOLV2	FOLV1	OLVL1	IEDG1	OLVL1
RESET:							
0	0	0	0	0	0	U	0

U = Uneffected by RESET

ICIE — Input Capture Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

OCIE — Output Compare Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

TOIE — Timer Overflow Interrupt Enable
1 = Interrupt enabled
0 = Interrupt disabled

FOLV2 — Force Output Compare 2
1 = Forces the OLVL2 bit to the corresponding output latch
0 = No effect

FOLV1 — Force Output Compare 1
1 = Forces the OLVL1 bit to the corresponding output latch
0 = No effect

OLVL2 — Output Level 2
1 = The value of the output level 2 bit, which is copied to the output level latch by the next successful output compare 2, appears at TCMP2
0 = No effect

IEDG1 — Input Edge
Value of input edge determines which level transition on TCAP1 pin will trigger free-running counter transfer to the input capture register.
1 = Positive edge
0 = Negative edge

OLVL1 — Output Level 1
Value of output level 1, which is copied into output level register by the next successful output compare 1, will appear on the TCMP1 pin.
1 = High output
0 = Low output

TIMER STATUS REGISTER (TSR) $13

The TSR is a read-only register containing three status flag bits. Bits 0–4 always read zero.

7	6	5	4	3	2	1	0
ICF1	OCF1	TOF	ICF2	OCF2	—	—	—
RESET:							
U	U	U	U	U	—	—	—

ICF1 — Input Capture Flag 1
1 = Flag set when selected polarity edge is sensed by input capture edge detector
0 = Flag cleared when TSR and input capture 1 low register ($15) are accessed

OCF1 — Output Capture Flag 1
1 = Flag set when output compare register contents match the free-running counter contents
0 = Flag cleared when TSR and output compare 1 low register ($17) are accessed

TOF — Timer Overflow Flag
1 = Flag set when free-running counter transition from $FFFF to $0000 occurs
0 = Flag cleared when TSR and counter low register ($19) are accessed

ICF2 — Input Capture Flag 2
1 = Flag set when selected polarity edge is sensed by input capture 2 edge detector
0 = Flag cleared when TSR and input capture 2 low register ($1D) are accessed

OCF2 — Output Capture Flag 2
1 = Flag set when output compare register contents match the free-running counter contents
0 = Flag cleared when TSR and output compare low register 2 ($1F) are accessed

Bits 0–2 — Not Used
Can read either zero or one.

TIMER DURING WAIT MODE

The CPU clock halts during the WAIT mode, but the timer remains active. An interrupt from the timer causes the processor to exit the WAIT mode.

TIMER DURING STOP MODE

In the STOP mode, the timer stops counting and holds the last count value if STOP is exited by an interrupt. If reset is used, the counter is forced to $FFFC. During STOP, if at least one valid input capture edge occurs at the TCAP pin, the input capture detect circuit is armed. This does not set any timer flags or wake up the MCU, but when the MCU does wake up, there is an active input capture flag and data from the first valid edge that occurred during the STOP mode. If reset is used to exit STOP mode, then no input capture flag or data remains, even if a valid input capture edge occurred.

Accessing the timer status register satisfies the first condition required to clear status bits. The remaining step is to access the register corresponding to the status bit. A problem can occur when using the timer overflow function and reading the free-running counter at random times to measure an elapsed time. Without incorporating the proper precautions into software, the timer overflow flag could unintentionally be cleared if 1) the timer status register is read or written when TOF is set, and 2) the least significant byte of the free-running counter is read but not for the purpose of servicing the flag.

The counter alternate register at address $1A and $1B contains the same value as the free-running counter (at address $18 and $19); therefore, this alternate register can be read at any time without affecting the timer overflow flag in the timer status register.

SERIAL COMMUNICATIONS INTERFACE

A full-duplex asynchronous SCI is provided with a standard NRZ format and a variety of baud rates. The SCI transmitter and receiver are functionally independent but use the same data format and baud rate prescaler. The terms baud and bit rate are used synonymously in the following description.

SCI TWO-WIRE SYSTEM FEATURES

- Standard NRZ (mark/space) format
- Advanced error detection method includes noise detection for noise duration of up to one-sixteenth bit time
- Full-duplex operation (simultaneous transmit and receive)
- Software programmable for one of 32 different baud rates
- Different baud rates possible for transmit and receive
- Software-selectable word length (eight- or nine-bit words)
- Separate transmitter and receiver enable bits
- SCI may be interrupt driven
- Four separate interrupt conditions

SCI RECEIVER FEATURES

- Receiver wake-up function (idle or address bit)
- Idle line detect
- Framing error detect
- Noise detect
- Overrun detect
- Receiver data register full flag

SCI TRANSMITTER FEATURES

- Transmit data register empty flag
- Transmit complete flag
- Break send

Any SCI two-wire system requires receive data in (RDI) and transmit data out (TDO).

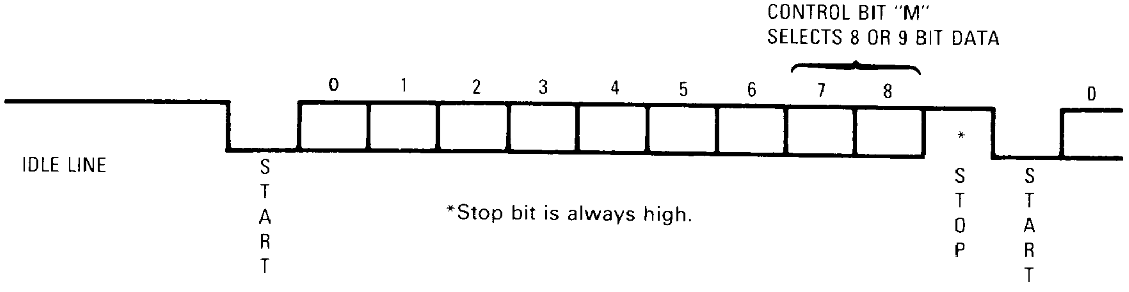
DATA FORMAT

Receive data in (RDI) or transmit data out (TDO) is the serial data presented between the internal data bus and the output pin (TDO) and between the input pin (RDI) and the internal data bus. Data format is as shown for the NRZ in Figure 11.

WAKE-UP FEATURE

In a typical multiprocessor configuration, the software protocol will usually identify the addressee(s) at the beginning of the message. To permit uninterested MPUs to ignore the remainder of the message, a wake-up feature is included, whereby all further SCI receiver flag (and interrupt) processing can be inhibited until its data line returns to the idle state. An SCI receiver is re-enabled by an idle string of at least ten (or eleven) consecutive ones. Software for the transmitter must provide for the required idle string between consecutive messages and prevent it from occurring within messages.

A second wake-up method is available in which sleeping SCI receivers can be awakened by a logic one in the high-order bit of a received character.



*Stop bit is always high.

Figure 11. Data Format

RECEIVE DATA IN

Receive data in (RDI) is the serial data which is presented from the input pin via the SCI to the receive data register (RDR). While waiting for a start bit, the receiver samples the input at a rate 16 times higher than the set baud rate. This increased rate is referred to as the RT rate. When the input (idle) line is detected low, it is tested for three more sample times. If at least two of these three samples detect a logic low, a valid start bit is assumed to be detected. If in two or more samples, a logic high is detected, the line is assumed to be idle. The receive clock generator is controlled by the baud rate register (see Figure 13); however, the SCI is synchronized by the start bit independent of the transmitter. Once a valid start bit is detected, the start bit, each data bit, and the stop bit are each sampled three times. The value of the bit is determined by voting logic, which takes the value of a majority of samples. A noise flag is set when all three samples on a valid start bit, data bit, or stop bit do not agree. A noise flag is also set when the start verification samples do not agree.

START BIT DETECTION FOLLOWING A FRAMING ERROR

If there has been a framing error (FE) without detection of a break (10 zeros for 8-bit format or 11 zeros for a 9-bit format), the circuit continues to operate as if there actually were a stop bit, and the start edge will be placed artificially. The last bit received in the data shift register is inverted to a logic one, and the three logic-one start qualifiers are forced into the sample shift register during the interval when detection of a start bit is anticipated; therefore, the start bit will be accepted no sooner than it is anticipated.

If the receiver detects that a break (RDRF = 1, FE = 1, receiver data register = $00) produced the framing error, the start bit will not be artificially induced, and the receiver must actually receive a logic one before start.

SCI SYNCHRONOUS TRANSMISSION

The SCI transmitter allows a one-way synchronous transmission, with the SCLK pin as the clock output. No clock is sent to the SCLK pin during the stop and start bits. The LCL bit (SSCR1) controls whether clocks are active during the last valid data bit (address mark). The CPOL bit selects clock polarity, and the CPHA bit selects the phase of the external clock. During idle, preamble, and send break, the external SCLK clock is not active.

These options allow the SCI to control serial peripherals consisting of shift registers without losing any function of the SCI transmitter. These options do not affect the SCI receiver, which is independent of the transmitter.

The SCLK pin works in conjunction with the TDO pin. When the SCI transmitter is disabled, the SCLK and the TDO pins assume a high-impedance state.

NOTE

THE LBCL, CPOL and CPHA bits must be selected before the transmitter is enabled to ensure that the clocks function correctly. These bits should not be changed while the transmitter is enabled.

TRANSMIT DATA OUT

Transmit data out (TDO) is the serial data presented from the transmit data register (TDR) via the SCI to the output pin. The transmitter generates a bit time by using a derivative of the RT clock, producing a transmission rate equal to one-sixteenth that of the receiver sample clock (if the same baud rate is used for transmit and receive).

FUNCTIONAL DESCRIPTION

A block diagram of the SCI is shown in Figure 12. The user has option bits in the serial communications control register 1 (SCCR1) to determine the SCI wake-up method and data word length. Serial communications control register 2 (SCCR2) provides control bits that individually enable/disable the transmitter or receiver, enable system interrupts, and provide wake-up enable, and send break code bits. The baud rate register bits allow the user to select different baud rates for the transmitter and receiver.

Data transmission is initiated by a write to the serial communications data register (SCDAT). Provided the transmitter is enabled, data stored in the SCDAT is transferred to the transmit data shift register. This data transfer sets the SCI status register (SCSR) transmit data register empty (TDRE) bit and generates an interrupt if the transmit interrupt is enabled. Data transfer to the transmit data shift register is synchronized with the bit rate clock. All data is transmitted LSB first. Upon completion of data transmission, the transmission complete (TC) bit is set (provided no pending data, preamble, or break code is sent), and an interrupt is generated if the transmit complete interrupt is enabled. If the transmitter is disabled, and the data, preamble, or break code has been sent, the TC bit will also be set, which will also generate an interrupt if the TCIE bit is set. If the transmitter is disabled in the middle of a transmission, that character will be completed before the transmitter gives up control of the TDO pin.

When the SCDAT is read, it contains the last data byte received, provided that the receiver is enabled. The SCSR receive data register full (RDRF) bit is set to indicate that a data byte is transferred from the input serial shift register to the SCDAT, which can cause an interrupt if the receiver interrupt is enabled. Data transfer from the input serial shift register to the SCDAT is synchronized by the receiver bit rate clock. The SCSR overrun (OR), noise flag (NF), or FE bits are set if data reception errors occur.

An idle line interrupt is generated if the idle line interrupt is enabled and the SCSR IDLE bit (which detects idle line transmission) is set. This allows a receiver that is not in the wake-up mode to detect the end of a message, the preamble of a new message, or to resynchronize with the transmitter. A valid character must be received before the idle line condition for the IDLE bit to be set and for an idle line interrupt to be generated.

REGISTERS

There are five registers used in the SCI; the internal configuration of these registers is discussed in the following paragraphs.

Serial Communications Data Register (SCDAT) $11

The SCDAT is a read/write register used to receive and transmit SCI data.

	7	6	5	4	3	2	1	0
	SCD7	SCD6	SCD5	SCD4	SCD3	SCD2	SCD1	SCD0
RESET:	U	U	U	U	U	U	U	U

As shown in Figure 12, SCDAT functions as two separate registers. The transmit data register (TDR) provides the parallel interface from the internal data bus to the transmit shift register. The receive data register (RDR) provides the interface from the receive shift register to the internal data bus.

Serial Communications Control Register 1 (SCCR1) $OE

The SCCR1 provides control bits that determine word length, select the wake-up method, and control the options to output the transmitter clocks for synchronous transmissions.

	7	6	5	4	3	2	1	0
	R8	T8	—	M	WAKE	CPOL	CPHA	LBCL
RESET:	U	U	—	U	U	U	U	U

R8 — Receive Data Bit 8
R8 bit provides storage location for the ninth bit in the receive data byte (if M = 1).

T8 — Transmit Data Bit 8
T8 bit provides storage location for the ninth bit in the transmit data byte (if M = 1).

M — SCI Character Word Length
1 = one start bit, nine data bits, one stop bit
0 = one start bit, eight data bits, one stop bit

WAKE — Wake-Up Select
Wake bit selects the receiver wake-up method.
1 = Address bit (most significant bit)
0 = Idle line condition

CPOL - Clock Polarity
Selects the clock polarity sent to the SCLK pin.
1 = Steady state high outside the transmission window
0 = Steady state low outside the transmission window
The CPOL bit should not be changed with the transmitter active.

CPHA — Clock Phase
Selects the clock phase sent to the SCLK pin.
1 = SCLK line activated at the beginning of the data bit
0 = SCLK line activated in the middle of the data bit (see Figures 13 and 14)
The CPHA bit should not be changed with the transmitter active.

LBCL — Last Bit Clock
Selects whether the clock associated with the last data bit transmitted is output to the SCLK pin.
1 = Last data bit output
0 = Last data bit not output
The last data bit is the eighth or ninth bit, depending on whether an 8- or 9-bit format is used (see Table 5).
The LBCL bit should not be changed while the transmitter is enabled.

Bit 5 — Not used
Can read either one or zero

Table 5. SCI Clock on SCLK Pin

Data Format	M Bit	LBCL Bit	Number of Clocks on SCLK Pin
8 Bit	0	0	7
8 Bit	0	1	8
9 Bit	1	0	8
9 Bit	1	1	9

The address bit is dependent on both the wake-bit and the M-bit level. Additionally, the receiver does not use the wake-up feature unless the RWU control bit in SCCR2 is set.

Wake	M	Receiver Wake-Up
0	X	Detection of an idle line allows the next data byte received to cause the receive data register to fill and produce an RDRF flag.
1	0	Detection of a received one in the eighth data bit allows an RDRF flag and associated error flags.
1	1	Detection of a received one in the ninth data bit allows an RDRF flag and associated error flags.

Serial Communications Control Register 2 (SCCR2) $OF

The SCCR2 provides control of individual SCI functions such as interrupts, transmit/receive enabling, receiver wake-up, and break code.

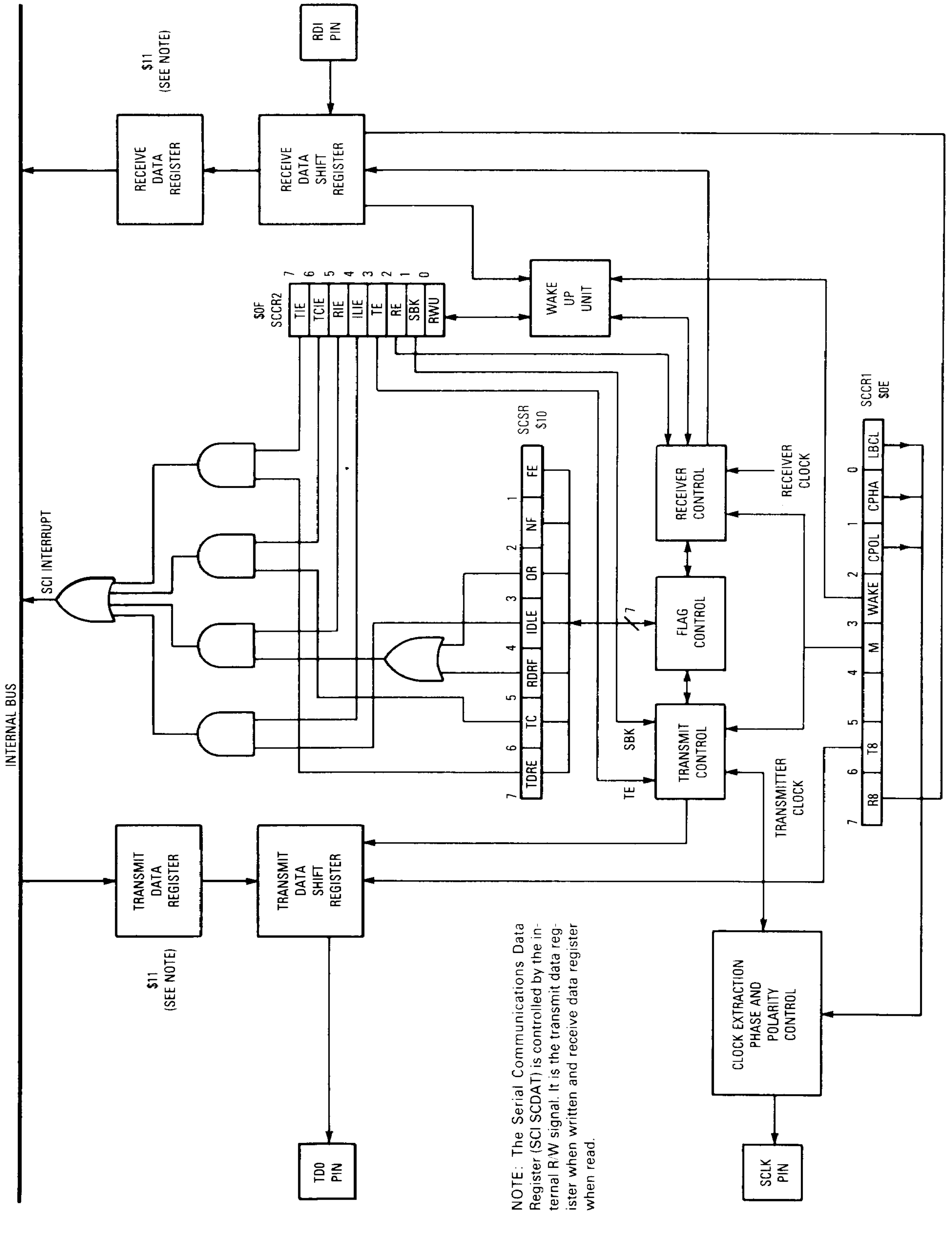
	7	6	5	4	3	2	1	0
	TIE	TCIE	RIE	ILIE	TE	RE	RWU	SBK
RESET:	0	0	0	0	0	0	0	0

TIE — Transmit Interrupt Enable
1 = SCI interrupt enabled, provided TDRE is set
0 = TDRE interrupt disabled

TCIE — Transmit Complete Interrupt Enable
1 = SCI interrupt enabled, provided TC is set
0 = TC interrupt disabled

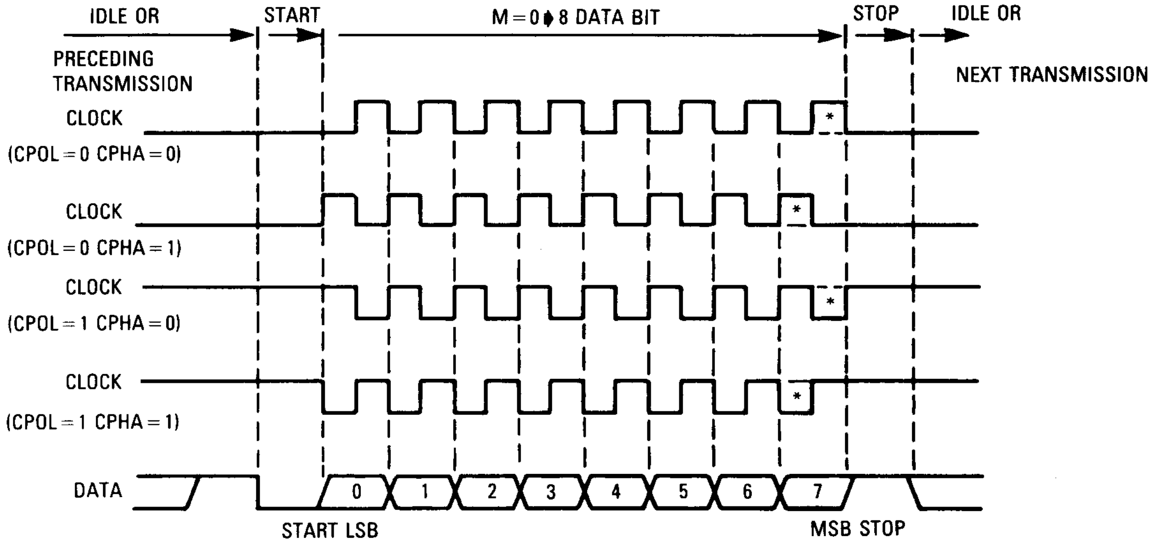
RIE — Receive Interrupt Enable
1 = SCI interrupt enabled, provided OR or RDRF is set
0 = RDRF and OR interrupts disabled

ILIE — Idle Line Interrupt Enable
1 = SCI interrupt enabled, provided IDLE is set
0 = Idle interrupt disabled



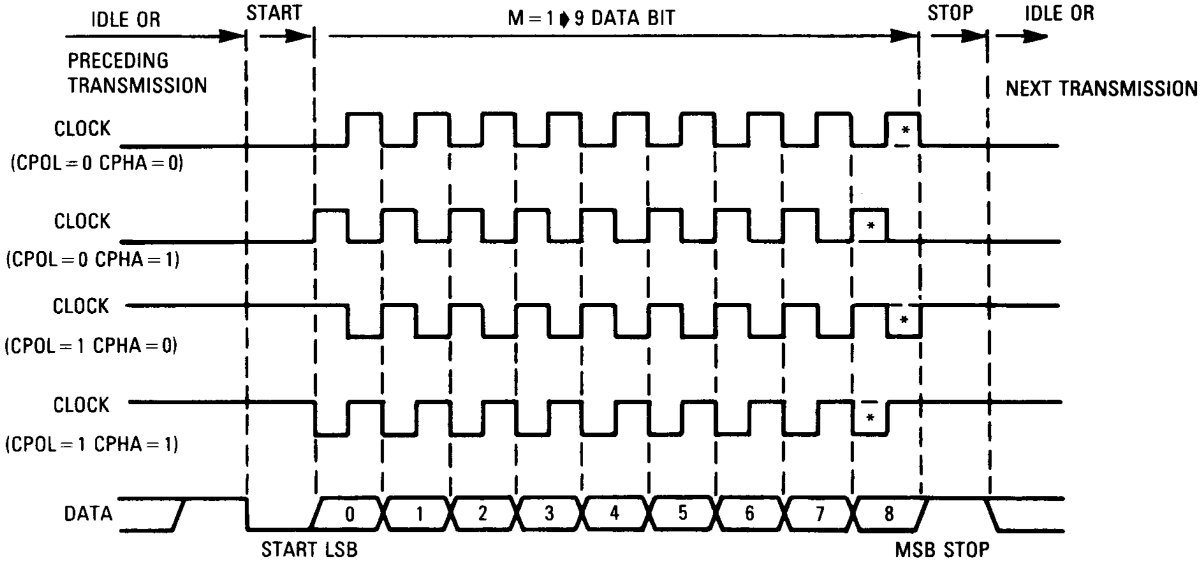
NOTE: The Serial Communications Data Register (SCI SCDAT) is controlled by the internal R/W signal. It is the transmit data register when written and receive data register when read.

Figure 12. SCI Block Diagram



*LBCL Bit Controls Last Data Clock

Figure 13. SCI Data Clock Timing Diagram (M=0)



*LBCL Bit Controls Last Data Clock

Figure 14. SCI Data CLock Timing Diagram (M=1)

TE — Transmit Enable

1 = Transmit shift register output is applied to the TD0 line, and the corresponding clocks are applied to the SCLK pin. Depending upon the SCCR1 M bit, a preamble of 10 (M=0) or 11 (M=1) consecutive ones is transmitted.

0 = Transmitter disabled after last byte is loaded in the SCDAT and TDRE is set. After last byte is transmitted, TD0 line becomes a high-impedance line.

RE — Receive Enable

1 = Receiver shift register input is applied to the RDI line.

0 = Receiver disabled and RDRF, IDLE, OR, NF, and FE status bits are inhibited.

RWU — Receiver Wake-Up

1 = Places receiver in sleep mode and enables wake-up function

0 = Wake-up function disabled after receiving data word with MSB set (if WAKE=1)

Wake-up function also disabled after receiving 10 (M = 0) or 11 (M = 1) consecutive ones (if WAKE = 0)

SBK — Send Break

1 = Transmitter continually sends blocks of zeros (sets of 10 or 11) until cleared. Upon completion of break code, transmitter sends one high bit for recognition of valid start bit.

0 = Transmitter sends 10 (M = 0) or 11 (M = 1) zeros then reverts to an idle state or continues sending data. If transmitter is empty and idle, setting and clearing the SBK bit may queue up to two character times of break because the first break transfers immediately to the shift register, and the second is queued into the parallel transmit buffer.

Serial Communications Status Register (SCSR) $10

The SCSR provides inputs to the SCI interrupt logic circuits. Noise flag and framing error bits are also contained in the SCSR.

	7	6	5	4	3	2	1	0
	TDRE	TC	RDRF	IDLE	OR	NF	FE	—
RESET:	1	1	0	0	0	0	0	—

TDRE — Transmit Data Register (TDR) Empty

1 = TDR contents transferred to the transmit data shift register

0 = TDR still contains data. TDRE is cleared by reading the SCSR, followed by a write to the TDR.

TC — Transmit Complete

1 = Indicates end of data frame, preamble, or break condition has occurred if:

1. TE = 1, TDRE = 1, and no pending data, preamble or break is to be transmitted; or
2. TE = 0 and the data preamble or break (in the transmit shift register) has been transmitted.

0 = TC bit cleared by reading the SCSR, followed by a write to the TDR

The TC bit is a status register that indicates one of the above conditions has occurred. It does not inhibit the transmitter in any way.

RDRF — Receive Data Register (RDR) Full

1 = Receive data shift register contents transferred to the RDR

0 = Receive data shift register transfer did not occur. RDRF is cleared by reading the SCSR, followed by a read of the RDR

IDLE — Idle Line Detect

1 = Indicates receiver has detected an idle line

0 = IDLE is cleared by reading the SCSR, followed by a read of the RDR. Once IDLE is cleared, IDLE cannot be set until RDI line becomes active and idle again.

OR — Overrun Error

1 = Indicates receive data shift register data is ready to be sent to a full RDR (RDRF = 1). Data causing the overrun is lost, and RDR data is not disturbed.

0 = OR is cleared by reading the SCSR, followed by a read of the RDR.

NF — Noise Flag

1 = Indicates noise is present on the receive bits, including the start and stop bits. NF is not set until RDRF = 1.

0 = NF is cleared by reading the SCSR, followed by a read of the RDR.

FE — Framing Error

1 = Indicates stop bit not detected in received data character. FE is set the same time RDRF is set. If received byte causes both framing and overrun errors, processor will only recognize the overrun error. Further data transfer into the RDR is inhibited until FE is cleared.

0 = FE is cleared by reading the SCSR, followed by a read of the RDR.

Bit 0 — Not used

Can read either one or zero

Baud Rate Register $0D

The baud rate register selects the SCI transmitter and receiver baud rate. The SCP1 and SCP0 prescaler bits are used in conjunction with the SCR2–SCR0 bits to generate the receiver baud rate and in conjunction with the SCT2–SCT0 baud rate bits to generate the transmitter baud rate.

Tables 6 and 7 tabulate the divide chain used to obtain the baud rate clock (transmit or receive clock). The actual divider chain is controlled by the combined SCP1–SCP0 and SCR2–SCR0 or SCT2–SCT0 bits in the baud rate register. The divided frequencies shown in Table 6 represent the final baud rate that results from prescaler division only (SCR or SCT bits all zero). Table 7 lists the prescaler output frequency divided by the action of the SCR or SCT bits.

For example, assume that 9600-Hz baud rate is desired from a 2.4576-MHz system clock crystal. The prescaler bits could be set for either a divide-by-one or divide-by-four. If a divide-by-four prescaler is used, then the SCR and SCT bits must be set for divide-by-two. The same result, using the same crystal frequency, can be obtained with a prescaler divide-by-one and SCR and SCT bit divide-by-eight.

	7	6	5	4	3	2	1	0
	SCP1	SCP0	SCT2	SCT1	SCT0	SCR2	SCR1	SCR0
RESET:	0	0	U	U	U	U	U	U

SCP1–SCP0 — SCI Prescaler Bits 1 and 0

These two prescaler bits are used to increase the range of standard baud rates controlled by the SCT2-SCT0 and SCR2–SCR0 bits. Prescaler internal processor clock division versus bit levels are shown in Table 6.

SCT2-SCT0 — SCI Transmit Baud Rate Selection Bits

These three bits, taken in conjunction with bits SCP1–SCP0, are used to select the SCI transmit baud rate. Baud rates versus bit levels are listed in Table 7.

Table 6. Prescaler Highest Baud Rate Frequency Output

SCP Bit		Clock* Divided By	Crystal Frequency MHz				
1	0		4.194304	4.0	2.4576	2.0	1.8432
0	0	1	131.072 kHz	125.000 kHz	76.80 kHz	62.50 kHz	57.60 kHz
0	1	3	43.691 kHz	41.666 kHz	25.60 kHz	20.833 kHz	19.20 kHz
1	0	4	32.768 kHz	31.250 kHz	19.20 kHz	15.625 kHz	14.40 kHz
1	1	13	10.082 kHz	9600 Hz	5.907 kHz	4800 Hz	4430 Hz

*Refers to the internal processor clock.

NOTE: The divided frequencies shown in Table 6 represent baud rates which are the highest transmit baud rate (Tx) that can be obtained by a specific crystal frequency and only using the prescaler division. Lower baud rates may be obtained by providing a further division using the SCI rate select bits as shown below for some representative prescaler outputs.

Table 7. Transmit Baud Rate Output for a Given Prescaler Output

SCR/T Bits			Divided By	Representative Highest Prescaler Baud Rate Output				
2	1	0		131.072 kHz	32.768 kHz	76.80 kHz	19.20 kHz	9600 Hz
0	0	0	1	131.072 kHz	32.768 kHz	76.80 kHz	19.20 kHz	9600 Hz
0	0	1	2	65.536 kHz	16.384 kHz	38.40 kHz	9600 Hz	4800 Hz
0	1	0	4	32.768 kHz	8.192 kHz	19.20 kHz	4800 Hz	2400 Hz
0	1	1	8	16.384 kHz	4.096 kHz	9600 Hz	2400 Hz	1200 Hz
1	0	0	16	8.192 kHz	2.048 kHz	4800 Hz	1200 Hz	600 Hz
1	0	1	32	4.096 kHz	1.024 kHz	2400 Hz	600 Hz	300 Hz
1	1	0	64	2.048 kHz	512 Hz	1200 Hz	300 Hz	150 Hz
1	1	1	128	1.024 kHz	256 Hz	600 Hz	150 Hz	75 Hz

NOTE: Table 7 illustrates how the SCI select bits can be used to provide lower transmitter or receiver baud rates by further dividing the prescaler output frequency. The five examples are only representative samples. In all cases, the baud rates shown are transmit baud rates (transmit clock), and the receive clock is 16 times higher in frequency than the actual baud rate.

SCR2-SCR0 — SCI Receive Baud Rate Selection Bits

These three bits, taken in conjunction with bits SCP1–SCP0, are used to select the SCI receive baud rate. Baud rates versus bit levels are listed in Table 7.

Load Program in RAM and Execute

This function is entered if the following conditions are met when reset is released:

IRQ is at V_{DD} + 4 V for at least two machine cycles after reset

TCAP1 is at V_{DD} for at least two machine cycles after reset

PD3 is at V_{DD} for at least 30 machine cycles after reset

PD4 is at V_{SS} for at least 30 machine cycles after reset

User programs are loaded into RAM using the SCI port and then executed. Data is loaded sequentially, starting at RAM location $50, until the last byte is loaded. Program control is then transferred to the RAM progam starting at location $51. The first byte loaded is the count of the number of bytes in the program plus the count byte. The program starts at the second byte in the RAM. During firmware initialization, the SCI is configured for the NRZ format (idle line, eight data bits, and stop bit). The baud rate is 9600 with a 4-MHz crystal. Figure 15 shows a schematic for the load program in RAM and execute function.

Immediate execution can be avoided by setting the byte count to a value greater than the length of data loaded, which causes the firmware to wait for additional data after loading is complete. Resetting the MCU then allows entering any routine without disturbing the RAM data that was loaded.

Jump to Any Address

This function is entered if the following conditions are met when reset is released:

IRQ is at V_{DD} + 4 V for at least two machine cycles after reset

TCAP1 is at V_{DD} for at least two machine cycles after reset

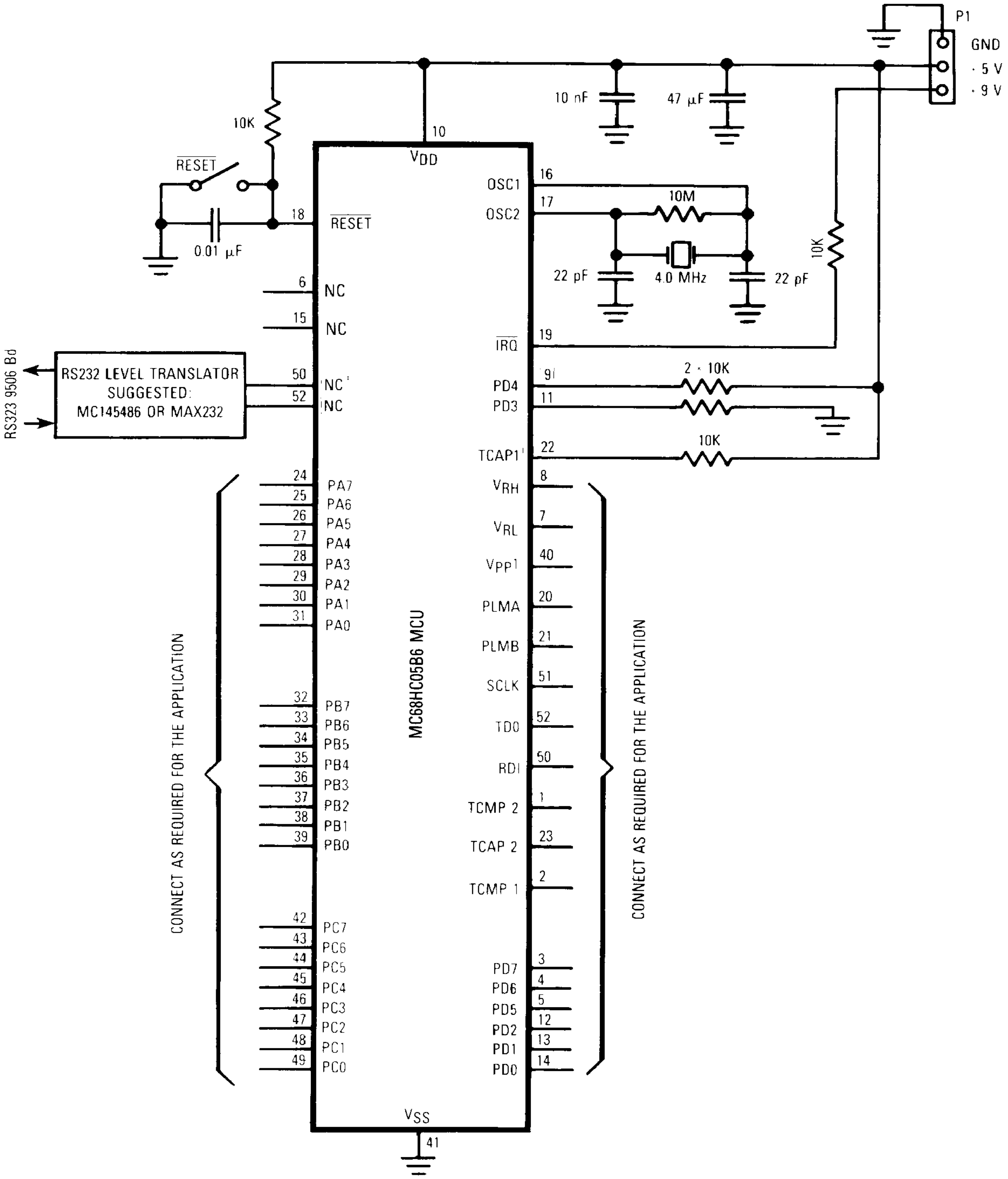
PD3 is at V_{DD} for at least 30 machine cycles after reset

PD4 is at V_{DD} for at least 30 machine cycles after reset

To execute the jump to any address function, port A data input should be $CC, and port B and C should be the MSB and LSB, respectively, of the address desired for the jump. Figure 16 shows a schematic for the jump function.

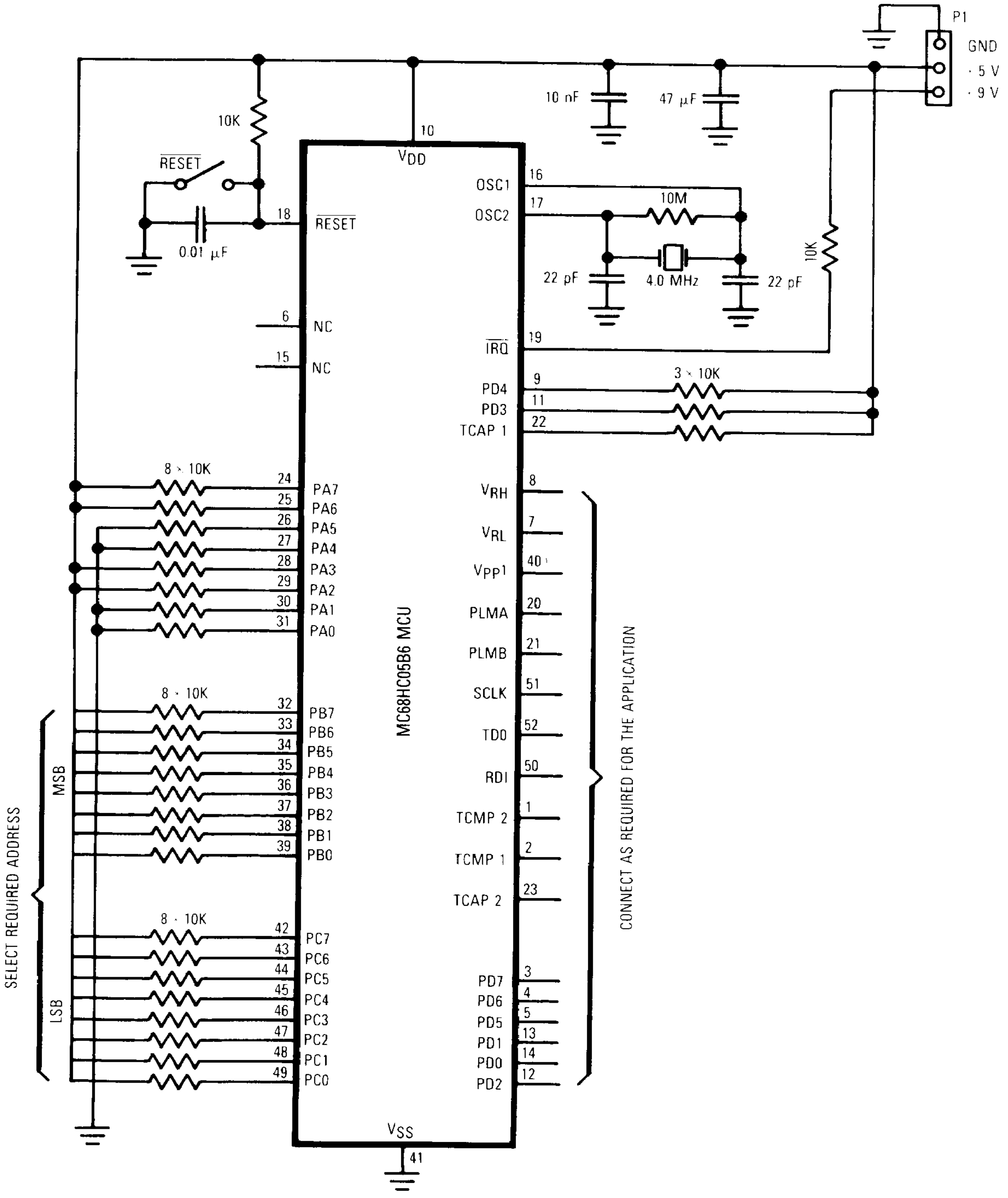
PULSE-LENGTH D/A CONVERTERS

The pulse-length D/A converter (PLM) works in conjunction with the timer to execute two 8-bit conversions



NOTE: Pin numbers are valid for the 52-pin PLCC packge only.

Figure 15. Load Program in RAM and Execute Diagram



NOTE: Pin numbers are valid for 52-pin PLCC package only.

Figure 16. Jump to Any Address Diagram

with a choice of two repetition rates. The outputs are pulse-length modulated signals whose duty-cycle ratio may be modified. These signals can be used directly as PLM, or the filtered average values can be used as general-purpose analog outputs.

Registers PLMA and PLMB contain the pulse-length values for the two PLMs. A value of $00 results in a continuously low output from the D/A. A value of $80 results in a 50-percent duty-cycle output, and a value of $FF gives an output that is a logic 1 for 255/256 of the cycle. When the MCU writes to the PLMA or PLMB register, the D/A picks up the new value at the end of a complete conversion cycle. A monotonic change in the dc component of the output results, without overshoots or vicious starts (a vicious start is an output that gives totally erroneous output during the first cycle following an update of the registers). WAIT mode does not affect the output waveform of the D/A converters.

NOTE

Since the PLM system uses the timer counter, PLM results will be affected while resetting the timer counter.

Figure 17 shows a block diagram of the PLM system.

PLMA (0A)

7	6	5	4	3	2	1	0
PLMA7	PLMA6	PLMA5	PLMA4	PLMA3	PLMA2	PLMA1	PLMA0
RESET:							
0	0	0	0	0	0	0	0

PLMB (0B)

7	6	5	4	3	2	1	0
PLMB7	PLMB6	PLMB5	PLMB4	PLMB3	PLMB2	PLMB1	PLMB0
RESET:							
0	0	0	0	0	0	0	0

Miscellaneous (0C)

7	6	5	4	3	2	1	0
—	—	—	—	SFA	SFB	—	—
RESET:							
—	—	—	—	0	0	—	—

SFA — Slow/Fast Control for PLMA Clock

1 = Slow speed of PLMA used (4096 times the timer clock period)

0 = Fast speed of PLMA used (256 times the timer clock period)

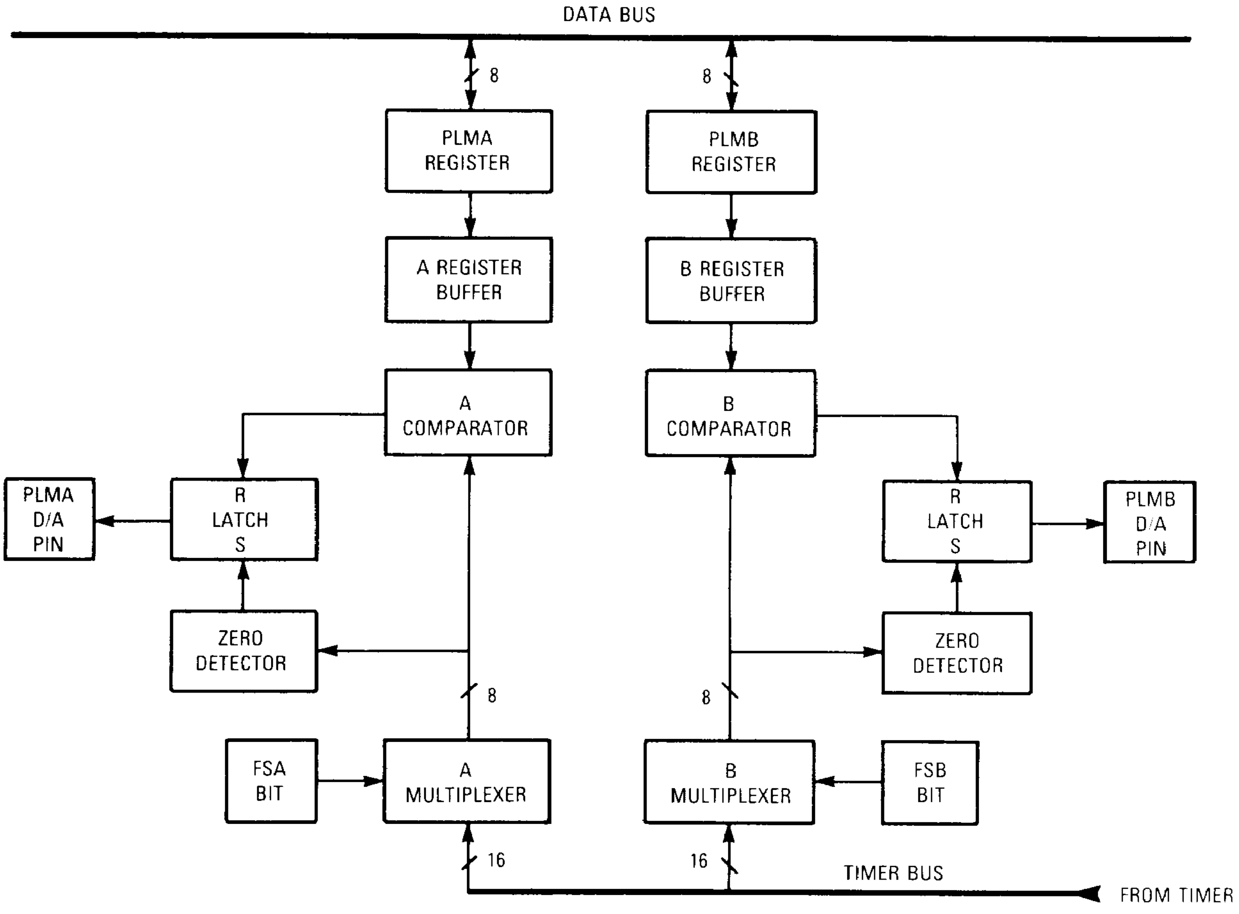


Figure 17. PLM Block Diagram

SFB — Slow/Fast Control for PLMB Clock
1 = Slow speed of PLMB used (4096 times the timer clock period)
0 = Fast speed of PLMB used (256 times the timer clock period)

NOTE

The highest speed of the PLM system corresponds to the frequency of the TOF bit being set, multiplied by 256. The slowest speed of the PLM system corresponds to the frequency of the TOF bit being set, multipled by 16.

The SFA and SFB bits are not double buffered; therefore, these bits must be selected before writing to either PLM register to avoid temporary wrong values from the PLM outputs. Figure 18 shows some examples of the PLM output waveforms.

A/D CONVERTER

The A/D converter system consists of an 8-bit successive approximation converter and a 16-channel multiplexer. Eight of the channels are available for output, and the other eight channels are dedicated to internal test functions. There is one 8-bit result data register (address $08) and one 8-bit status/control register (address $09).

NOTE

In the 48-pin dual-in-line package, the fixed input port (D) of the MC68HC05B6 is reduced to six pins (PD5–PD0, AN5–AN0). This change has no effect on either programming or operation of port D or the A/D converter.

The reference supply for the converter uses dedicated input pins instead of the power supply lines, because drops caused by loading in the power supply lines would degrade the accuracy of the A/D conversion. An internal RC oscillator is available if the bus speed is low enough to degrade the A/D accuracy. An ADON bit allows the A/D to be switched off to reduce power consumption, which is particularly useful in the WAIT mode.

For ratiometric conversions, the source of each analog input should use V_{RH} as the supply voltage and be referenced to V_{RL}. An input voltage greater than or equal to V_{RH} converts as $FF (full scale) with no overflow indication. An input voltage equal to V_{RL} converts as $00. The conversion is monotonic with no missing codes.

A/D STATUS/CONTROL REGISTER ($09)

7	6	5	4	3	2	1	0
COCO	ADRC	ADON	0	CH3	CH2	CH1	CH0
RESET:							
0	0	0	0	0	0	0	0

COCO — Conversion Complete
1 = Conversion is complete; a new result can be read from the result data register ($08).
0 = No conversion since last reset

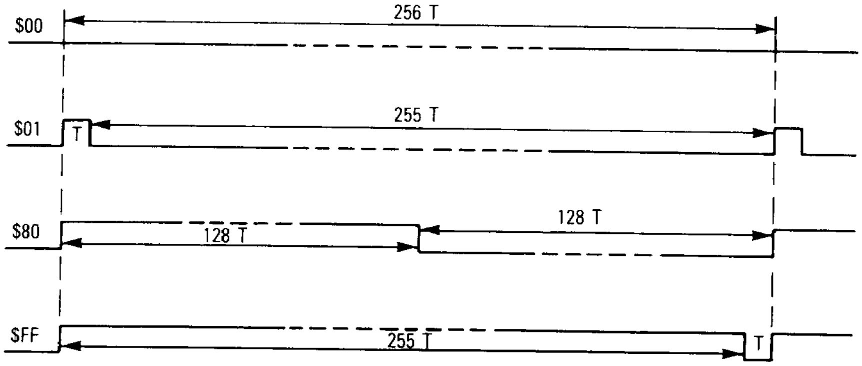
ADRC — A/D RC Oscillator Control
1 = A/D uses RC clock
0 = A/D uses CPU clock
When the RC oscillator is turned on, it requires a time t_{adrc} to stabilize, and results can be inaccurate during this time.

ADON — A/D On
1 = A/D enabled
0 = A/D disabled
When the A/D is turned on, it requires a time t_{ADON} for the current sources to stabilize, and results can be inaccurate during this time.

CH3–CH0 — Channel 3 through Channel 0
These bits select the A/D channel assignment (see Table 8).

NOTE

Using one or more pins of PD7/AN7–PD0/AN0 as analog inputs does not affect the ability to use port D inputs as digital inputs. However, using port D



T = 4 CPU Clocks Fast Mode
T = 64 CPU Clocks Slow Mode

Figure 18. PLM Output Waveform Examples

for digital inputs during an analog conversion sequence may inject noise on the analog inputs and reduce the accuracy of the A/D result.

Performing a digital read of port D with levels other than V_{DD} or V_{SS} on the inputs causes greater than normal power dissipation during the read and may give erroneous results.

Table 8. A/D Channel Assignments

CH3	CH2	CH1	CH0	Channel Selected
0	0	0	0	AN0, Port D Bit 0
0	0	0	1	AN1, Port D Bit 1
0	0	1	0	AN2, Port D Bit 2
0	0	1	1	AN3, Port D Bit 3
0	1	0	0	AN4, Port D Bit 4
0	1	0	1	AN5, Port D Bit 5
0	1	1	0	AN6, Port D Bit 6
0	1	1	1	AN7, Port D Bit 7
1	0	0	0	V_{RH} Pin (High)
1	0	0	1	$((V_{RH})+(V_{RL}))/2$
1	0	1	0	V_{RL} Pin (Low)
1	0	1	1	V_{RL} Pin (Low)
1	1	0	0	V_{RL} Pin (Low)
1	1	0	1	V_{RL} Pin (Low)
1	1	1	0	V_{RL} Pin (Low)
1	1	1	1	V_{RL} Pin (Low)

INSTRUCTION SET

The MCU instructions can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type.

This MCU uses all the instructions available in the M146805 CMOS Family plus one more: the unsigned multiply (MUL) instruction. This instruction allows unsigned multiplication of the contents of the accumulator (A) and the index register (X). The high-order product is then stored in the index register, and the low-order product is stored in the accumulator. A detailed definition of the MUL instruction is shown below.

Operation	X:A ← X · A		
Description	Multiplies the eight bits in the index register by the eight bits in the accumulator to obtain a 16-bit unsigned number in the concatenated accumulator and index register		
Condition Codes	H: Cleared I: Not affected N: Not affected Z: Not affected C: Cleared		
Source Form(s)	MUL		
Addressing Mode	Cycles	Bytes	Opcode
Inherent	11	1	$42

REGISTER/MEMORY INSTRUCTIONS

Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to the following instruction list.

Function	Mnemonic
Load A from Memory	LDA
Load X from Memory	LDX
Store A in Memory	STA
Store X in Memory	STX
Add Memory to A	ADD
Add Memory and Carry to A	ADC
Subtract Memory	SUB
Subtract Memory from A with Borrow	SBC
AND Memory to A	AND
OR Memory with A	ORA
Exclusive OR Memory with A	EOR
Arithmetic Compare A with Memory	CMP
Arithmetic Compare X with Memory	CPX
Bit Test Memory with A (Logical Compare)	BIT
Jump Unconditional	JMP
Jump to Subroutine	JSR

BRANCH INSTRUCTIONS

This set of instructions branches if a particular condition is met; otherwise, no operation is performed. Branch instructions are two-byte instructions. Refer to the following list for branch instructions.

Function	Mnemonic
Branch Always	BRA
Branch Never	BRN
Branch if Higher	BHI
Branch if Lower or Same	BLS
Branch if Carry Clear	BCC
Branch if Higher or Same	BHS
Branch if Carry Set	BCS
Branch if Lower	BLO
Branch if Not Equal	BNE
Branch if Equal	BEQ
Branch if Half Carry Clear	BHCC
Branch if Half Carry Set	BHCS
Branch if Plus	BPL
Branch if Minus	BMI
Branch if Interrupt Mask Bit is Clear	BMC
Branch if Interrupt Mask Bit is Set	BMS
Branch if Interrupt Line is Low	BIL
Branch if Interrupt Line is High	BIH
Branch to Subroutine	BSR

READ-MODIFY-WRITE INSTRUCTIONS

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is an exception to the read-modify-write sequence since it does not modify the value. Refer to the following list of instructions.

Function	Mnemonic
Increment	INC
Decrement	DEC
Clear	CLR
Complement	COM
Negate (Twos Complement)	NEG
Rotate Left Thru Carry	ROL
Rotate Right Thru Carry	ROR
Logical Shift Left	LSL
Logical Shift Right	LSR
Arithmetic Shift Right	ASR
Test for Negative or Zero	TST
Multiply	MUL

CONTROL INSTRUCTIONS

These instructions are register reference instructions and are used to control processor operation during program execution. Refer to the following list for control instructions.

Function	Mnemonic
Transfer A to X	TAX
Transfer X to A	TXA
Set Carry Bit	SEC
Clear Carry Bit	CLC
Set Interrupt Mask Bit	SEI
Clear Interrupt Mask Bit	CLI
Software Interrupt	SWI
Return from Subroutine	RTS
Return from Interrupt	RTI
Reset Stack Pointer	RSP
No-Operation	NOP
Stop	STOP
Wait	WAIT

BIT MANIPULATION INSTRUCTIONS

The MCU is capable of setting or clearing any writable bit which resides in the first 256 bytes of the memory space where all port registers, port DDRs, timer, timer control, ROM, and on-chip RAM reside. An additional feature allows the software to test and branch on the state of any bit within these 256 locations. The bit set, bit clear and bit test, and branch functions are all implemented with a single instruction. For test and branch instructions, the value of the bit tested is also placed in the carry bit of the condition code register. Refer to the following list for bit manipulation instructions.

Function	Mnemonic
Branch if Bit n is Set	BRSET n (n = 0 . . . 7)
Branch if Bit n is Clear	BRCLR n (n = 0 . . . 7)
Set Bit n	BSET n (n = 0 . . . 7)
Clear Bit n	BCLR n (n = 0 . . . 7)

OPCODE MAP SUMMARY

Table 9 is an opcode map for the instructions used on the MCU.

ADDRESSING MODES

The MCU uses ten different addressing modes to provide the programmer with an opportunity to optimize the code for all situations. The various indexed addressing modes make it possible to locate data tables, code conversion tables, and scaling tables anywhere in the memory space. Short indexed accesses are single byte instructions; the longest instructions (three bytes) permit accessing tables throughout memory. Short and long absolute addressing is also included. One- or two-byte direct addressing instructions access all data bytes in most applications. Extended addressing permits jump instructions to reach all memory.

The term "effective address" (EA) is used in describing the various addressing modes. Effective address is defined as the address from which the argument for an instruction is fetched or stored.

IMMEDIATE

In the immediate addressing mode, the operand is contained in the byte immediately following the opcode. The immediate addressing mode is used to access constants that do not change during program execution (e.g., a constant used to initialize a loop counter).

DIRECT

In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single two-byte instruction.

EXTENDED

In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode byte. Instructions with extended addressing mode are capable of referencing arguments anywhere in memory with a single three-byte instruction. When using the Motorola assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the shortest form of the instruction.

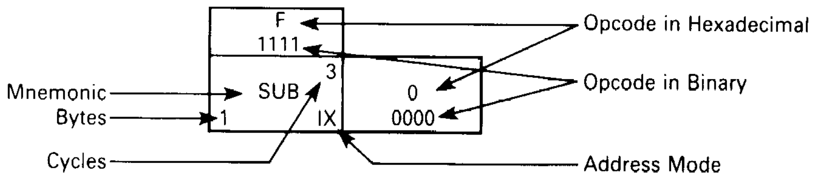
Table 9. Opcode Map

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory						
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX	
HI / LOW	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111	HI / LOW
0 0000	BRSET0 5 3 BTB	BSET0 5 2 BSC	BRA 3 2 REL	NEG 5 2 DIR	NEGA 3 1 INH	NEGX 3 1 INH	NEG 6 2 IX1	NEG 5 1 IX	RTI 9 1 INH		SUB 2 2 IMM	SUB 3 2 DIR	SUB 4 3 EXT	SUB 5 3 IX2	SUB 4 2 IX1	SUB 3 1 IX	0 0000
1 0001	BRCLR0 5 3 BTB	BCLR0 5 2 BSC	BRN 3 2 REL						RTS 6 1 INH		CMP 2 2 IMM	CMP 3 2 DIR	CMP 4 3 EXT	CMP 5 3 IX2	CMP 4 2 IX1	CMP 3 1 IX	1 0001
2 0010	BRSET1 5 3 BTB	BSET1 5 2 BSC	BHI 3 2 REL		MUL 11 1 INH						SBC 2 2 IMM	SBC 3 2 DIR	SBC 4 3 EXT	SBC 5 3 IX2	SBC 4 2 IX1	SBC 3 1 IX	2 0010
3 0011	BRCLR1 5 3 BTB	BCLR1 5 2 BSC	BLS 3 2 REL	COM 5 2 DIR	COMA 3 1 INH	COMX 3 1 INH	COM 6 2 IX1	COM 5 1 IX	SWI 10 1 INH		CPX 2 2 IMM	CPX 3 2 DIR	CPX 4 3 EXT	CPX 5 3 IX2	CPX 4 2 IX1	CPX 3 1 IX	3 0011
4 0100	BRSET2 5 3 BTB	BSET2 5 2 BSC	BCC 3 2 REL	LSR 5 2 DIR	LSRA 3 1 INH	LSRX 3 1 INH	LSR 6 2 IX1	LSR 5 1 IX			AND 2 2 IMM	AND 3 2 DIR	AND 4 3 EXT	AND 5 3 IX2	AND 4 2 IX1	AND 3 1 IX	4 0100
5 0101	BRCLR2 5 3 BTB	BCLR2 5 2 BSC	BCS 3 2 REL								BIT 2 2 IMM	BIT 3 2 DIR	BIT 4 3 EXT	BIT 5 3 IX2	BIT 4 2 IX1	BIT 3 1 IX	5 0101
6 0110	BRSET3 5 3 BTB	BSET3 5 2 BSC	BNE 3 2 REL	ROR 5 2 DIR	RORA 3 1 INH	RORX 3 1 INH	ROR 6 2 IX1	ROR 5 1 IX			LDA 2 2 IMM	LDA 3 2 DIR	LDA 4 3 EXT	LDA 5 3 IX2	LDA 4 2 IX1	LDA 3 1 IX	6 0110
7 0111	BRCLR3 5 3 BTB	BCLR3 5 2 BSC	BEQ 3 2 REL	ASR 5 2 DIR	ASRA 3 1 INH	ASRX 3 1 INH	ASR 6 2 IX1	ASR 5 1 IX		TAX 2 1 INH		STA 4 2 DIR	STA 5 3 EXT	STA 6 3 IX2	STA 5 2 IX1	STA 4 1 IX	7 0111
8 1000	BRSET4 5 3 BTB	BSET4 5 2 BSC	BHCC 3 2 REL	LSL 5 2 DIR	LSLA 3 1 INH	LSLX 3 1 INH	LSL 6 2 IX1	LSL 5 1 IX		CLC 2 1 INH	EOR 2 2 IMM	EOR 3 2 DIR	EOR 4 3 EXT	EOR 5 3 IX2	EOR 4 2 IX1	EOR 3 1 IX	8 1000
9 1001	BRCLR4 5 3 BTB	BCLR4 5 2 BSC	BHCS 3 2 REL	ROL 5 2 DIR	ROLA 3 1 INH	ROLX 3 1 INH	ROL 6 2 IX1	ROL 5 1 IX		SEC 2 1 INH	ADC 2 2 IMM	ADC 3 2 DIR	ADC 4 3 EXT	ADC 5 3 IX2	ADC 4 2 IX1	ADC 3 1 IX	9 1001
A 1010	BRSET5 5 3 BTB	BSET5 5 2 BSC	BPL 3 2 REL	DEC 5 2 DIR	DECA 3 1 INH	DECX 3 1 INH	DEC 6 2 IX1	DEC 5 1 IX		CLI 2 1 INH	ORA 2 2 IMM	ORA 3 2 DIR	ORA 4 3 EXT	ORA 5 3 IX2	ORA 4 2 IX1	ORA 3 1 IX	A 1010
B 1011	BRCLR5 5 3 BTB	BCLR5 5 2 BSC	BMI 3 2 REL							SEI 2 1 INH	ADD 2 2 IMM	ADD 3 2 DIR	ADD 4 3 EXT	ADD 5 3 IX2	ADD 4 2 IX1	ADD 3 1 IX	B 1011
C 1100	BRSET6 5 3 BTB	BSET6 5 2 BSC	BMC 3 2 REL	INC 5 2 DIR	INCA 3 1 INH	INCX 3 1 INH	INC 6 2 IX1	INC 5 1 IX		RSP 2 1 INH		JMP 2 2 DIR	JMP 3 3 EXT	JMP 4 3 IX2	JMP 3 2 IX1	JMP 2 1 IX	C 1100
D 1101	BRCLR6 5 3 BTB	BCLR6 5 2 BSC	BMS 3 2 REL	TST 4 2 DIR	TSTA 3 1 INH	TSTX 3 1 INH	TST 5 2 IX1	TST 4 1 IX		NOP 2 1 INH	BSR 6 2 REL	JSR 5 2 DIR	JSR 6 3 EXT	JSR 7 3 IX2	JSR 6 2 IX1	JSR 5 1 IX	D 1101
E 1110	BRSET7 5 3 BTB	BSET7 5 2 BSC	BIL 3 2 REL						STOP 2 1 INH		LDX 2 2 IMM	LDX 3 2 DIR	LDX 4 3 EXT	LDX 5 3 IX2	LDX 4 2 IX1	LDX 3 1 IX	E 1110
F 1111	BRCLR7 5 3 BTB	BCLR7 5 2 BSC	BIH 3 2 REL	CLR 5 2 DIR	CLRA 3 1 INH	CLRX 3 1 INH	CLR 6 2 IX1	CLR 5 1 IX	WAIT 2 1 INH	TXA 2 1 INH		STX 4 2 DIR	STX 5 3 EXT	STX 6 3 IX2	STX 5 2 IX1	STX 4 1 IX	F 1111

Abbreviations for Address Modes

INH	Inherent	REL	Relative
A	Accumulator	BSC	Bit Set/Clear
X	Index Register	BTB	Bit Test and Branch
IMM	Immediate	IX	Indexed (No Offset)
DIR	Direct	IX1	Indexed, 1 Byte (8-Bit) Offset
EXT	Extended	IX2	Indexed, 2 Byte (16-Bit) Offset

LEGEND



RELATIVE

The relative addressing mode is only used in branch instructions. In relative addressing, the contents of the 8-bit signed byte (the offset) following the opcode is added to the PC if, and only if, the branch conditions are true. Otherwise, control proceeds to the next instruction. The span of relative addressing is from − 126 to + 129 from the opcode address. The programmer need not calculate the offset when using the Motorola assembler, since it calculates the proper offset and checks to see that it is within the span of the branch.

INDEXED, NO OFFSET

In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. This addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is often used to move a pointer through a table or to hold the address of a frequently referenced RAM or I/O location.

INDEXED, 8-BIT OFFSET

In the indexed, 8-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the unsigned byte following the opcode. The addressing mode is useful for selecting the Kth element in an n element table. With this two-byte instruction, K would typically be in X with the address of the beginning of the table in the instruction. As such, tables may begin anywhere within the first 256 addressable locations and could extend as far as location 510 ($1FE is the last location at which the instruction may begin).

INDEXED, 16-BIT OFFSET

In the indexed, 16-bit offset addressing mode, the effective address is the sum of the contents of the unsigned 8-bit index register and the two unsigned bytes following the opcode. This address mode can be used in a manner similar to indexed, 8-bit offset except that this three-byte instruction allows tables to be anywhere in memory. As with direct and extended addressing, the Motorola assembler determines the shortest form of indexed addressing.

BIT SET/CLEAR

In the bit set/clear addressing mode, the bit to be set or cleared is part of the opcode, and the byte following the opcode specifies the direct addressing of the byte in which the specified bit is to be set or cleared. Any read/write bit in the first 256 locations of memory, including I/O, can be selectively set or cleared with a single two-byte instruction.

BIT TEST AND BRANCH

The bit test and branch addressing mode is a combination of direct addressing and relative addressing. The bit that is to be tested and its condition (set or clear), is included in the opcode. The address of the byte to be tested is in the single byte immediately following the opcode byte. The signed relative 8-bit offset in the third byte is added to the PC if the specified bit is set or cleared in the specified memory location. This single three-byte instruction allows the program to branch based on the condition of any readable bit in the first 256 locations of memory. The span of branching is from − 125 to + 130 from the opcode address. The state of the tested bit is also transferred to the carry bit of the condition code register.

INHERENT

In the inherent addressing mode, all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator as well as the control instruction with no other arguments are included in this mode. These instructions are one byte long.

ELECTRICAL SPECIFICATIONS

MAXIMUM RATINGS (Voltages referenced to V_{SS})

Rating	Symbol	Value	Unit
Supply Voltage	V_{DD}	− 0.5 to + 7.0	V
Input Voltage	V_{in}	V_{SS} − 0.5 to V_{DD} + 0.5	V
Self-Check Mode ($\overline{\text{IRQ}}$ Pin Only)	V_{in}	V_{SS} − 0.5 to 2 × V_{DD} + 0.5	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	25	mA
Operating Temperature Range MC68HC05B6P, FN (Standard) MC68HC05B6CP, CFN (Extended) MC68HC05B6MP, MFN (Automotive)	T_A	T_L to T_H 0 to + 70 − 40 to + 85 − 40 to + 125	°C
Storage Temperature Range	T_{stg}	− 65 to + 150	°C

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum-rated voltages to this high-impedance circuit. For proper operation, it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \leq (V_{in}$ or $V_{out}) \leq V_{DD}$. Reliability of operation is enhanced if unused inputs are connected to an appropriate logic voltage level (e.g., either V_{SS} or V_{DD}).

THERMAL CHARACTERISTICS

Characteristic	Symbol	Value	Unit
Thermal Resistance Plastic Plastic Leaded Chip Carrier (PLCC)	θ_{JA}	 40 50	°C/W

V_{DD} = 4.5 V

Pins	R1	R2	C
PA7–PA0, PB7–PB0, PC7–PC0, TCMP1 TCMP2	3.26 kΩ	2.38 kΩ	50 pF
TDO, SCLK, PLMA, PLMB	1.9 kΩ	2.26 kΩ	200 pF

V_{DD} = 3.0 V

Pins	R1	R2	C
PA7–PA0, PB7–PB0, PC7–PC0, TCMP1, TCMP2	10.91 kΩ	6.32 kΩ	50 pF
TDO, SCLK, PLMA, PLMB	6 kΩ	6 kΩ	200 pF

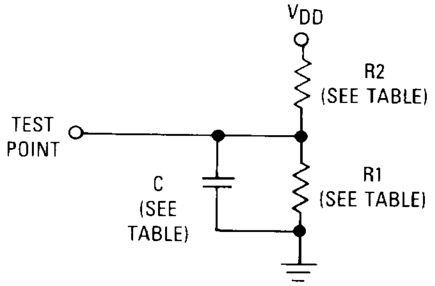


Figure 19. Equivalent Test Load

POWER CONSIDERATIONS

The average chip-junction temperature, T_J, in °C can be obtained from:

$$T_J = T_A + (P_D \cdot \theta_{JA}) \qquad (1)$$

where:

T_A	= Ambient Temperature, °C
θ_{JA}	= Package Thermal Resistance, Junction-to-Ambient, °C/W
P_D	= $P_{INT} + P_{I/O}$
P_{INT}	= $I_{CC} \times V_{CC}$, Watts — Chip Internal Power
$P_{I/O}$	= Power Dissipation on Input and Output Pins — User Determined

For most applications $P_{I/O} < P_{INT}$ and can be neglected.

The following is an approximate relationship between P_D and T_J (if $P_{I/O}$ is neglected):

$$P_D = K \div (T_J + 273°C) \qquad (2)$$

Solving equations (1) and (2) for K gives:

$$K = P_D \cdot (T_A + 273°C) + \theta_{JA} \cdot P_D^2 \qquad (3)$$

where K is a constant pertaining to the particular part. K can be determined from equation (3) by measuring P_D (at equilibrium) for a known T_A. Using this value of K, the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.

DC ELECTRICAL CHARACTERISTICS (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage, $I_{Load} \le$ 10.0 μA	V_{OL} V_{OH}	— V_{DD} − 0.1	— —	0.1 —	V
Output High Voltage (I_{Load} = 0.8 mA) PA7–PA0, PB7–PB0, PC7–PC0, TCMP1, TCMP2 (I_{Load} = 1.6 mA) TDO, SCLK, PLMA, PLMB	V_{OH}	 V_{DD} − 0.8 V_{DD} − 0.8	 V_{DD} − 0.4 V_{DD} − 0.8	 — —	V
Output Low Voltage (I_{Load} = 1.6 mA) PA7–PA0, PB7–PB0, PC7–PC0, TCMP1, TCMP2, PLMA, PLMB, TDO, SCLK RESET	V_{OL}	 — —	 0.1 0.4	 0.4 1.0	V
Input High Voltage PA7–PA0, PB7–PB0, PC7–PC0, PD7–PD0, TCAP1, TCAP2, $\overline{IRQ}$, $\overline{RESET}$, OSC1, RDI	V_{IH}	0.7 × V_{DD}	—	V_{DD}	V
Input Low Voltage PA7–PA0, PB7–PB0, PC7–PC0, PD7–PD0, TCAP1, TCAP2, $\overline{IRQ}$, $\overline{RESET}$, OSC1, RDI	V_{IL}	V_{SS}	—	0.2 × V_{DD}	V
Supply Current (see Notes) RUN (SM = 0) RUN (SM = 1, t_{cyc} = 8 μs) WAIT (SM = 0) WAIT (SM = 1, t_{cyc} = 8 μs) STOP 0 to 70 (Standard) −40 to 85 (Extended) 40 to 125 (Automotive)	I_{DD}	 — — — — — — —	 3.5 0.5 1 0.35 2 — —	 9 2 4 1 10 20 50	 mA mA mA mA μA μA μA
I/O Ports Hi-Z Leakage Current PA7–PA0, PB7–PB0, PC7–PC0, TDO, $\overline{RESET}$, SCLK	I_{IL}	—	0.2	± 1	μA
Input Current $\overline{IRQ}$, TCAP1, TCAP2, OSC1, RDI PD7/AN7–PD0/AN0 (A/D off) PD7/AN7–PD0/AN0 (A/D on)	I_{in}	 — — —	 ± 0.2 ± 0.2 ± 10	 ± 1 ± 1 TBD	μA
Capacitance Ports (as Input or Output), $\overline{RESET}$ TDO, SCLK $\overline{IRQ}$, TCAP1, TCAP2, OSC1, RDI PD7/AN7–PD0/AN0 (A/D off) PD7/AN7–PD0/AN0 (A/D on)	 C_{out} C_{out} C_{in} C_{in} C_{in}	 — — — — —	 — — — 12 22	 12 12 8 TBD TBD	pF

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Wait I_{DD}: Only timer system active (TE = RE = 0). If SCI active (TE = RE = 1) add 10% current draw.
4. Run (Operating) I_{DD}, Wait I_{DD}: Measured using external square wave clock source (f_{osc} = 4.0 MHz), all inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, C_L = 20 pF on OSC2.
5. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, $V_{IH} = V_{DD}$ − 0.2 V.
6. Wait I_{DD} is affected linearly by the OSC2 capacitance.

TBD — To be determined.

DC ELECTRICAL CHARACTERISTICS (V_{DD} = 3.3 Vdc ± 10%, V_{SS} = 0 Vdc, T_A = T_L to T_H, unless otherwise noted)

Characteristic	Symbol	Min	Typ	Max	Unit
Output Voltage, $I_{Load} \le 10.0\ \mu A$	V_{OL} V_{OH}	— V_{DD} − 0.1	— —	0.1 —	V
Output High Voltage	V_{OH}				V
(I_{Load} = 0.2 mA) PA7–PA0, PB7–PB0, PC7–PC0, TCMP1, TCMP2		V_{DD} − 0.3	V_{DD} − 0.1	—	
(I_{Load} = 0.4 mA) TDO, SCLK, PLMA, PLMB		V_{DD} − 0.3	V_{DD} − 0.1	—	
Output Low Voltage	V_{OL}				V
(I_{Load} = 0.4 mA) PA7–PA0, PB7–PB0, PC7–PC0, TCMP1, TCMP2, PLMA, PLMB, TDO, SCLK		—	0.1	0.3	
$\overline{\text{RESET}}$		—	0.2	0.6	
Input High Voltage PA7–PA0, PB7–PB0, PC7–PC0, PD7–PD0, TCAP1, TCAP2, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, OSC1, RDI	V_{IH}	0.7 × V_{DD}	—	V_{DD}	V
Input Low Voltage PA7–PA0, PB7–PB0, PC7–PC0, PD7–PD0, TCAP1, TCAP2, $\overline{\text{IRQ}}$, $\overline{\text{RESET}}$, OSC1, RDI	V_{IL}	V_{SS}	—	0.2 × V_{DD}	V
Supply Current (see Notes)	I_{DD}				
RUN (SM = 0)		—	1.2	5	mA
RUN (SM = 1, t_{cyc} = 8 µs)		—	0.2	1	mA
WAIT (SM = 0)		—	0.4	2	mA
WAIT (SM = 1, t_{cyc} = 8 µs)		—	0.15	0.5	mA
STOP					
0 to 70 (Standard)		—	1	10	µA
−40 to 85 (Extended)		—	—	10	µA
−40 to 125 (Automotive)		—	—	30	µA
I/O Ports Hi-Z Leakage Current PA7–PA0, PB7–PB0, PC7–PC0, TDO, $\overline{\text{RESET}}$, SCLK	I_{IL}	—	±0.2	±10	µA
Input Current	I_{in}				µA
$\overline{\text{IRQ}}$, TCAP1, TCAP2, OSC1, RDI		—	±0.2	±1	
PD7/AN7–PD0/AN0 (A/D off)		—	±0.2	±1	
PD7/AN7–PD0/AN0 (A/D on)		—	±10	TBD	
Capacitance					pF
Ports (as Input or Output), $\overline{\text{RESET}}$, TDO	C_{out}	—	—	12	
TDO, SCLK	C_{out}	—	—	12	
$\overline{\text{IRQ}}$, TCAP1, TCAP2, OSC1, RDI	C_{in}	—	—	8	
PD7/AN7–PD0/AN0 (A/D off)	C_{in}	—	12	TBD	
PD7/AN7–PD0/AN0 (A/D on)	C_{in}	—	22	TBD	

NOTES:

1. All values shown reflect average measurements.
2. Typical values at midpoint of voltage range, 25°C only.
3. Wait I_{DD}: Only timer system active (TE = RE = 0). If SCI active (TE = RE = 1) add 10% current draw.
4. Run (Operating) I_{DD}, Wait I_{DD}: Measured using external square wave clock source (f_{osc} = 4.0 MHz), all inputs 0.2 V from rail; no dc loads, less than 50 pF on all outputs, C_L = 20 pF on OSC2.
5. Wait, Stop I_{DD}: All ports configured as inputs, V_{IL} = 0.2 V, V_{IH} = V_{DD} − 0.2 V.
6. Wait I_{DD} is affected linearly by the OSC2 capacitance.

TBD — To be determined.

A/D CONVERTER CHARACTERISTICS (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc)

Characteristic	Parameter	Min	Max	Unit
Resolution	Number of bits resolved by the A/D	8	—	Bit
Non-Linearity	Maximum deviation from the best straight line through the A/D transfer characteristics (V_{RH} = V_{DD} and V_{RL} = 0 V)	—	± ½	LSB
Quantization Error	Uncertainty due to converter resolution	—	± ½	LSB
Absolute Accuracy	Difference between the actual input voltage and the full-scale equivalent of the binary code output code for all errors	—	± 1	LSB
Conversion Range	Analog input voltage range	V_{RL}	V_{RH}	V
V_{RH}	Maximum analog reference voltage	V_{RL}	V_{DD} + 0.1	V
V_{RL}	Minimum analog reference voltage	V_{SS} − 0.1	V_{RH}	V
Conversion Time	Total time to perform a single analog to digital conversion a. External Clock (XTAL, EXTAL) b. Internal RC oscillator	 — —	 32 32	 t_{cyc} µs
Monotonicity	Conversion result never decreases with an increase in input voltage and has no missing codes	Guaranteed		
Zero-Input Reading	Conversion result when V_{in} = V_{RL}	00	—	Hex
Full-Scale Reading	Conversion result when V_{in} = V_{RH}	—	FF	Hex
Sample Acquisition Time (see Note 1)	Analog input acquisition sampling a. External Clock (XTAL, EXTAL) b. Internal RC oscillator	 — —	 12 12	 t_{cyc} µs
Sample Hold Capacitance	Input capacitance on PD7/AN7–PD0/AN0	—	12	pF
Input Leakage (see Note 2)	Input leakage on A/D pins PD7/AN7–PD0/AN0, V_{RL}, V_{RH}	— —	1 1	µA

NOTES:

1. Source impedances greater than 10K ohm will adversely affect internal RC charging time during input sampling.
2. The external system error caused by input leakage current is approximately equal to the product of R source and input current.

CONTROL TIMING (V_{DD} = 5.0 Vdc ± 10%, V_{SS} = 0 Vdc, T_A = T_L to T_H)

Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Crystal Option External Clock Option	f_{osc}	 — dc	 4.2 4.2	MHz
Internal Operating Frequency Crystal ($f_{osc}/2$) External Clock ($f_{osc}/2$)	f_{op}	 — dc	 2.1 2.1	MHz
Cycle Time (see Figure 21)	t_{cyc}	480	—	ns
Crystal Oscillator Startup Time (see Figure 21)	t_{OXOV}	—	100	ms
Stop Recovery Startup Time (Crystal Oscillator)	t_{ILCH}	—	100	ms
External RESET Input Pulse Width (see Figure 21)	t_{RL}	1.5	—	t_{cyc}
Power-On RESET Output Pulse Width 4064 Cycle Option 16 Cycle Option	t_{PORL}	 4064 16	 — —	t_{cyc}
Watchdog RESET Output Pulse Width	t_{DOGL}	1.5	—	t_{cyc}
Watchdog Time-Out	t_{DOG}	6144	7168	t_{cyc}
EEPROM Byte Erase Time 0 to 70 (Standard) 40 to 85 (Extended) −40 to 125 (Automotive)	t_{ERA}	 10 10 10	 — — —	ms
EEPROM Byte Programming Time 0 to 70 (Standard) 40 to 85 (Extended) 40 to 125 (Automotive)	t_{PROG}	 10 10 20	 — — —	ms
Timer Resolution** Input Capture Pulse Width (see Figure 20) Input Capture Pulse Period (see Figure 20)	 t_{RESL} t_{TH}, t_{TL} t_{TL}, t_{TL}	 4.0 125 ***	 — — —	 t_{cyc} ns t_{cyc}
Interrupt Pulse Width (Edge-Triggered)	t_{ILIH}	125	—	ns
Interrupt Pulse Period	t_{ILIL}	*	—	t_{cyc}
OSC1 Pulse Width	t_{OH}, t_{OL}	90	—	ns

NOTES:

*The minimum period t_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{cyc}.

**Since a 2-bit prescaler in the timer must count four internal cycles (t_{cyc}), this is the limiting minimum factor in determining the timer resolution.

***The minimum period t_{TLTL} should not be less than the number of cycle times it takes to execute the capture interrupt service routine plus 24 t_{cyc}.

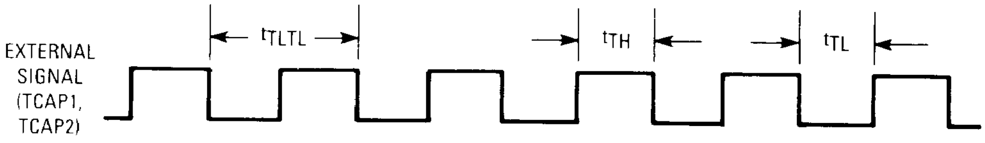


Figure 20. Timer Relationship

CONTROL TIMING (V_{DD} = 3.3 Vdc ± 10%, V_{SS} = 0 Vdc, $T_A = T_L$ to T_H)

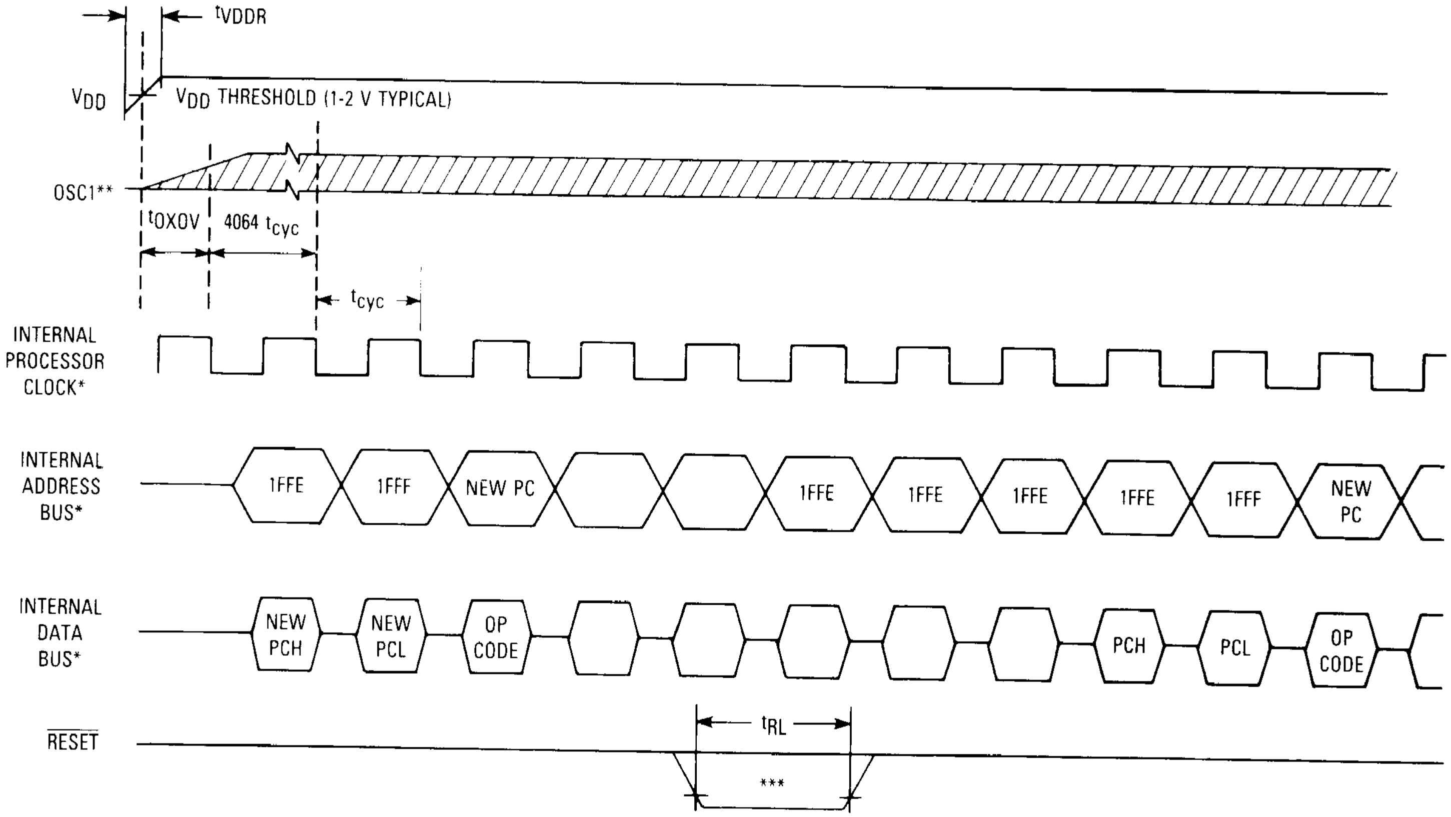
Characteristic	Symbol	Min	Max	Unit
Frequency of Operation Crystal Option External Clock Option	f_{osc}	 — dc	 2.0 2.0	MHz
Internal Operating Frequency Crystal ($f_{osc}/2$) External Clock ($f_{osc}/2$)	f_{op}	 — dc	 1.0 1.0	MHz
Cycle Time (see Figure 21)	t_{cyc}	1000	—	ns
Crystal Oscillator Startup Time (see Figure 21)	t_{OXOV}	—	100	ms
Stop Recovery Startup Time (Crystal Oscillator)	t_{ILCH}	—	100	ms
External RESET Input Pulse Width (see Figure 21)	t_{RL}	1.5	—	t_{cyc}
Power-On RESET Output Pulse Width 4064 Cycle Option 16 Cycle Option	t_{PORL}	 4064 16	 — —	t_{cyc}
Watchdog RESET Output Pulse Width	t_{DOGL}	1.5	—	t_{cyc}
Watchdog Time-Out	t_{DOG}	6144	7168	t_{cyc}
EEPROM Byte Erase Time 0 to 70 (Standard) −40 to 85 (Extended) −40 to 125 (Automotive)	t_{ERA}	 30 TBD TBD	 — — —	ms
EEPROM Byte Programming Time 0 to 70 (Standard) −40 to 85 (Extended) −40 to 125 (Automotive)	t_{PROG}	 30 TBD TBD	 — — —	ms
Timer Resolution** Input Capture Pulse Width (see Figure 20) Input Capture Pulse Period (see Figure 20)	 t_{RESL} t_{TH}, t_{TL} t_{TL}, t_{TL}	 4.0 250 ***	 — — —	 t_{cyc} ns t_{cyc}
Interrupt Pulse Width (Edge-Triggered)	t_{ILIH}	250	—	ns
Interrupt Pulse Period	t_{ILIL}	*	—	t_{cyc}
OSC1 Pulse Width	t_{OH}, t_{OL}	200	—	ns

NOTES:

*The minimum period t_{ILIL} should not be less than the number of cycle times it takes to execute the interrupt service routine plus 21 t_{cyc}.

**Since a 2-bit prescaler in the timer must count four internal cycles (t_{cyc}), this is the limiting minimum factor in determining the timer resolution.

***The minimum period t_{TLTL} should not be less than the number of cycle times it takes to execute the capture interrupt service routine plus 24 t_{cyc}.



*Internal timing signal and bus information not available externally.

**OSC1 line is not meant to represent frequency. It is only used to represent time.

***The next rising edge of the internal processor clock following the rising edge of $\overline{\text{RESET}}$ initiates the reset sequence.

Figure 21. Power-On Reset and $\overline{\text{RESET}}$

ORDERING INFORMATION

The following information is required when ordering a custom MCU. The information may be transmitted to Motorola in the following media:

MS™-DOS/PC-DOS disk file (360K)
EPROM(s) 2764, MCM68764, MCM68766, or EEPROM MC68HC805C4

To initiate a ROM pattern for the MCU, it is necessary to first contact the local field service office, a sales person, or a Motorola representative.

FLEXIBLE DISKS

A flexible disk (MS-DOS/PC-DOS disk file), programmed with the customer's program may be submitted for pattern generation. The diskette should be clearly labeled with the customer's name, data, project or product name, and the name of the file containing the pattern.

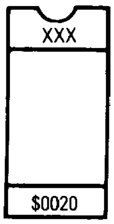
MS-DOS is Microsoft's Disk Operating System. PC-DOS is the IBM® Personal Computer (PC) Disk Operating System. Disk media submitted must be a standard density (360K) double-sided 5 1/4 inch compatible floppy diskette. The diskette must contain object file code in Motorola's S-record format. The S-record format is a character-based object file format generated by M6805 cross assemblers and linkers on IBM PC style machines.

In addition to the program pattern, a file containing the program source code listing can be included. This data will be kept confidential and used to expedite the process in case of any difficulty with the pattern file.

EPROMs

A 2764, 68764, or 68766 type EPROM, programmed with the customer's program (positive logic sense for address and data), may be submitted for pattern generation. Since all program and data space information will fit on one 2764, 68764, or 68766 EPROM device, the EPROM must be programmed as described in the following paragraphs.

For an MC68HC805B6 MCU start the page zero, user ROM at EEPROM address $0020 through $004F. Start the user ROM at EEPROM address $0800 through $1EFF with vectors from $1FF0 to $1FFF. All unused bytes, including the user's space, must be set to zero. For shipment to Motorola, EPROMs should be placed in a conductive IC carrier and packed securely. Styrofoam is not acceptable for shipment.



xxx = Customer ID

Verification Media

All original pattern media (EPROMs or floppy disks) are filed for contractual purposes and are not returned. A computer listing of the ROM code will be generated and returned along with a listing verification form. The listing should be thoroughly checked, and the verification form should be completed, signed, and returned to Motorola. The signed verification form constitutes the contractual agreement for the creation of the customer mask. To aid in the verification process, Motorola will program *customer supplied* blank EPROM(s) or DOS disks from the data file used to create the custom mask.

ROM VERIFICATION UNITS (RVUs)

Ten MCUs containing the customer's ROM pattern will be sent for program verification. These units will have been made using the custom mask, but are for the purpose of ROM verification only. For expediency, the MCUs are unmarked, packaged in ceramic, and tested with five volts at room temperature. These RVUs are free with the minimum order quantity, but are not production parts. RVUs are not backed or guaranteed by Motorola Quality Assurance.

ORDERING INFORMATION

The following table provides ordering information pertaining to the package type, temperature, and MC order numbers for the MC68HC05B6 device.

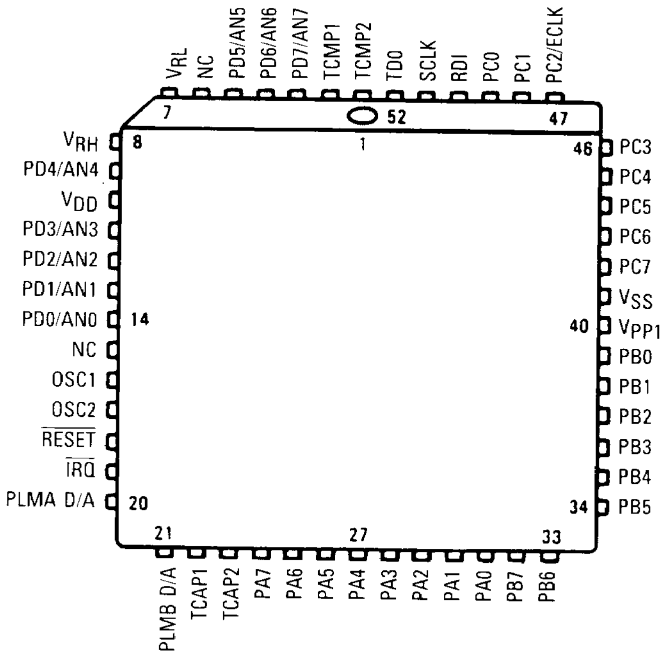
Package Type	Temperature	MC Order Number
Plastic (P Suffix)	0°C to +70°C −40°C to +85°C −40°C to +125°C	MC68HC05B6P MC68HC05B6CP MC68HC05B6MP
PLCC (FN Suffix)	0°C to +70°C −40°C to +85°C −40°C to +125°C	MC68HC05B6FN MC68HC05B6CFN MC68HC05B6MFN

MS is a trademark of Microsoft, Inc.
IBM is a registered trademark of International Business Machines Corporation.

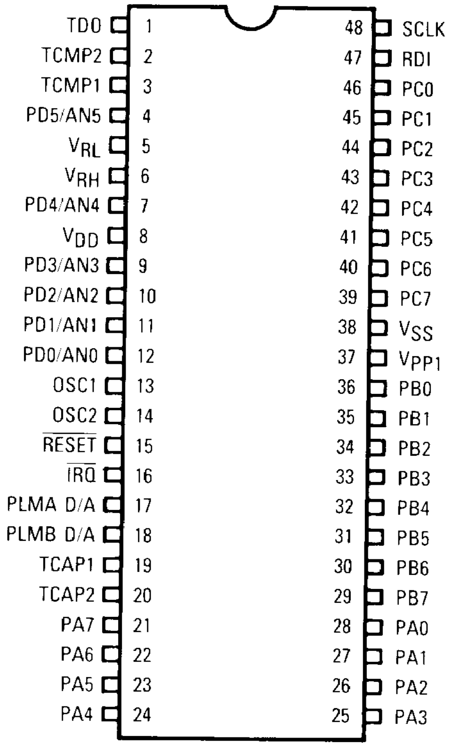
MECHANICAL DATA

PIN ASSIGNMENTS

52-Pin PLCC

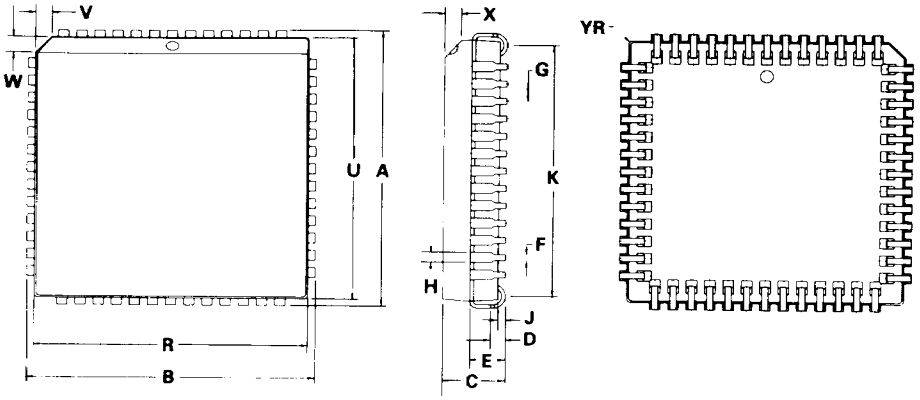


48-Pin Dual-in-Line Package



PACKAGE DIMENSIONS

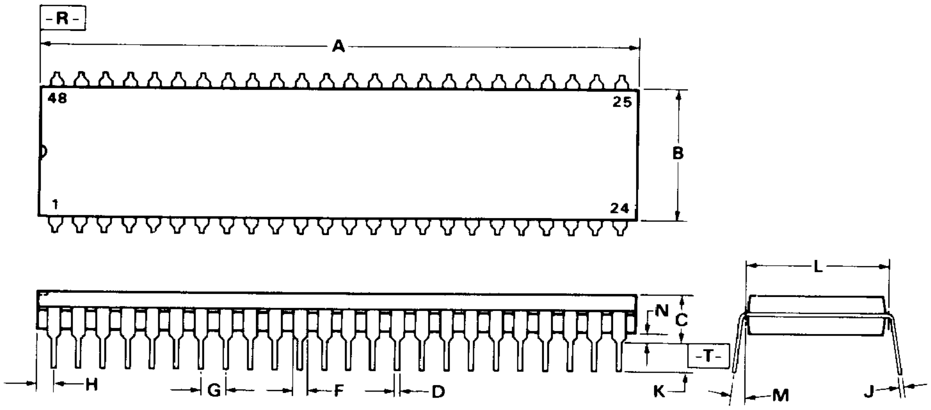
FN SUFFIX
QUAD PACK
CASE 778-01



NOTES:
1. DIMENSIONS R AND U DO NOT INCLUDE MOLD FLASH.
2. DIMENSIONING AND TOLERANCING PER ANSI Y14.5M, 1982.
3. CONTROLLING DIMENSION: INCH

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	19.94	20.19	0.785	0.795
B	19.94	20.19	0.785	0.795
C	4.19	4.57	0.165	0.180
D	0.64	1.01	0.025	0.040
E	2.16	2.79	0.085	0.110
F	0.33	0.53	0.013	0.021
G	1.27 BSC		0.050 BSC	
H	0.66	0.81	0.026	0.032
J	0.38	0.63	0.015	0.025
K	17.52	18.54	0.690	0.730
R	19.05	19.20	0.750	0.756
U	19.05	19.20	0.750	0.756
V	1.07	1.21	0.042	0.048
W	1.07	1.21	0.042	0.048
X	1.07	1.42	0.042	0.056
Y	0.00	0.50	0.000	0.020

P SUFFIX
PLASTIC
CASE 767-02



NOTES:
1. -R- IS END OF PACKAGE DATUM PLANE.
-T- IS BOTH A DATUM AND SEATING PLANE.
2. POSITIONAL TOLERANCE FOR LEADS 1 AND 48:
⊕ | 0.51 (0.020) | T | B Ⓜ | R
POSITIONAL TOLERANCE FOR LEAD PATTERN:
⊕ | 0.25 (0.010) | T | B Ⓜ
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.
4. DIMENSION L IS TO CENTER OF LEADS WHEN FORMED PARALLEL.
5. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1982.
6. CONTROLLING DIMENSION: INCH

	MILLIMETERS		INCHES	
DIM	MIN	MAX	MIN	MAX
A	61.34	62.10	2.415	2.445
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.55	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.79 BSC		0.070 BSC	
J	0.20	0.38	0.008	0.015
K	2.92	3.42	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.01	0.020	0.040

Motorola reserves the right to make changes without further notice to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. Motorola products are not authorized for use as components in life support devices or systems intended for surgical implant into the body or intended to support or sustain life. Buyer agrees to notify Motorola of any such intended end use whereupon Motorola shall determine availability and suitability of its product or products for the use intended. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Employment Opportunity/Affirmative Action Employer.

Literature Distribution Centers:

USA: Motorola Literature Distribution; P.O. Box 20912; Phoenix, Arizona 85036.
EUROPE: Motorola Ltd.; European Literature Center; 88 Tanners Drive, Blakelands Milton Keynes, MK145BP, England.
ASIA PACIFIC: Motorola Semiconductors H.K. Ltd.; P.O. Box 80300; Cheung Sha Wan Post Office; Kowloon Hong Kong.

MOTOROLA

A23067-1 PRINTED IN USA 2-89 IMPERIAL LITHO C62304 18,000 MCU