Field Programmable Microcontrollers (FPMs) from Texas Instruments
The inexpensive ROM alternative

TEXAS INSTRUMENTS

Field Programmable Microcontrollers (FPMs)

Data sheets

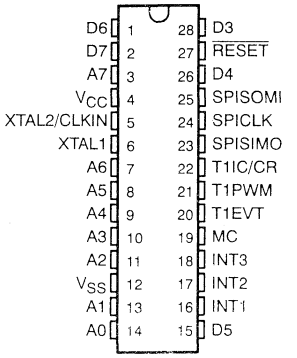
Contents

TEXAS INSTRUMENTS

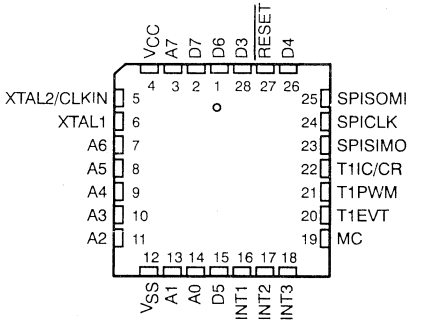
SPNS012B — MAY 1987 — REVISED DECEMBER 1990

- CMOS/EEPROM/EPROM Technologies on a Single Device
  - — Form Factor Emulator (FFE) Devices for Prototyping Purposes
  - — One-Time Programmable (OTP) Devices for Low Volume Production
  - — Mask ROM Devices for High Volume Production
- Flexible Operating Features
  - — Power Reduction STANDBY and HALT Modes
  - — Commercial and Industrial Temperature Range
  - — Input Clock Frequency 2 MHz to 20 MHz
  - — Voltage (V_{CC}) 5 V ± 10%
- Internal System Memory Configurations
  - — 4K-Byte On-Chip Program Memory
  - — Mask ROM (TMS370C010 and TMS370C310) or
  - — EPROM (TMS370C710 and TMS370C610)
  - — Data EEPROM, 256 Bytes (TMS370C010 and TMS370C710)
  - — Static RAM, 128 Bytes Usable as Registers
- 16-Bit General Purpose Timer
  - — Software Configurable for a 16-Bit Event Counter, or a 16-Bit Pulse Accumulator, or a 16-Bit Input Capture Function, or Two Compare Registers, or a Self-Contained PWM Output Function
  - — Software Programmable Input Polarity
  - — 8-Bit Prescaler, Providing a 24-Bit Real-time Timer
- On-Chip 24-Bit Watchdog Timer
- Serial Peripheral Interface (SPI)
  - — Variable-Length High-Speed Shift Register
  - — Synchronous Master/Slave Operation
- Flexible Interrupt Handling
  - — Two S/W Programmable Interrupt Levels
  - — Global and Individual Interrupt Masking
  - — Programmable Rising or Falling Edge Detect
- 22 CMOS/TTL Compatible I/O Pins
  - — All Peripheral Function Pins Software Configurable for Digital I/O
  - — 21 Bidirectional Pins, 1 Input Pin

J and N Packages
(Top View)



FZ and FN Packages
(Top View)



- 28-Pin Packages
  - — Plastic DIP – ROM or OTP (N Suffix)
  - — Ceramic DIP – FFE (J Suffix)
  - — Plastic PLCC – ROM or OTP (FN Suffix)
  - — Ceramic CLCC – FFE (FZ Suffix)
- TMS370 Series Compatibility
  - — Register-to-Register Architecture
  - — 128 General-Purpose Registers
  - — 14 Powerful Addressing Modes
- PC-Based Workstation Development Support Emphasizes Productivity, Featuring:
  - — Realtime In-Circuit Emulation
  - — Symbolic Debug
  - — Extensive Breakpoint/Trace Capability
  - — C-Compiler Support
  - — Multi-Window User Interface
  - — EEPROM/EPROM Programming

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

description

The TMS370C010, TMS370C310, TMS370C610, and TMS370C710 devices are members of the TMS370 family of single-chip 8-bit microcontrollers. The TMS370 family provides cost-effective realtime system control through VLSI integration of advanced on-chip memory and peripheral function modules.

The TMS370 family is implemented using high-performance silicon-gate CMOS technology. The low operating power, wide operating temperature range, and high noise immunity of CMOS technology coupled with the high performance and extensive on-chip peripheral functions make the TMS370Cx10 devices attractive in system designs for automotive electronics, industrial control, computer peripheral control, and telecommunications.

Unless otherwise noted, the term TMS370Cx10 refers to the TMS370C010, TMS370C310, TMS370610, and the TMS370C710 for all peripheral function modules available on those devices. All TMS370Cx10 devices contain the following modules:

- 4 K bytes Program memory.
- 128 bytes Ram (usable as registers).
- Serial Peripheral Interface (SPI).
- Timer1-16-bit general-purpose timer with Watchdog

The following table provides a memory configuration overview of the TMS370Cx10 devices.

DEVICE	PROGRAM MEMORY	DATA EEPROM	PACKAGE
TMS370C010	4K Bytes ROM	256 Bytes	N – DIP FN – PLCC
TMS370C310	4K Bytes ROM	None	N – DIP FN – PLCC
TMS370C610	4K Bytes EPROM	None	N – DIP FN – PLCC
TMS370C710	4K Bytes EPROM	256 Bytes	N – DIP FN – PLCC
SE370C710†	4K Bytes EPROM	256 Bytes	J – CDIP FZ – CLCC

† System evaluators and development tools are for use only in a prototype environment and their reliability has not been characterized.

The 4K bytes of mask-programmable ROM in the TMS370C010 and TMS370C310 are replaced in the TMS370C710 and TMS370C610 with 4K bytes of EPROM while all other available memory and on-chip peripherals are identical, with the exception of no Data EEPROM on the TMS370C310 and TMS370C610 devices. The TMS370C610 and TMS370C710 OTP devices allow customers to change code without reordering a MASK ROM device. The OTP devices reduce manufacturing design times for new code and are excellent low volume production alternatives.

The windowed ceramic FFE devices, SE370C710 FZ, J, offer reprogramming capabilities for prototyping requirements.

The TMS370Cx10 provides two power reduction modes (STANDBY and HALT) for applications where low power consumption is critical. Both modes stop all CPU activity (i.e., no instructions are executed). In the STANDBY mode the internal oscillator and the general purpose timer remain active. In the HALT mode, all device activity is stopped. The device retains all RAM data and peripheral configuration bits throughout both powerdown modes.

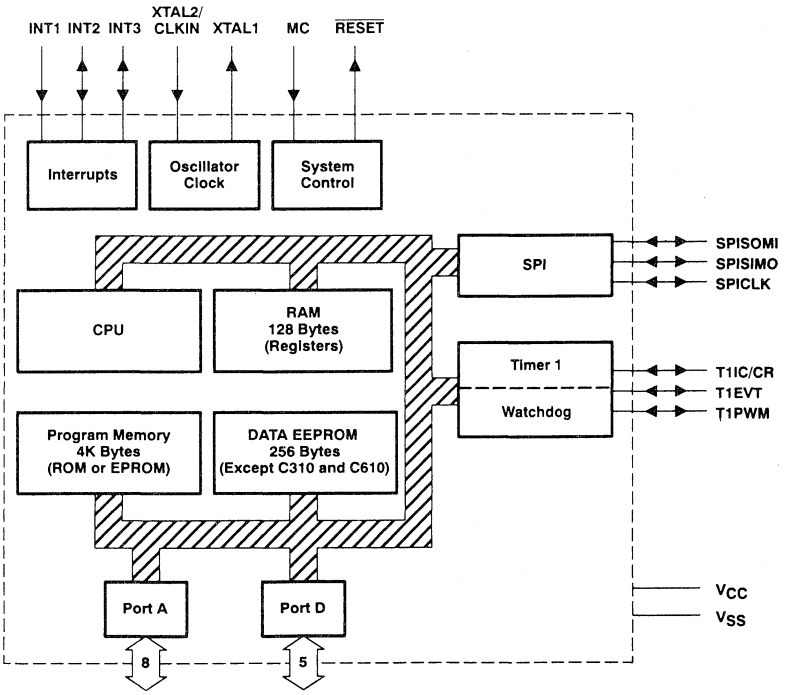
The TMS370Cx10 features advanced register-to-register architecture that allows direct arithmetic and logical operations without requiring an accumulator (e.g., ADD R24, R47; add the contents of register 24 to the contents of register 47 and store the result in register 47). The TMS370Cx10 instruction set is fully compatible with other TMS370 family members, allowing easy transition between members.

The SPI gives a convenient method of serial interaction for high speed communications between simpler shift register type devices, such as display drivers, A/D converters, PLL, I/O expansion, or other microcontrollers in the system.

The TMS370 family provides the system designer with an economical, efficient solution to realtime control applications. The TMS370 family eXtended Development System (XDS) solves the challenge of efficiently developing the software and hardware required to design the TMS370 into an ever-increasing number of complex applications. The application source code modules can be written in Assembly and/or C language and the output code can be generated by the linker. The TMS370 family XDS communicates via a standard RS-232-C interface with an existing personal computer to form a PC-DOS hosted workstation, using the PC's editors and software utilities already familiar to the designer. The TMS370 family XDS emphasizes ease-of-use through extensive use of menus and screen windowing so that a system designer can begin developing software with minimum training. Precise realtime in-circuit emulation and extensive symbolic debug and analysis tools ensure efficient software and hardware implementation as well as reduced time-to-market.

The TMS370Cx10 mask ROM devices, the TMS370x10 OTP devices, together with the TMS370 family XDS for applications development, the SE370C710 EPROM devices, and comprehensive product documentation and customer support provide a complete solution to the low end needs of the systems designer. Other TMS370 family members are available with additional memory, I/O and peripheral functions such as SCI and A/D to meet more complex system requirements.

functional block diagram



pin descriptions

PIN NAME	PIN NO.	I/O	DESCRIPTION
A0	14	I/O	Port A is a general purpose bidirectional port.
A1	13	I/O	
A2	11	I/O	
A3	10	I/O	
A4	9	I/O	
A5	8	I/O	
A6	7	I/O	
A7	3	I/O	
D3	28	I/O	General purpose bidirectional pin. Also configurable as CLKOUT.
D4	26	I/O	General purpose bidirectional pin.
D5	15	I/O	General purpose bidirectional pin.
D6	1	I/O	General purpose bidirectional pin.
D7	2	I/O	General purpose bidirectional pin.
INT1	16	I	External interrupt non-maskable or maskable interrupt / general purpose input pin.
INT2	17	I/O	External maskable interrupt input/general purpose bidirectional pin.
INT3	18	I/O	External maskable interrupt input/general purpose bidirectional pin.
T1IC/CR	22	I/O	Timer 1 Input Capture/Counter Reset input pin/general purpose bidirectional pin.
T1PWM	21	I/O	Timer 1 PWM output pin/general purpose bidirectional pin.
T1EVT	20	I/O	Timer 1 External Event input pin/general purpose bidirectional pin.
SPISOMI	25	I/O	SPI Slave Output pin, Master Input pin/general purpose bidirectional pin.
SPISIMO	23	I/O	SPI Slave Input pin, Master Output pin/general purpose bidirectional pin.
SPICLK	24	I/O	SPI bidirectional Serial Clock pin/general purpose bidirectional pin.
$\overline{\text{RESET}}$	27	I/O	System reset bidirectional pin. As an input, it initializes microcontroller. As open-drain output it indicates an internal failure was detected by the Watchdog or Oscillator Fault circuit.
MC	19	I	Mode control input pin; enables EEPROM Write Protection Override (WPO) mode. Normal operation = 0 V, WPO = 12 V. V_{PP} supply for EPROM programming.
XTAL2/ CLKIN	5	I	Internal oscillator crystal input/external clock source input.
XTAL1	6	O	Internal oscillator output for crystal.
V_{CC}	4		Positive supply voltage.
V_{SS}	12		Ground reference.

NOTE 1: Each pin associated with Interrupt 2, Interrupt 3, Timer 1, and SPI functional blocks may be individually programmed as a general purpose bidirectional pin if it is not used for its primary block function.

memory map

The TMS370 family architecture is based on the Von Neumann architecture, where the program memory and data memory share a common address space. All peripheral input/output is memory mapped in this same common address space. As shown in Figure 1, the TMS370 provides memory-mapped RAM, ROM, EEPROM, EPROM, input/output pins, and peripheral functions.

The peripheral file contains all input/output port control, peripheral status and control, EEPROM memory programming, and system-wide control functions. The peripheral file is located from 1010h to 104Fh and is logically divided into 5 Peripheral File Frames of 16 bytes each. Each on-chip peripheral is assigned to a separate frame through which peripheral control and data information is passed. The TMS370Cx10 has its peripherals and system control assigned to Peripheral File Frames 1 through 4, addresses 1010h through 104Fh.

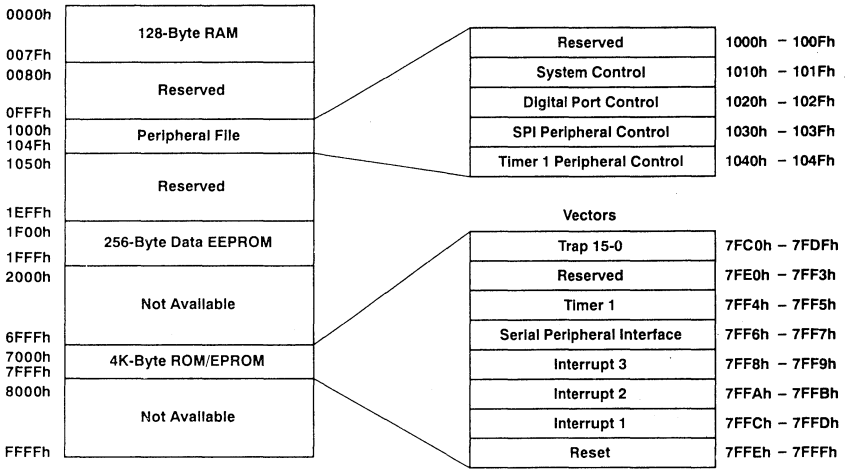


Figure 1. TMS370Cx10 Memory Map

memories

RAM/register file

The TMS370Cx10 has 128 bytes of static RAM, which serve as both the CPU register file and general-purpose memory. The RAM is treated as registers by the instruction set and is referenced as R0 through R127. The first two registers, R0 and R1, are also called the A and B registers, respectively. The stack is located in the RAM, and operates as a last-in first-out read/write memory. It is used to store the return address on subroutine calls and the status register during interrupts. Accessing this memory as registers is performed in one system clock cycle (t_c), while general-purpose memory access is performed in two system clock cycles.

data EEPROM

The TMS370C010 and TMS370C710 have 256 bytes of on-chip Electrically Erasable Programmable ROM (EEPROM), addressed as 256 consecutive bytes mapped from locations 1F00h to 1FFFh. The Data EEPROM provides nonvolatile programmable storage for items such as calibration constants and configuration information for personalization of a generic algorithm for use in specific end applications. The Data EEPROM supports bit, byte, and block write/erase modes. Instructions may be executed from Data EEPROM, providing additional program space and the ability to patch algorithms by placing a branch table for volatile routines in Data EEPROM.

The Data EEPROM uses the 5-V V_{CC} supply voltage and provides the programming voltage via an internal dedicated generator, eliminating the need for an external high-voltage programming source. The dedicated voltage generator optimizes the programming voltage characteristics, increasing the reliability as well as extending the write/erase endurance of the array.

Programming control and status monitoring are performed through the Data EEPROM control register (DEECTL) in the peripheral file. An EEPROM write/erase operation is performed in the following sequence:

1. Perform normal memory write to the target EEPROM location.
2. Write to DEECTL control register to select WRITE1/WRITE0 and set the EXECUTE (EXE) bit to 1.
3. Wait for program time to elapse [$t_{w(PGM)B}$ or $t_{w(PGM)AR}$].
4. Write to DEECTL control register to set the EXECUTE (EXE) bit to 0.

The WRITE1/WRITE0 control bit selects whether the zeros or the ones in the data byte are to be programmed into the selected EEPROM location. For example, a WRITE1 operation will program ones into all bit positions within the EEPROM byte that have ones in the data byte, while bits that are zero in the data byte will not affect the EEPROM contents. The WRITE1 operation effectively performs a logical OR of the information previously stored on the EEPROM byte with the data byte. The WRITE0 operation effectively performs a logical AND between these two bytes. Single bit programming within an EEPROM byte is performed by writing only the zeros or ones of the data byte. The EEPROM programming algorithm may use this bit-programming capability to optimize the life of the EEPROM.

When a data value cannot be achieved by writing only zeros or only ones into the EEPROM byte, a WRITE1 followed by a WRITE0 will program any data value into the EEPROM byte, regardless of the previous data stored at that location.

All unprotected bytes within the Data EEPROM array may be programmed during a single EEPROM programming cycle by setting the ARRAY PROG bit of DEECTL to 1 at the start of the programming cycle.

Data EEPROM read accesses are performed as normal memory read operations in two system clock cycles. A memory read cycle to any EEPROM location while EXE = 1 returns the value currently being written to the EEPROM. Following an EEPROM write operation, the EEPROM voltages must stabilize prior to performing an EEPROM read operation. The BUSY FLAG indicates the status of the EEPROM voltage. When set, the EEPROM is not ready for a read operation. The BUSY flag is reset to 0 by the EEPROM control logic when 128

system clock cycles have elapsed following the EXE bit being set to 0. If an EEPROM read operation is performed while BUSY = 1, automatic WAIT states will be generated until BUSY = 0, and then the read operation will be performed.

Bytes within the Data EEPROM may be protected from inadvertent overwriting of critical information. As shown in Figure 2, the 8-bit Write Protect Register (WPR), located at 1F00h within the data EEPROM, provides write protection for the 256-byte Data EEPROM, segmenting the array into eight blocks of 32 bytes each. Each of these 32-byte blocks may be individually write- and erase-protected by setting the corresponding bit to 1 in the WPR. Since the WPR resides in the array in BLK0, the WPR may also be write-protected, thereby increasing the system reliability by preventing bytes from being reprogrammed. Bytes left unprotected may be written to by the normal EEPROM programming sequence. The Write Protection Override (WPO) mode overrides the write protection of all blocks in the Data EEPROM, and enables data to be written to any location in the Data EEPROM, regardless of the WPR contents. Enter the WPO mode by placing 12-V on the MC pin. The WPO mode is typically used in a service environment to update the protected EEPROM contents. The 12-V input level on the MC pin to enter the WPO mode is not normally present in an application except in a service environment; therefore, the data integrity of the program is ensured during normal operation.

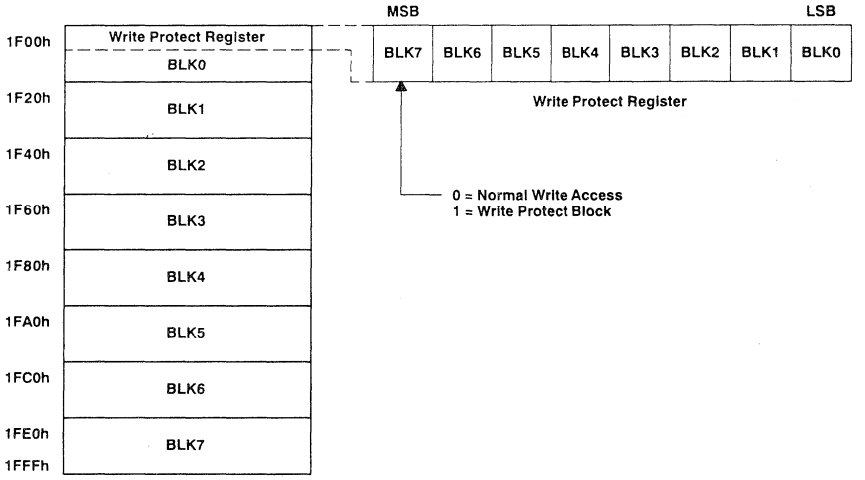


Figure 2. Write Protect Register

program ROM

The Program ROM consists of 4K bytes of mask programmable read-only memory. The Program ROM is used for permanent storage of data and instructions, with read operations performed in two system clock cycles. Memory addresses 7FF4h through 7FFFh are reserved for interrupt and reset vectors. Trap vectors, used with TRAP0 through TRAP15 instructions, are located between addresses 7FC0h and 7FDFh. Programming of the mask ROM is performed at the time of device fabrication.

program EPROM

The Program EPROM of the TMS370C610 and TMS370C710 is a 4K ultraviolet-light-erasable, electrically programmable read-only memory, addressed as 4K consecutive bytes mapped form location 7000h to 7FFFh. It provides application performance identical to the TMS370Cx10 mask ROM devices with up to 4K bytes. Program instructions are read from the Program EPROM in two system clock cycles, providing the prototyping capability of the Mask Program ROM.

An external supply (V_{PP}) is needed at the MC pin to provide the necessary voltage (V_{PP}) for programming. Programming is controlled through the EPCTL register (P01C) in the peripheral file.

Before programming, the SE370C710's EPROM must be erased by exposing the device through the transparent window to high-intensity ultraviolet light (wavelength 2537Å). The recommended minimum exposure dose (UV intensity × exposure time) is 15- W•s/cm^2. A typical 12-mW/cm^2, filterless UV lamp will erase the device in 21 minutes. The lamp should be located about 2.5 cm above the chip during erasure. After erasure, the entire array is in a logic high state. A programmed low can be erased only by exposure to ultraviolet light. It should be noted that normal ambient light contains the correct wavelength for erasure. Therefore, when using the SE370C710, the window should be covered with an opaque label. All devices are erased to logical high when delivered from the factory.

CAUTION

Exposing the EPROM module to ultraviolet light may also cause erasure in any EEPROM module. Any useful data stored in the EEPROM must be reprogrammed after exposure to UV light.

Programming lows into the EPROM is controlled by the EPCTL register via the EXE bit and the VPPS bit. The EXE bit initiates EPROM programming when set and disables programming when cleared. The VPPS bit connects the programming voltage V_{PP} at the MC pin to the EPROM module. VPPS (EPCTL.6) and EXE (EPCTL.0) should be set separately, and the VPPS bit should be set at least two microseconds before the EXE bit is set. After programming, the application programming should wait for four microseconds before any read attempt is made. The programming operation (see Figure 3) is performed in the following recommended sequence:

1. Supply the programming voltage to the MC pin.
2. Write to EPCTL register to set the VPPS bit to 1 (high).
3. Perform normal memory write register to the target EPROM location.
4. Write to EPCTL register to set the EXE bit register to 1 (high). (Wait at least two microseconds after step 2.
5. Wait for program time to elapse (one millisecond).
6. Write to EPCTL register to clear the EXE bit (leave VPPS bit set to 1 (high)).
7. Read the byte being programmed; if correct data is not read, repeat steps 4 through 6 up to a maximum X of 25.
8. Write to EPCTL register to set the EXE bit to 1 (high) for Final programming.
9. Wait for program time to elapse (3X milliseconds duration).
10. Write to EPCTL register to clear the EXE and VPPS bits.

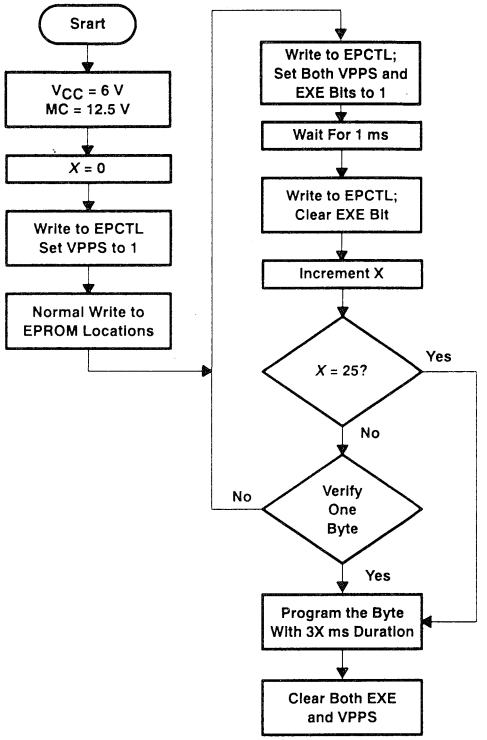


Figure 3. EPROM Programming Operation

An external power supply at V_{PP}, I_{PP} (30 mA), is required for programming operation. Programming voltage V_{PP} is supplied via the MC pin. This also automatically puts the microcontroller in the Write Protection Override (WPO) mode. Programming voltage may be applied via the MC pin anytime after reset and remain at V_{PP} after programming (after the EXE bit is cleared). Applying programming voltage while $\overline{\text{RESET}}$ is active will put the microcontroller in a reserved mode, where programming operation is inhibited.

write protect of program EPROM

To override the EPROM write protection, the V_{PP} voltage must be applied to the MC pin and the VPPS bit (EPCTL.6) must be set. This dual requirement ensures that the program EPROM will not accidentally be overwritten during the Data EEPROM operations when V_{PP} is applied to the MC pin. The Data EEPROM may be programmed when the VPPS bit is set.

central processing unit

The central processing unit (CPU) of the TMS370 series is an enhanced version of the TMS7000 Family CPU. The enhancements include additional user instructions such as integer divide, conditional jump instructions based on the overflow status bit, and addressing modes such as stack-pointer-relative addressing for subroutine parameter passing. The efficient register-to-register architecture of the TMS7000 family has been carried over to the TMS370 family, which avoids the conventional accumulator bottleneck. The complete TMS370 family instruction set is summarized in the TI *TMS370 INSTRUCTION SET SUMMARY* beginning on page 28.

In addition to the interpretation and execution of the user program, the CPU performs the functions of bus protocol generation and interrupt priority arbitration. While the CPU is implemented independent of the memory, input/output, and peripheral modules, it performs the central system control function through communications with these on-chip modules.

The TMS370 family CPU registers accessible to the programmer are shown in Figure 4. The register file consists of 128 general purpose registers, R0 through R127 implemented in on-chip RAM, and is used by the CPU for general purpose 8- and 16-bit source and destination operands, index registers, and indirect addressing. The first two registers, R0 and R1, are also called registers A and B and are used by the CPU as general purpose registers or for implied operands. The program counter (PC) contains the address of the last or top entry on the stack, which is located in the on-chip RAM. The status register (ST) contains four bits that reflect the outcome of the instruction just executed, and two bits that control the masking of the interrupt priority chains.

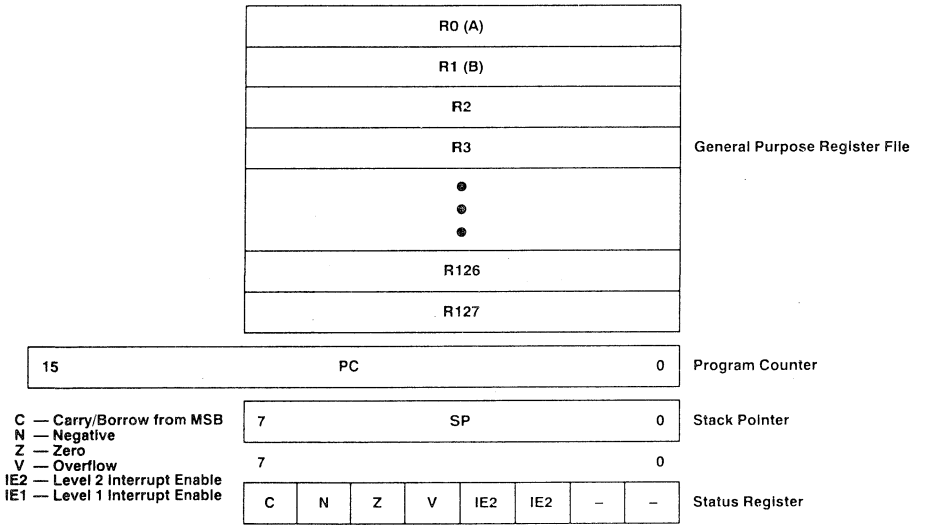


Figure 4. CPU Registers

system resets

The TMS370Cx10 has three possible reset sources: a low input to the $\overline{\text{RESET}}$ pin, a programmable watchdog timer timeout, or a programmable oscillator fault failure. The $\overline{\text{RESET}}$ pin, an input/output pin, initiates TMS370Cx10 hardware initialization and ensures an orderly software startup. A minimum 50-ns low level input initiates the reset sequence. The microcontroller is held in reset until the $\overline{\text{RESET}}$ pin goes inactive (high). If the $\overline{\text{RESET}}$ input signal is low for less than eight system clock cycles, the TMS370Cx10 will hold the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The $\overline{\text{RESET}}$ should be released only after stable oscillation and valid V_{CC}. The $\overline{\text{RESET}}$ pin must be activated by the application at power-up, which can be accomplished by an external input or an RC power-up reset circuit. In addition, the application must activate $\overline{\text{RESET}}$ when V_{CC} goes out of spec to prevent corruption of data and erroneous operation. The $\overline{\text{RESET}}$ pin can be asserted at any time during operation, resulting in an immediate initiation of the reset sequence.

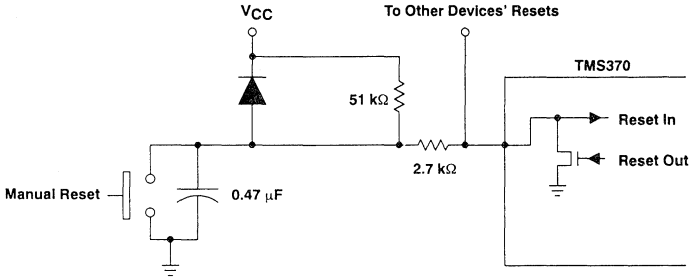


Figure 5. Typical Reset Circuit

The watchdog timer provides system integrity by detecting a program that has become lost or is not executing as expected. A system reset is generated if the watchdog timer is not properly re-initialized by a specific software sequence, or if the re-initialization does not occur before the watchdog timer times out. The watchdog timer timeout initiates the TMS370Cx10 reset sequence and drives the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The Watchdog reset function is enabled by setting WD OVRFL RST ENA bit of T1CTL2 to 1. Once the software enables the watchdog reset function, subsequent writes to the WD OVRFL RST ENA bit are ignored. Watchdog control bits can be initialized only following a power-up reset. The timer section discusses additional information on the watchdog timer and its configurations.

The oscillator fault circuit provides the means to monitor failures of the oscillator input signal (XTAL2/CLKIN). If the oscillator input signal frequency remains above the 90% point of the minimum operating frequency (CLKIN), the oscillator input will not to be activated. However, if the oscillator input is lost or its frequency falls below 20 kHz and the oscillator fault reset has not been disabled, the TMS370Cx10 is reset and the external $\overline{\text{RESET}}$ pin is driven low.

When an oscillator input failure occurs, the internal clocks are stopped and $\overline{\text{RESET}}$ is held active until the oscillator input frequency is greater than 100 kHz typical. The reset action can be disabled by clearing the OSC FLT RST ENA bit (P012.5). Oscillator fault detection can be disabled by setting the OSC FLT DISABLE bit (P012.2). Since operation is unpredictable with slow or intermittent clocks, neither of these actions is recommended. During the HALT mode the oscillator fault circuitry is disabled.

Reset puts the address at 7FFEh and 7FFFh (lsb) into the PC and then clears Registers A and B and the status register. The Stack Pointer is set to 01h during reset. A reset when the device is already running will not affect the other RAM registers.

Control Bit States Following Reset

REGISTER	CONTROL BIT	POWER-UP	WARM RESET
SCCR0	μP/μC MODE	0	0
SCCR0	MC PIN DATA	0	0
SCCR0	COLD START	1	†
SCCR0	OSC FLT FLAG	0	†
T1CTL2	WD OVRFL FLAG	0	†

† Status bit corresponding to active reset source is set to 1.

Interrupts

The TMS370 family software-programmable interrupt structure supports flexible on-chip and external interrupt configurations to meet realtime interrupt-driven application requirements. The hardware interrupt structure incorporates two priority levels as shown in Figure 5. Interrupt level 1 has a higher priority than interrupt level 2. The two priority levels can be independently enabled by the global interrupt enable bits (IE1 and IE2) of the Status Register.

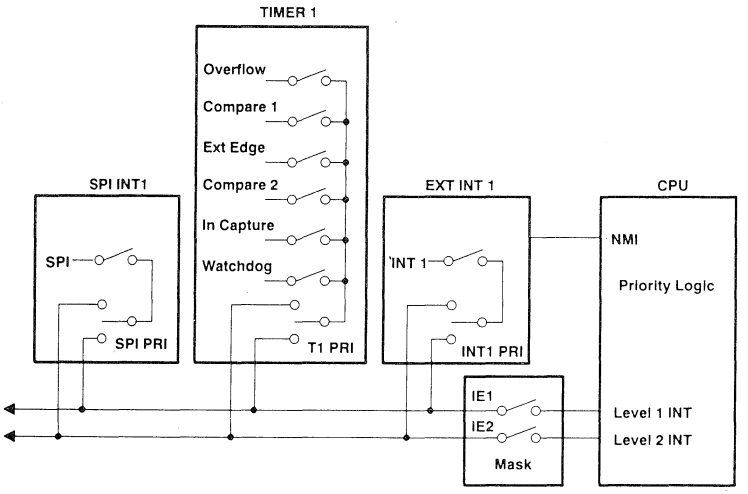


Figure 6. Interrupt Control

Each system interrupt is independently configured on either the high or low priority chain by the application program during system initialization. Within each interrupt chain, the interrupt priority is fixed by the position of the system interrupt. However, since each system interrupt is selectively configured on either the high or low priority interrupt chain, the application program can elevate any system interrupt to the highest priority. Arbitration between the two priority levels is performed within the CPU. Arbitration within each of the priority chains is performed within the peripheral modules. Pending interrupts are serviced upon completion of current instruction execution, depending on their interrupt mask and priority conditions.

The TMS370Cx10 has six hardware system interrupts as shown in the following table. Each system interrupt has a dedicated interrupt vector located in program memory through which control is passed to the interrupt service routines. A system interrupt may have multiple interrupt sources. All of the interrupt sources are

individually maskable by local interrupt enable control bits in the associated peripheral file. Each interrupt source FLAG bit is individually readable for software polling or to determine which interrupt source generated the associated system interrupt.

Two of the system interrupts are generated by on-chip peripheral functions, and three external interrupts are supported. Software configuration of the external interrupts is performed through the INT1, INT2, and INT3 control registers in peripheral file frame 1. Each external interrupt is individually software configurable for input polarity (rising or falling) for ease of system interface. External interrupt INT1 is software configurable as either a maskable or non-maskable interrupt. When INT1 is configured as non-maskable, it cannot be masked by the individual or global mask bits. Recall that the INT1 NMI bit is protected during nonprivileged operation and therefore should be configured during the initialization sequence following reset. To maximize pin flexibility, external interrupts INT2 and INT3 can be software configured as general purpose input/output pins if the interrupt function is not required (INT1 can be similarly configured as an input pin).

Hardware System Interrupts

INTERRUPT SOURCE	INTERRUPT FLAG	SYSTEM INTERRUPT	VECTOR INTERRUPT	PRIORITY§
External Reset Watchdog Overflow Oscillator Fault Detect	COLD START WD OVRFL INT FLAG OSC FLT FLAG	$\overline{\text{RESET}}$†	7FFEh, 7FFFh	1
External INT1	INT1 FLAG	INT1†	7FFCh, 7FFDh	2
External INT2	INT2 FLAG	INT2†	7FFAh, 7FFBh	3
External INT3	INT3 FLAG	INT3†	7FF8h, 7FF9h	4
SPI RX/TX Complete	SPI INT FLAG	SPIINT	7FF6h, 7FF7h	5
Timer 1 Overlfow Timer 1 Compare 1 Timer 1 Compare 2 Timer 1 External Edge Timer 1 Input Capture Watchdog Overflow	T1 OVRFL INT FLAG T1C1 INT FLAG T1C2 INT FLAG T1EDGE INT FLAG T1IC INT FLAG WD OVRFL INT FLAG	T1INT‡	7FF4h, 7FF5h	6

† Releases microcontroller from STANDBY and HALT low power modes.
‡ Releases microcontroller from STANDBY low power mode.
§ Relative priority within an interrupt level.

privileged operation and EEPROM write protection override

The TMS370Cx10 family is designed with significant flexibility to enable the designer to software-configure the system and peripherals to meet the requirements of a broad variety of applications. The non-privileged mode of operation ensures the integrity of the system configuration once defined for an end application. Following a hardware reset, the TMS370Cx10 operates in the privileged mode, where all peripheral file registers have unrestricted read/write access and the application program will configure the system during the initialization sequence following reset. As the last step of system initialization, the PRIVILEGED MODE bit of SCCR3 will be set to 1, entering the non-privileged mode and disabling write operations to specific configuration control bits within the peripheral file. The following system configuration bits are write-protected during the non-privileged mode and must be configured by software prior to exiting the privileged mode:

REGISTER†		CONTROL BIT
NAME	LOCATION	
SCCR0	P010.6	OSC POWER
SCCR2	P012.0	PRIVILEGED MODE
SCCR2	P012.6	PWRDWN/IDLE
SCCR2	P012.7	HALT/STANDBY
SCCR2	P012.1	INT1 NMI
SCCR2	P012.2	OSC FLT DISABLE
SCCR2	P012.5	OSC FLT RST ENA
SPIPRI	P03F.6	SPI PRIORITY
T1PRI	P04F.6	T1 PRIORITY

† Identified by name and bit location within the register

The privileged bits are shown in a **bold typeface** in the following Peripheral File Frame sections.

The EPROM can only be written to when V_{PP} is applied to the MC pin and the VPPS (EPCTL.6) bit is set. When V_{PP} is applied to the MC pin all on-chip EEPROM is in the Write Protect Override (WPO) mode regardless of the state of the VPPS bit. This allows the EPROM to be protected while the EEPROM is in WPO.

low-power operating modes

The STANDBY and HALT low-power modes significantly reduce power consumption by reducing or stopping the activity of the various on-chip peripherals when processing is not required. Each of the low-power modes is entered by executing the IDLE instruction when the POWERDOWN/IDLE bit in SCCR2 has been set to 1. The HALT/STANDBY bit in SCCR2 controls which low-power mode is entered.

In the STANDBY mode (HALT/STANDBY=0), all CPU activity and most peripheral module activity is stopped; however, the oscillator, internal clocks, and Timer 1 remain active. System processing is suspended until a qualified interrupt (hardware $\overline{\text{RESET}}$, external interrupt on INT1, INT2, or INT3, or a Timer 1 interrupt) is detected.

In the HALT mode (HALT/STANDBY=1), the TMS370x10 is placed in its lowest power consumption mode. The oscillator and internal clocks are stopped, causing all internal activity to be halted. System activity is suspended until a qualified interrupt (hardware $\overline{\text{RESET}}$ or external interrupt on INT1, INT2, or INT3) is detected. The powerdown mode selection bits are summarized in the following table:

POWERDOWN CONTROL BITS		MODE SELECTED
PWRDWN/IDLE (SCCR2.6)	HALT/STANDBY (SCCR2.7)	
1	0	Standby
1	1	Halt

The following information is preserved throughout both the STANDBY and HALT modes: RAM (register file), CPU registers (stack pointer, program counter, and status register), I/O pin direction and output data, and status registers of all on-chip peripheral functions. All CPU instruction processing is stopped during the STANDBY and HALT modes, and clocking of the watchdog timer is inhibited.

peripheral file frame 1

Peripheral File Frame 1 contains system configuration and control functions and registers for controlling EPROM and EEPROM programming. The privileged bits are shown in a **bold typeface** in the Peripheral File Frames.

Peripheral Fiie Frame 1: System Configuration and Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1010h	P010	COLD START	**OSC POWER**	**PF AUTO WAIT**	OSC FLT FLAG	MC PIN WPO	MC PIN DATA	—	μP/μC MODE	SCCR0
1011h	P011	—	—	—	**AUTOWAIT DISABLE**	—	**MEMORY DISABLE**	—	—	SCCR1
1012h	P012	**HALT/ STANDBY**	**PWRDWN/ IDLE**	**OSC FLT RST ENA**	**BUS STEST**	**CPU STEST**	**OSC FLT DISABLE**	**INT1 NMI**	**PRIVILEGE DISABLE**	SCCR2
1013h to 1016h	P013 to P016	RESERVED								
1017h	P017	INT1 FLAG	INT1 PIN DATA	—	—	—	INT1 POLARITY	INT1 PRIORITY	INT1 ENABLE	INT1
1018h	P018	INT2 FLAG	INT2 PIN DATA	—	INT2 DATA DIR	INT2 DATA OUT	INT2 POLARITY	INT2 PRIORITY	INT2 ENABLE	INT2
1019h	P019	INT3 FLAG	INT3 PIN DATA	—	INT3 DATA DIR	INT3 DATA OUT	INT3 POLARITY	INT3 PRIORITY	INT3 ENABLE	INT3
101Ah	P01A	BUSY	—	—	—	—	AP	W1W0	EXE	DEECTL
101Bh	P01B	RESERVED								
101Ch	P01C	BUSY	VPPS	—	—	—	—	W0	EXE	EPCTL

peripheral file frame 2

Peripheral File Frame 2 contains the digital I/O pin configuration and control registers. The following figure details the specific addresses, registers, and control bits within this Peripheral File Frame.

Peripheral File Frame 2: Digital Port Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1020h	P020	RESERVED								APORT1
1021h	P021	Port A Control Register 2 (must be 0)								APORT2
1022h	P022	Port A Data								ADATA
1023h	P023	Port A Direction								ADIR
1024h to 102Bh	P024 to P02B	RESERVED								
102Ch	P02C	Port D Control Register 1 (must be 0)					X	X	X	DPORT1
102Dh	P02D	Port D Control Register 2 (must be 0)†					X	X	X	DPORT2
102Eh	P02E	Port D Data					X	X	X	DDATA
102Fh	P02F	Port D Direction					X	X	X	DDIR

† To configure pin D3 as CLKOUT, set Port D Control Register 2 equal to 08h.

Port Configuration Registers Set-up

PORT	PIN	abcd 00x0	abcd 00q1
A	0 — 7	Data In	Out q
D	3 — 7	Data In	Out q

a = Port x Control Register 1
b = Port x Control Register 2
c = Data
d = Direction

NOTES: 2. Each bit controls the corresponding pin; for example, bit 6 controls Port pin 6. Each pin is individually configurable.
3. Only register combination 00xx is defined for TMS370Cx10.

programmable timers

The programmable timer module of the TMS370Cx10 provides the designer with the enhanced timer resources required to perform realtime system control. The Timer 1 module contains the general-purpose T1 and the watchdog timer WD. The two independent 16-bit timers, T1 and WD, allow program selection of input clock sources (realtime, external event, or pulse accumulate) with multiple 16-bit registers (input capture and compare) for special timer function control. These timers provide the capabilities for:

System Requirements	Timer Resource
Realtime System Control	Interval Timers with Interrupts
Input Pulse Width Measurement	Pulse Accumulate or Input Capture Functions
External Event Synchronization	Event Counter Function
Timer Output Control	Compare Function
Pulse-Width Modulated Output Control	PWM Output Function
System Integrity	Watchdog Function

timer 1 module

The timer 1 module consists of three major blocks:

1. Prescaler/Clock Source, which determines the independent clock sources for the general purpose timer and the watchdog timer.
2. 16-bit General Purpose Timer, T1, which provides the event count, input capture, and compare functions.
3. 16-bit Watchdog Timer, which may be software programmed as an event counter, pulse accumulator, or interval timer if the watchdog function is not desired.

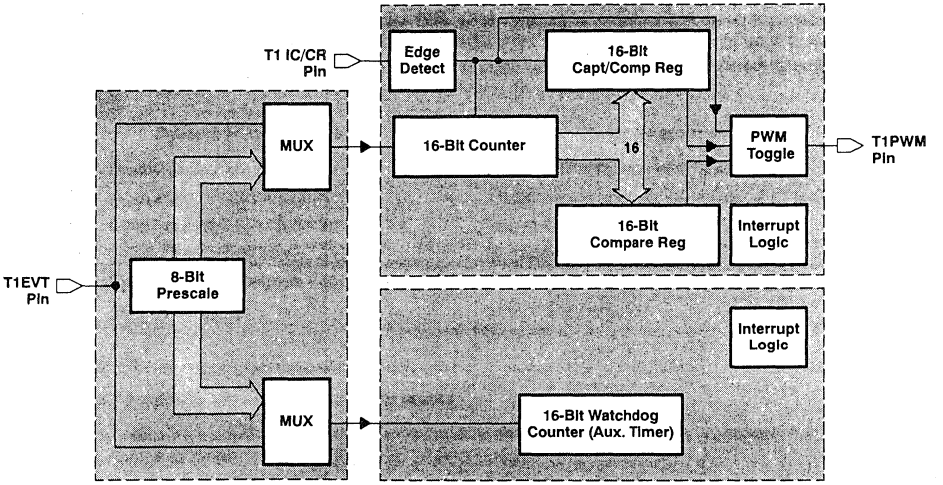


Figure 7. Timer 1 Module Block Diagram

timer 1 module prescaler/clock source

The clock source inputs for the general purpose timer and the watchdog timer are independently configured by the T1 and WD INPUT SELECT control bits of the T1CTL1 cotnrol register. The WD INPUT SELECT control bits cannot be changed after entering the watchdog mode (WD RST ENA = 1). Eight possible clock sources are programmable for each counter.

T1 INPUT				WD INPUT		
SELECT 2	SELECT 1	SELECT 0		SELECT 2	SELECT 1	SELECT 0
0	0	0	System Clock	0	0	0
0	0	1	Pulse Accumulate	0	0	1
0	1	0	Event Input	0	1	0
0	1	1	No Clock Input	0	1	1
1	0	0	System Clock/4	1	0	0
1	0	1	System Clock/16	1	0	1
1	1	0	System Clock/64	1	1	0
1	1	1	System Clock/256	1	1	1

For realtime control applications, both the general-purpose timer and the watchdog timer are independently programmable from 15 to 24 bits in length. The 24-bit prescaler/timer generates overflow rates ranging from 13.1 ms with 200 ns timer resolution to 3.35 seconds with 51.2 μs timer resolution (external clock = 20 MHz).

In the **Event Counter mode**, an external high-to-low transition on the T1EVT pin is used to provide the clock for the internal timers. As shown in Figure 8, the T1EVT input provides the timer clock and is not routed through the prescaler. The T1EVT external clock frequency may not exceed the system clock frequency divided by 2. The general-purpose timer and the watchdog timer are programmable as 16-bit event counters.

In the **Pulse Accumulate mode**, an external input on the T1EVT pin is used to gate the internal system clock to the internal timers. While T1EVT input is logic one (high), the timers will be clocked at the system clock rate and will accumulate system clock pulses until the T1EVT pin returns to logic zero. Both the general purpose timer and the watchdog timer are programmable as 16-bit pulse accumulators.

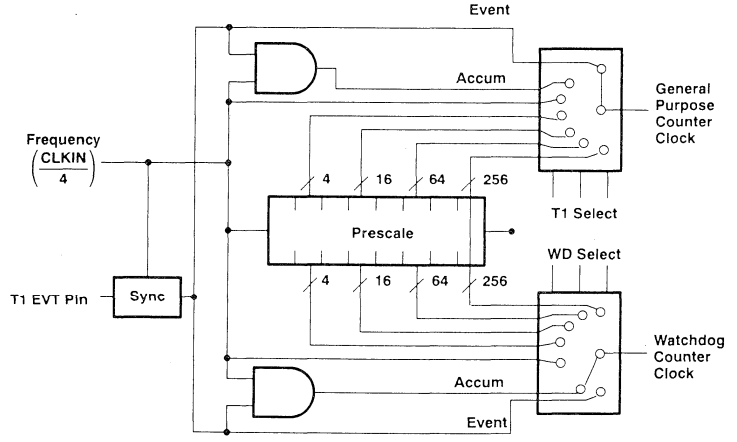


Figure 8. Timer 1 Counter Prescaler

timer 1 general purpose timer

The 16-bit general purpose timer, T1, is composed of a 16-bit resettable counter, a 16-bit compare register and associated compare logic, and a 16-bit register that functions as a capture register in one mode and a compare register in the other mode. The T1 MODE bit selects whether T1 operates in the Capture/Compare mode or the Dual Compare mode.

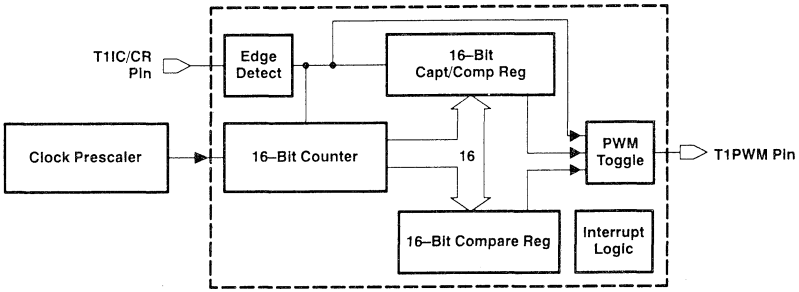


Figure 9. Timer 1 — General Purpose Timer

The counter is a free-running, 16-bit up-counter, clocked by the output of the Prescaler/Clock source. During initialization the counter is loaded with 0000h and begins its up-count. If the counter is not reset before reaching FFFFh, the counter will roll over to 0000h and continue counting. Upon counter roll-over, the T1 OVRFL INT FLAG is set to 1, and a timer interrupt is generated if the T1 OVRFL INT ENA bit is set to 1.

The counter may be reset to 0000h during counting by either; 1) writing a 1 to the T1 SW RESET bit; 2) a compare equal condition from the dedicated T1 compare function; or 3) an external pulse on the T1 IC/CR pin (Dual Compare mode). The designer may select via software (T1 EDGE POLARITY bit) which external transition, low-to-high or high-to-low, on the T1IC/CR pin will cause the counter to be reset.

Special circuitry prevents the 16-bit registers, including the Counter, Compare, or Capture registers, from changing in the middle of a 16-bit read or write operation. When reading a 16-bit register, read the LSB first and then read the MSB. When writing to a 16-bit register, write the MSB first and then write the LSB. The register value will not change between reading or writing the bytes when done in this order.

The timer 1 module has three I/O pins used for the functions shown in the following table. Any of these three pins not used in a timer application may be individually configured as general purpose digital I/O pins by the timer 1 module port control registers (T1PC1 and T1PC2).

Timer 1 Module I/O Pin Functions

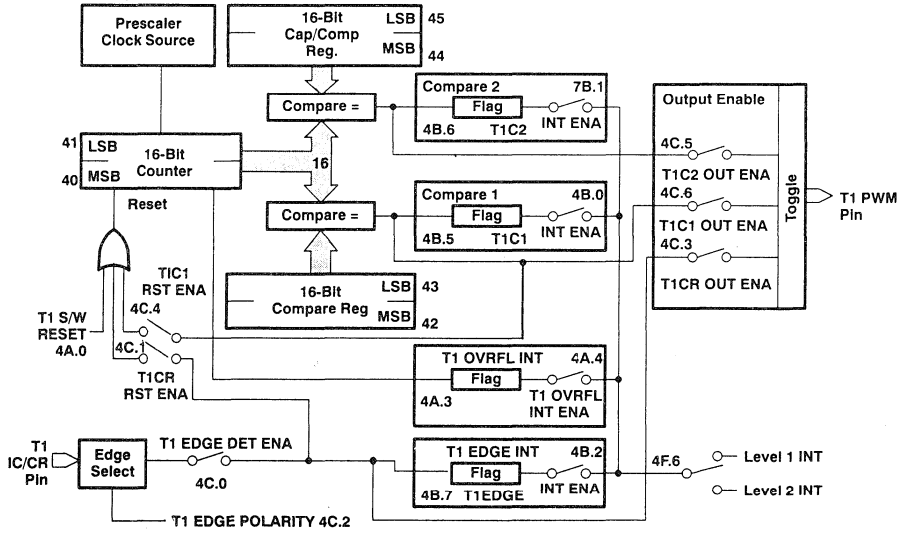
PIN	DUAL COMPARE MODE	CAPTURE/COMPARE MODE
T1IC/CR	Counter Reset input	Input Capture input
T1PWM	PWM output	Compare output
T1EVT	External Event input or Pulse Accumulate input	External Event input or Pulse Accumulate input

The **Dual Compare mode** (T1 MODE = 0) provides two compare registers, an external resettable counter, and a timer output pin. These allow the timer to act as an interval timer, a PWM output, simple output toggle, or many other timer functions. The Dual Compare mode as shown in Figure 10 continuously compares the contents of the two compare registers to the current value of the 16-bit counter. If a timer compare register equals the counter, the circuit sets the associated interrupt flag to 1 and toggles the T1PWM output pin if enabled, and/or

generates a Timer 1 interrupt. An output compare equal condition from the dedicated compare register can also initiate a counter reset. A programmable interval timer function, selected by using the compare equal condition to generate a system interrupt and the counter reset function, generates a periodic interrupt.

Either compare function may be used to toggle the T1PWM output pin when a timer compare equal occurs, while the other compare function may be used for another system timing function. Using both compare functions to control the T1PWM pin allows direct PWM generation with minimal CPU software overhead. In typical PWM applications, the compare register is written with the periodic interval and is configured to allow counter reset on compare equal, and the Capture/Compare register is written with the pulse width to be generated within that interval. The program pulse width may be changed by the application program during the timer operation to alter the PWM output. For high-speed control applications, a minimum pulse width of 200 ns and a period as low as 400 ns can be maintained when using a clock of 20 MHz.

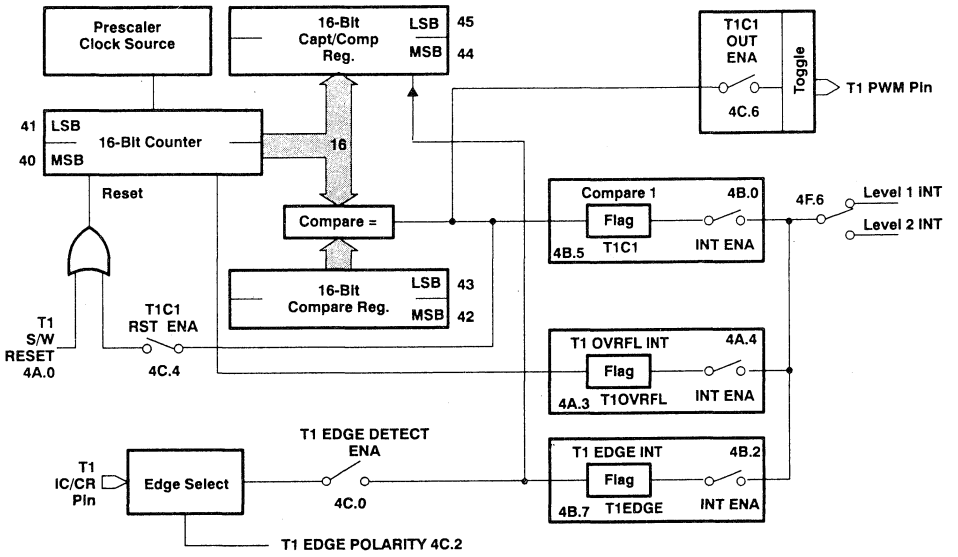
In addition, a PWM output that is initiated by a transition on an external pin is provided by the timer hardware to support time-critical control applications. Typically, in these applications an external input (T1IC/CR) is used to reset the counter, generate a timer interrupt, and toggle the T1PWM pin to start the PWM output. The compare function will then toggle the output after the programmed pulse width has elapsed. The input edge detect function is enabled under program control by the T1CR DET ENA bit, and upon the next occurrence of the selected edge transition, the T1EDGE INT FLAG bit is set to 1, a timer interrupt is generated (if T1EDGE INT ENA = 1), and the T1PWM output pin is toggled (if T1CR OUT ENA = 1). Selection of the active input transition is under control of T1EDGE POLARITY. In the Dual Compare mode, the edge detect function must be re-enabled after each valid edge detect.



NOTE 4: The numbers on the diagram, such as 4C.0, identify the register and the bit in the peripheral frame. For example, the actual address of 4C.0 is 104Ch, bit 0, in the T1CTL4 register.

Figure 10. Timer 1 — Dual Compare Mode

In the **Capture/Compare mode** (T1 MODE = 1), T1 is configured to provide one input capture register for external timing and pulse width measurement, and one compare register for use as a programmable interval timer. The compare register in this mode functions the same as in the Dual Compare mode described above, including the ability to toggle the PWM pin. The capture/compare register functions in this mode as a 16-bit input capture register, as shown in Figure 11. On the occurrence of a valid input on the T1IC/CR pin, the current counter value is loaded into the 16-bit input capture register, the T1 EDGE INT FLAG is set to 1, and a timer interrupt is generated (if TI EDGE INT ENA = 1). The input detect function is enabled by the T1 EDGE DET ENA bit, with T1 EDGE POLARITY selecting the active input transition. In the Capture/Compare mode, the edge detect function, once enabled, remains enabled following a valid edge detect.



NOTE 4: The numbers on the diagram, such as 4C.0, identify the register and the bit in the peripheral frame. For example, the actual address of 4C.0 is 104Ch, bit0, in the T1CTL4 register.

Figure 11. Timer 1 — Capture/Compare Mode

timer 1 module watchdog timer

The watchdog timer, contained in the timer 1 module, is a free-running 16-bit resettable up-counter clocked by the output of the Prescaler/Clock Source. The timer is software configured as either a watchdog timer to protect against system software failures and corruption, or as a general purpose timer if the watchdog function is not desired. The 16-bit up-counter is programmable (via the WD OVRFL TAP SEL bit) to set the initial count at either 0000h or 8000h. The current value of the watchdog timer may be read at any time during its operation.

In the **Watchdog mode** (WD OVRFL RST ENA = 1), the timer will generate a system reset if the timer is re-initialized by an incorrect value or if the counter overflows. The required re-initialization frequency is determined by the system clock frequency, the prescaler/clock source selected, and whether the WD OVRFL TAP SEL bit is set for 15 or 16 bit counter rollover. With a clock = 20 MHz, the watchdog timer overflow rates range form 6.55 ms to 3.35 seconds. These values are selected prior to entering the watchdog mode because once the software enables the watchdog reset function (WD OVRFL RST ENA set to 1), subsequent writes to these control bits are ignored. Writes to these watchdog control bits can occur only following a powerup reset, which enhances watchdog timer system integrity.

The watchdog timer is re-initialized by writing a predefined value to the watchdog reset key (WDRST) located in the peripheral file. The proper reset key alternates between 55h and AAh, beginning with 55h following the enable of the watchdog reset function. Writes of the correct value must occur prior to the timer overflow period. A write of any value other than the correct predefined value to the watchdog reset key will be interpreted as a lost program and a system reset will be initiated. A watchdog timer overflow or incorrect reset key will set the WD OVRFL INT FLAG bit to 1 and may be interrogated by the program following system reset to determine the source of the reset.

In the **Non-watchdog mode** (WD OVRFL RST ENA = 0), the watchdog timer may be used as an event counter, pulse accumulator, or as an interval timer. In this mode, the system reset function is disabled. The watchdog counter is re-initialized by writing any value to the watchdog reset key (WDRST). In real-time control applications, the timer overflow rates are determined by the system clock frequency, the prescaler/clock source value selected, and the value of the WD OVRFL TAP SEL bit. If the WD counter is not reset before overflowing, the counter will roll over to either 0000h or 8000h, as determined by the WD OVRFL TAP SEL bit, and continue counting. Upon counter overflow, the WD OVRFL INT FLAG is set to 1 and a timer interrupt is generated if the WD OVRFL INT ENA bit set to 1. Alternately, an external input on the T1EVT pin may be used with the watchdog timer to provide an additional 16-bit event counter or pulse accumulator.

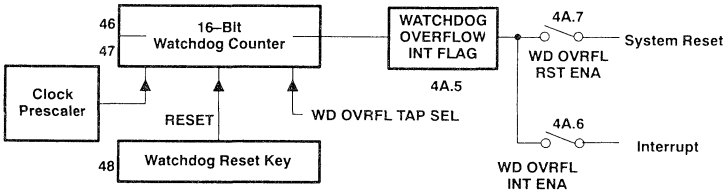


Figure 12. Watchdog/General Purpose Timer

Peripheral File Frame 4: Timer 1 Module Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1040h	P040	Bit 15			Counter MSB				Bit 8	T1CNTR
1041h	P041	Bit 7			Counter LSB				Bit 0	
1042h	P042	Bit 15			Compare Register MSB				Bit 8	T1C
1043h	P043	Bit 7			Compare Register LSB				Bit 0	
1044h	P044	Bit 15			Capture/Compare Register MSB				Bit 8	T1CC
1045h	P045	Bit 7			Capture/Compare Register LSB				Bit 0	
1046h	P046	Bit 15			Watchdog Counter MSB				Bit 8	WDCNTR
1047h	P047	Bit 7			Watchdog Counter LSB				Bit 0	
1048h	P048	Watchdog Reset Key								WDRST
1049h	P049	WD OVRFL TAP SEL†	WD INPUT SELECT 2†	WD INPUT SELECT 1†	WD INPUT SELECT 0†	—	TI INPUT SELECT 2	TI INPUT SELECT 1	TI INPUT SELECT 0	T1CTL1
104Ah	P04A	WD OVRFL RST ENA†	WD OVRFL INT ENA	WD OVRFL INT FLAG	TI OVRFL INT ENA	T1 OVRFL INT FLAG	—	—	T1 SW RESET	T1CTL2
Mode: Dual Compare										
104Bh	P04B	T1 EDGE INT FLAG	T1C2 INT FLAG	T1C1 INT FLAG	—	—	T1 EDGE INT ENA	T1C2 INT ENA	T1C1 INT ENA	T1CTL3
104Ch	P04C	T1 MODE = 0	T1C1 OUT ENA	T1C2 OUT ENA	T1C1 RST ENA	T1CR OUT ENA	T1EDGE POLARITY	T1CR RST ENA	T1EDGE DET ENA	T1CTL4
Mode: Capture/Compare										
104Bh	P04B	T1EDGE INT FLAG	—	T1C1 INT FLAG	—	—	T1 EDGE INT ENA	—	T1C1 INT ENA	T1CTL3
104Ch	P04C	T1 MODE = 1	T1C1 OUT ENA	—	T1C1 RST ENA	—	T1EDGE POLARITY	—	T1EDGE DET ENA	T1CTL4
104Dh	P04D	—	—	—	—	T1EVT DATA IN	T1EVT DATA OUT	T1EVT FUNCTION	T1EVT DATA DIR	T1PC1
104Eh	P04E	T1PWM DATA IN	T1PWM DATA OUT	T1PWM FUNCTION	T1PWM DATA DIR	T1IC/CR DATA IN	T1IC/CR DATA OUT	T1IC/CR FUNCTION	T1IC/CR DATA DIR	T1PC2
104Fh	P04F	**T1 STEST**	**T1 PRIORITY**	—	—	—	—	—	—	T1PRI

† Subsequent writes to these control bits are ignored after the WD OVRFL RST ENA bit is set to 1 (see **timer1 module watchdog timer)** and can be initialized only following a **powerup reset**.

The formulas in Figure 13 show the calculations for the resulting time, given values in the compare registers T1C and T1CC.

$$\text{time} = \left(\frac{4}{\text{CLKIN}}\right) (\text{prescale}) \ (\text{compare} + 1)$$

or

$$\text{time} = t_c \ (\text{prescale}) \ (\text{compare} + 1)$$

Figure 13. Timer 1 Compare Register Formulas

serial peripheral interface (SPI)

The Serial Peripheral Interface (SPI) is a high-speed synchronous serial I/O port that allows a serial bit stream of programmed length (one to eight data bits) to be shifted into and/or out of the device at a programmed bit transfer rate. The SPI is normally used for communications between the microcontroller and external peripherals or another microcontroller. Typical applications include external I/O or peripheral expansion using devices such as shift registers, display drivers, A/D converters, etc. Multiprocessor communications are also supported by the master/slave operation of the SPI.

Three I/O pins are associated with the SPI.These include the SPI slave-in master-out (SPISIMO), SPI slave-out master-in (SPISOMI), and SPI serial serial clock (SPICLK). These I/O pins can be configured for three-wire full-duplex transmit/receive or two-wire receive or transmit only. Any of these three pins not used in an SPI application may be individually configured as general purpose digital I/O pins controlled by SPIPC1 and SPIPC2.

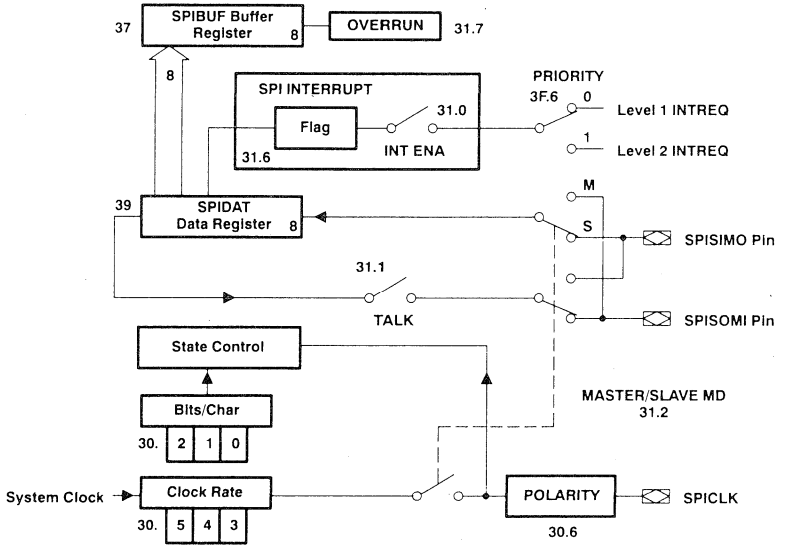


Figure 14. SPI Block Diagram

A variety of multiprocessor configurations can be supported, ranging from single master with multiple slaves to multi-master systems. General purpose I/O pins can be used to implement the slave enables and multi-master hardware handshakes between microcontrollers in the network.

The MASTER/SLAVE bit of the SPICTL control register determines if the SPI operates in the master or slave mode. Master or slave data transmission can be disabled by writing a zero to the TALK bit of the SPICTL control register, forming a two-wire receive-only network (SPICLK and data in).

In the **Master mode** (MASTER/SLAVE = 1), the SPI provides the serial clock on the SPICLK pin for the entire serial communications network. The SPICCR register (SPI BIT RATE2, RATE1, RATE0) determines the bit transfer rate for the network, both transmit and receive. For any specific system clock frequency, there are eight data transfer rates that can be selected by these control bits. The data transfer rate is defined by selecting a one-of-eight divide-by of the system clock frequency (divide-by-2, -4, -8, -16, -32, -64, -128, and -256).

$$\text{SPI Baud Rate} = \frac{\text{CLKIN}}{8 \times 2^{b}}$$

where b = bit rate in SPICCR bit 3, 4, 5 (range 0-7).

Data written to the SPIDAT register initiates data transmission on the SPISIMO pin, MSB of data transmitted first.Simultaneously, received data is shifted in the SPISOMI pin into the SPIDAT register, and upon completion of transmitting the selected number of bits, the data is transferred to the SPIBUF (double buffered receiver) for reading by the CPU to permit new transactions to take place. Data is shifted into the SPI the most significant bit first, there it is stored right-justified in the SPIBUF. To receive a character when operating as a master, data must be written to the SPIDAT to initiate the transaction. When the specified number of data bits have been shifted into or out of the SPIDAT register, the SPI INT FLAG bit is set and if the SPI INT ENA bit is set to one, an Interrupt is asserted.

In the **Slave mode** (MASTER/SLAVE = 0), data shifts out on the SPISOMI pin and in on the SPISIMO pin. The SPICLK pin is used as the input for the serial shift clock, which is supplied from the external network master. The transfer rate is defined by the input clock on the SPICLK pin, which is supplied from the network master. The SPICLK input frequency should be no greater than system clock frequency divided by eight.

Data written to the SPIDAT register will be transmitted to the network when the SPICLK is received from the network master. To receive data, the SPI waits for the network master to send SPICLK and then shifts the data on the SPISIMO pin into the SPIDAT register. If data is to be transferred by the slave simultaneously, then it must be written to the SPIDAT register prior to the beginning of SPICLK.

Compatibility with the broadest range of existing peripheral devices is provided by the SPI through its software programmable transmit/receive character length, bit transfer rate, and clock polarity. A character length from one to eight data bits is selected by writing to the SPICCR control register (SPI CHAR2, CHAR1, and CHAR0) to specifically match the peripheral's data length requirements, thereby not requiring the overhead of data bit padding during communications. Applications requiring more than eight bits of serial data use multiple back-to-back SPI operations.

External peripherals enable output data on either the rising or the falling edge of the serial clock, while latching incoming data on the opposite edge. The SPI supports data transfer using either of these approaches. The CLOCK POLARITY bit controls the steady-state or at-rest condition of the SPICLK signal. This bit affects both master and slave modes of operation. When CLOCK POLARITY is set to 1, the at-rest level of SPICLK is a logic one (high). Data is enabled at the output on the falling edge of SPICLK, and data is latched by the network master and slaves on the rising edge of SPICLK. When CLOCK POLARITY is set to zero, the at-rest level of SPICLK is a logic zero (low). Data is enabled for output on the rising edge of SPICLK, and data is latched by the network master on the falling edge of SPICLK.

Peripheral File Frame 3: Serial Peripheral Interface (SPI) Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1030h	P030	SPI SW RESET	CLOCK POLARITY	SPI BIT RATE2	SPI BIT RATE1	SPI BIT RATE 0	SPI CHAR2	SPI CHAR1	SPI CHAR0	SPICCR
1031h	P031	RECEIVER OVERRUN	SPI INT FLAG	—	—	—	MASTER/ SLAVE	TALK	SPI INT ENA	SPICTL
1032h to 1036h	P032 to P036	Reserved								
1037h	P037	SPI Receive Data Buffer Register								SPIBUF
1038h	P038	Reserved								
1039h	P039	SPI Serial Data Register								SPIDAT
103Ah to 103Dh	P03A to P03C	Reserved								
103Dh	P03D	—	—	—	—	SPICLK DATA IN	SPICLK DATA OUT	SPICLK FUNCTION	SPICLK DATA DIR	SPIPC1
103Eh	P03E	SPISIMO DATA IN	SPISIMO DATA OUT	SPISIMO FUNCTION	SPISIMO DATA DIR	SPISOMI DATA IN	SPISOMI DATA OUT	SPISOMI FUNCTION	SPISOMI DATA DIR	SPIPC2
103Fh	P03F	**SPI STEST**	**SPI PRIORITY**	**SPI ESPEN**	—	—	—	—	—	SPIPRI

instruction set

The TMS370 family instruction set consists of 73 instructions that control input, output, data manipulations, data comparisons, and program flow. The instruction set is supported with 14 addresssing modes to provide the flexibility to optimize programs to the user's applications. For example, the MOV instruction has 27 operand combinations supported by its addressing modes.

ADDRESSING MODE	EXAMPLE		OPERATION
GENERAL:			
Implied	LDSP		(B) → (SP)
Register	MOV	R5,R4	(0005) → (0004)
Peripheral	MOV	P025,A	(1025) → A
Immediate	ADD	#123,R3	123 + (03) → (03)
PC Relative	JMP	offset	PCN + offset → (PC)
Stack Pointer Relative	MOV	2(SP),(A)	(2 + (SP)) → (A)
EXTENDED:			
Absolute Direct	MOV	A,1234	(A) → (1234)
Absolute Indexed	MOV	1234(B),A	(1234) + (B) → (A)
Absolute Indirect	MOV	@R4,A	((R3:R4)) → (A)
Absolute Offset Indirect	MOV	12(R4),A	(12 + (R3:R4)) → (A)
Relative Direct	JMPL	1234	PCN + 1234 → (PC)
Relative Indexed	JMPL	1234(B)	PCN + 1234 + (B) → (PC)
Relative Indirect	JMPL	@R4	PCN + (R3:R4) → (PC)
Relative Offset Indirect	JMPL	12(R4)	PCN + 12 + (R3:R4) → (PC)

PCN = 16-bit address of next instruction.
(x) = Contents of memory at address x.
((x)) = Contents of memory location designated by contents at address x.

The CPU controls instruction execution by executing microinstructions from a dedicated control memory at a rate of one microinstruction per internal system clock cycle, t_c. The number of system clock cycles required to execute one assembly language instruction varies depending on the instruction complexity, operand addressing mode, and number of wait states. Instruction execution times are stated in terms of the number of internal system clock cycles per instruction. Instruction execution times vary from 5 to 63 internal system clock cycles, with most instructions requiring less than 10 cycles to complete.

Similarly, the number of bytes of program memory required to store an instruction will vary with instruction complexity and addressing mode. TMS370 instructions require from one to five bytes of program memory space, with most instructions occupying one or two bytes.

The TMS370 INSTRUCTION SET SUMMARY, beginning on page 28, shows the instruction set, the addressing modes, the program memory byte length, and the execution cycle count for each instruction. The Addressing Mode entries are in the format of BYTE LENGTH/CYCLE COUNT. The following symbols and abbreviations are used:

SYMBOL	DEFINITION	SYMBOL	DEFINITION
s	Source Operand	d/D	Destination Operand (8-bit/16-bit)
A	Register A or R0 in Register File	B	Register B or R1 in Register File
Rs	Source Register in Register File	Rd	Destination Register in Register File
Ps	Source Register in Peripheral File	Pd	Destination Register in Peripheral File
Rps	Source Register Pair (Rn, Rn-1)	Rpd	Destination Register Pair (Rn, Rn-1)
Rp	General Purpose Register Pair	label	16-bit Label
iop8	8-bit Immediate Operand	iop16	16-bit Immediate Operand
off8	8-bit Signed Offset (label – PCN)	off16	16-bit Signed Offset
PC	Program Counter	PCN	16-bit Address of Next Instruction
SP	Stack Pointer	ST	Status Register
#	Immediate Operand	@	Indirect Addressing Operand
C	Status Register Carry Bit	XADDR	16-bit Address
( )	Contents of	→	Is Assigned to

TMS370 Instruction Set Summary

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)	OTHER	
ADC	B, __	1/8										Add with Carry
	Rs, __	2/7	2/7	3/9								(s) + (d) + (C) → (d)
	#iop8, __	2/6	2/6	3/8								
ADD	B, __	1/8										Add
	Rs, __	2/7	2/7	3/9								(s) + (d) → (d)
	#iop8, __	2/6	2/6	3/8								
AND	A, __				2/9							And
	B, __	1/8			2/9							(s) AND (d) → (d)
	Rs, __	2/7	2/7	3/9								
	#iop8, __	2/6	2/6	3/8	3/10							
BR						3/9	2/8	3/11	4/15			Branch; D → XADDR
BTJO†	A, __,off8				3/10							Bit Test and Jump If One
	B, __,off8	2/10			3/10							If (s) AND (d) ≠ 0 then
	Rs, __,off8	3/9	3/9	4/11								PCN + offset → (PC)
	#iop8, __,off8	3/8	3/8	4/10	4/11							
BTJZ†	A, __,off8				3/10							Bit Test and Jump If Zero
	B, __,off8	2/10			3/10							If (s).AND.(not d) ≠ 0 then
	Rs, __,off8	3/9	3/9	4/11								PCN + offset → (PC)
	#iop8, __,off8	3/8	3/8	4/10	4/11							
CALL	__					3/13	2/12	3/15	4/19			Call; Push PCN, D → XADDR
CALLR	__					3/15	2/14	3/17	4/21			Call Relative
												Push PCN, PCN + XADDR → (PC)
CLR	__	1/8	1/8	2/6								Clear; 0 → (d)
CLRC											1/19	Clear Carry; 0 → (C)
CMP	__,A					3/11	2/10	3/13	4/17	2/8		Compare
	B, __	1/8										(d) – (s) computed and
	Rs, __	2/7	2/7	3/9								Status Register flags set
	#iop8, __	2/6	2/6	3/8								
CMPBIT	__			3/8	3/10							Complement Bit (One's complement)
CMPL	__	1/8	1/8	2/6								Two's complement;
												0100h – (s) → (d)
DAC	B, __	1/10										Decimal Add with Carry
	Rs, __	2/9	2/9	3/11								(s) + (d) + (C) → (d) (BCD)
	#iop8, __	2/8	2/8	3/10								
DEC	__	1/8	1/8	2/6								Decrement; (d) – 1 → (d)
DINT											2/6	Disable Interrupt; 00 → (ST)
DIV	Rs, __	3/47-63‡										Integer Divide; 16 by 8-bit
												A:B/Rs → A(=quo), B(=rem)
												# cycles depends on operands
DJNZ†	__,off8	2/10	2/10	3/8								Decrement and Jump If Not 0
												(d) – 1 → (d); if (d) ≠ 0
												then PCN + offset → (PC)

† Add 2 to cycle count if jump is taken.
‡ Actual number of cycles is 14 if the quotient is greater than 8 bits (overflow condition).

TMS370 Instruction Set Summary (continued)

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)	OTHER	
DSB	B, __	1/10										Decimal Subtract with Borrow
	RS, __	2/8	2/9	3/11								(d) – (s) – 1 + (C) → (d) (BCD)
	#iop8, __	2/8	2/8	3/10								
EINT											2/6	Enable Interrupts; 0Ch → (ST)
EINTH											2/6	EINT High Priority; 04h → (ST)
EINTL											2/6	EINT Low Priority; 08h → (ST)
IDLE											1/6	Idle Until Interrupt, Low Power entry
INC	__	1/8	1/8	2/6								Increment; (d) + 1 → (d)
INCW	#off8, __			3/11								(Rp) + offset → (Rp)
INV	__	1/8	1/8	2/6								Invert; NOT (d) → (d)
JMP	__										2/7	Jump; PCN + offset8 → (PC)
JMPL	__					3/9	2/8	3/11	4/15			Jump; PCN + XADDR → (PC)
Jcnd†											2/5	Jump Conditional
JN											2/5	Negative
JZ											2/5	Zero
JC											2/5	Carry
JP											2/5	Positive
JPZ											2/5	Positive or Zero
JNZ											2/5	Negative or Zero
JNC											2/5	No Carry
JV											2/5	Overflow, signed
JNV											2/5	No Overlfow, signed
JGE											2/5	Greater Than or Equal, signed
JL											2/5	Less Than, signed
JG											2/5	Greater Than, Signed
JLE											2/5	Less Than or Equal, signed
JLO											2/5	Lower Value
JHS											2/5	Higher or Same
JBIT0†	__			4/10	4/11							Jump If Bit = 0
JBIT1†	__			4/10	4/11							Jump If Bit = 1
LDSP											1/7	Load Stack Pointer; (B) → (SP)
LDST	#iop8										2/6	Load ST Register; (s) → (SP)
MOV	A, __		1/9	2/7	2/8	3/10	2/9	3/12	4/16	2/7		Move; (s) → (d)
	__, A		1/8	2/7	2/8	3/10	2/9	3/12	4/16	2/7		
	B, __	1/8		2/7	2/8							
	Rs, __	2/7	2/7	3/9	3/10							
	Ps, __	2/8	2/8	3/10								
	#iop8, __	2/6	2/6	3/8	3/9							
MOVW	Rps, __			3/12								Move Word; 16-bit operands
	#iop16, __			4/13								(s) → (d)
	#iop16(B), __			4/15								
	#off8(Rp), __			5/19								

† Add 2 to cycle count if jump is taken.

TMS370 Instruction Set Summary (concluded)

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)	OTHER	
MPY	B, _	1/47										Multiply
	RS, _	2/46	2/46	3/48								(s) × (d) → (A:B)
	#iop8, _	2/45	2/45	3/47								A = MSB, B = LSB
NOP											1/7	NOP; (PC) +1 → (PC)
OR	A, _				2/9							OR
	B, _	1/8			2/9							(s) OR (d) → (d)
	Rs, _	2/7	2/7	3/9								
	#iop8, _	2/6	2/6	3/8	3/10							
POP	_	1/9	1/9	2/7							1/8	Pop Top of Stack
												((SP)) → (d); (SP) – 1 → (SP)
PUSH	_	1/9	1/9	2/7							1/8	Push onto Stack
												(SP) + 1 → (SP); (s) → ((SP))
RL	_	1/8	1/8	2/6								Rotate Left
RLC	_	1/8	1/8	2/6								Rotate Left Through Carry
RR	_	1/8	1/8	2/6								Rotate Right
RRC	_	1/8	1/8	2/6								Rotate Right Through Carry
RTI											1/12	Return from Interrupt
												Pop PC, Pop ST
RTS											1/9	Return from Subroutine, Pop PC
SBIT0	_			3/8	3/10							Set Bit to 0
SBIT1	_			3/8	3/8							Set BIT to 1
SETC											17	Set Carry; A0h → (ST)
SSB	B, _	1/8										Subtract with Borrow
	Rs, _	2/7	2/7	3/9								(d) – (s) – 1 + (C) → (d)
	#iop8, _	2/6	2/6	3/8								
STSP											1/8	Store Stack Pointer; (SP) → (B)
SUB	B, _	1/8										Subtract
	Rs, _	2/7	2/7	3/9								(d) – (s) → (d)
	#iop8, _	2/6	2/6	3/8								
SWAP	_	1/11	1/11	2/9								Swap Nibbles
												s(7-4,3-0) → d(3-0,7-4)
TRAPn											1/14	Trap to Subroutine; Push PCN;
												Vector n → (PC)
TST	_	1/9	1/10									Test; Set flags from register
XCHB	_	1/10	1/10	2/8								Exchange B; (B) ↔ (d)
XOR	A, _				2/9							Exclusive OR
	B, _	1/8			2/9							(s) XOR (d) → (d)
	Rs, _	2/7	2/7									
	#iop8, _	2/6	2/6	3/8	3/10							

TMS370 Family OPCODE/Instruction Map

SECOND NIBBLE	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	JMP ra 2/7							INCW #n,Rd 3/11	MOV Pa,A 2/8			CLRC TST A 1/9	MOV A,B 1/9	MOV A,Rd 2/7	TRAP 15 1/14	LDST n 2/6
1	JN ra 2.5		MOV A,Pd 2/8			MOV B,Pd 2/8		MOV Rs,Pd 3/10		MOV Ps,Rd 3/10				MOV B,Rd 2/7	TRAP 14 1/14	MOV n(SP),A 2/7
2	JZ ra 2/5	MOV Rs,A 2/7	MOF #n,A 2/6	MOV Rs,B 2/7	MOV Rs,Rd 3/9	MOV #n,B 2/6	MOV B,A 1/8	MOV #n,Rd 3/8			MOV Ps,Rd 3/10	DEC A 1/8	DEC B 1/8	DEC Rn 2/8	TRAP 13 1/14	MOV A,n(SP) 2/7
3	JC ra 2/5	AND Rs,A 2/7	AND #n,A 2/7	AND Rs,B 3/9	AND Rs,Rd 3/9	AND #n,B 2/6	AND B,A 1/8	AND #n,Rd 3/8	AND A,Pd 2/9	AND B,Pd 2/9	AND #n,Pd 2/9	INC A 1/8	INC B 1/8	INC Rn 2/6	TRAP 12 1/14	CMP n(SP),A 2/8
4	JP ra 2/5	OR Rs,A 2/7	OR #n,A 2/6	OR Rs,B 2/7	OR Rs,Rd 3/9	OR #n,B 2/6	OR B,A 1/8	OR #n,Rd 3/8	OR A,Pd 2/9	OR B,Pd 2/9	OR #n,Pd 3/10	INV A 1/8	INV B 1/8	INV Rn 2/6	TRAP 11 1/14	extend inst,2 opcodes
5	JPZ ra 2/5	XOR Rs,A 2/7	XOR #n,A 2/6	XOR Rs,B 2/7	XOR Rs,Rd 3/9	XOR #n,B 2/6	XOR B,A 1/8	XOR #n,Rd 3/8	XOR A,Pd 2/9	XOR B,Pd 2/9	XOR #n,Pd 3/10	CLR A 1/8	CLR B 1/8	CLR Rn 2/6	TRAP 10 1/14	
6	JNZ ra 2/5	BTJO Rs,A 3/9	BTJO #n,A 3/8	BTJO Rs,Pd 3/9	BTJO Rs,Rd 4/11	BTJO #n,B 3/8	BTJO B,A 2/10	BTJO #n,Rd 4/10	BTJO A,Pd 3/11	BTJO B,Pd 3/10	BTJO #n,Pd 4/11	XCHB A 1/10	XCHB TESTB 1/10	XCHB Rn 2/8	TRAP 9 1/14	IDLE 1/6
7	JNC ra 2/5	BTJZ Rs,A 3/9	BTJZ #n,A 3/8	BTJZ Rs,B 3/9	BTJZ Rs,Rd 4/11	BTJZ #n,B 3/8	BTJZ B,A 2/10	BTJZ #n,Rd 4/10	BTJZ A,Pd 3/10	BTJZ B,Pd 3/10	BTJZ #n,Pd 4/11	SWAP A 1/11	SWAP B 1/11	SWAP Rn 2/9	TRAP 8 1/14	MOV #n,Pd 3/10
8	JV ra 2/5	ADD Rs,A 2/7	ADD #n,A 2/6	ADD Rs,B 2/7	ADD Rs,Rd 3/9	ADD #n,B 2/6	ADD B,A 1/8	ADD #n,Rd 3/8	MOVW #16,Rd 4/13	MOVW Rs,Rd 3/12	MOVW #16(B),Rd 4/15	PUSH A 1/9	PUSH B 1/9	PUSH Rs 2/7	TRAP 7 1/14	SETC 1/7
9	JL ra 2/5	ADC Rs,A 2/7	SX #n,A 2/6	ADC Rs,B 2/7	ADC Rs,Rd 3/9	ADC #n,B 2/6	ADC B,A 1/8	ADC #n,Rd 3/8	JMPL lab 3/9	JMPL @Rd 2/8	JMPL lab(B) 3/11	POP A 1/9	POP B 1/9	POP Rd 2/7	TRAP 6 1/14	RTS 1/9
A	JLE ra 2/5	SUB Rs,A 2/6	SUB #n,A 2/6	SUB Rs,B 2/7	SUB Rs,Rd 3/9	SUB #n,B 2/6	SUB B,A 1/8	SUB B,A 1/8	MOV lab,A 3/10	MOV @Rs,A 2/9	MOV lab(B),A 3/12	DJNZ A,ra 2/10	DJNZ B,ra 2/10	DJNZ Rn,ra 3/8	TRAP 5 1/14	RTI 1/12
B	JHS ra 2/5	SBB Rs,A 2/7	SBB #n,A 2/6	SBB Rs,B 2/7	SUB Rs,Rd 3/9	SBB #n,B 2/6	SBB B,A 1/8	SBB #n,Rd 3/8	MOV A,lab 3/10	MOV A,@Rd 2/9	MOV A,lab(B) 3/12	COMPL A 1/8	COMPL B 1/8	COMPL Rn 2/6	TRAP 4 1/14	PUSH ST 1/8
C	JNV ra 2/5	MPY Rs, A 2/46	MPY #n,A 2/45	MPY Rs,B 2/46	MPY Rs,Rd 2/46	MPY #n,B 2/48	MPY B,A 1/47	MPY #n,Rd 3/47	BR lab 3/9	BR @Rd 2/8	BR lab(B) 1/8	RR A 1/8	RR B 1/8	RR Rn 2/6	TRAP 3 1/14	POP ST 1/8
D	JGE ra 2/5	CMP Rs,A 2/7	CMP #n,A 2/6	CMP Rs,B 2/7	CMP Rs,Rd 3/9	CMP #n,B 2/6	CMP B,A 1/8	CMP #n,Rd 3/8	CMP lab,A 3/11	CMP @Rs,A 2/10	CMP lab(B),A 3/13	RRC A 1/8	RRC B 1/8	RRC Rn 2/6	TRAP 2 1/14	LDSP ST 1/7
E	JG ra 2/5	DAC Rs,A 2/9	DAC #n,A 2/8	DAC Rs,B 2/9	DAC Rs,Rd 3/11	DAC #n,B 2/8	DAC B,A 1/10	DAC #n,Rd 3/10	CALL lab 3/13	CALL @Rd 2/12	CALL lab(B) 3/13	RL A 1/8	RL B 1/8	RL Rn 2/6	TRAP 1 1/14	STSP ST 1/8
F	JLO ra 2/5	DSB Rs,A 2/9	DSB #n,A 2/8	DSB Rs,B 2/9	DSB Rs,Rd 3/11	DSB #n,B 2/8	DSB B,A 1/10	DSB #n,Rd 3/10	CALLR lab 3/15	CALLR @Rd 2/14	CALLR lab(B) 3/17	RLC A 1/8	RLC B 1/8	RLC Rn 2/6	TRAP 0 1/14	NOP 1/7

FIRST NIBBLE

Second byte of two-byte instructions (F4xx):

	E	F
8	MOVW n(Rn) 4/15	DIV Rn,A 3/14-63
9	JMPL n(Rn) 4/16	
A	MOV n(Rn),A 4/17	
B	MOV A,n(Rn) 4/16	
C	BR n(Rn) 4/16	
D	CMP n(Rn) 4/18	
E	CALL n(Rn) 4/20	
F	CALLR n(R) 4/22	

ra — relative address
Rn — Register
Rs — Register containing source byte
Rd — Register containing destination byte
Ps — Peripheral register containing source byte
Pd — Peripheral register containing destination byte
Pn — Peripheral register
#n — Immediate 8-bit number
#16 — Immediate 16-bit number
lab — 16-bit label
@Rn — 16-bit address of contents of register pair

development system support

The TMS370 family development support tools include an Assembler, a C-Compiler, a Linker, an In-Circuit emulator (XDS – eXtended Development Support), and an EEPROM/UVEPROM programmer. All of the tools work closely together using any MS ™-DOS-Based Personal Computer (PC) or a VMS ™-based DEC VAX™ computer as the host and central control element. This allows selection of the host computer and the text management and editing tools based on user preference.

- Assembler/Linker (Part Number TMDS3740810-02 for PC, Part No. TMDS3740210-08 for VAX/VMS)
  - Extensive macro capability.
  - High-speed operation.
  - Format conversion utilities available for popular formats.
- ANSI C-Compiler (Part No. TMDS3740815-02 for PC, Part No. TMD3704215-08 for VAX/VMS)
  - Generates assembly code fo the TMS370 that can be easily inspected.
  - The compilation, assembly and linking steps can all be performed with a single command.
  - Enables the user to directly reference the TMS370's port registers by using a naming convention.
  - Provides flexibility in specifying the storage for data objects.
  - C-functions and Assembly functions can be easily interfaced.
- XDS/11 (eXtended Development Support) In-Circuit Emulator (Part Number TMDS3761111)
  - PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
  - Symbolic debugging.
  - Execute single/multiple instructions, single/multiple statements, until/while condition, or at full speed until breakpoint.
  - The user needs to provide a regulated 5 V power supply with a 3A current capability.
- XDS/22 (eXtended Development Support) In-Circuit Emulator (Part Number TMDS3762210)
  - Contains all of the features of the XDS/11 described above but does not require an external power supply.
  - Contains sophisticated breakpoint trace and timing hardware that provides up to 2047 qualified trace samples with symbolic disassembly.
  - Allows break points to be qualified by address and/or data on any type of memory acquisition. Up to four levels of events can be combined to cause a breakpoint.
  - Provides timers fo analyzing total and average time in routines.
  - Contains an eight line logic probe for adding visibility of external signals to the breakpoint qualifier and to the trace display.
- EEPROM/EPROM Programmer (Part Number TMDS3760510)
  - PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
  - Supports all TMS370 series devices to provide rapid target prototyping capability.
  - Also programs TMS2732, TMS2764, TMS27128, and TMS27256 NMOS and CMOS EPROMs.
  - Plug-in personality boards and loadable parameters to support future packages and devices.
- Design Kit (Part No. TMDS3770110)
  - Includes TMS370 Application Board and TMS370 Assembler diskette and documentation.
  - Supports quick evaluation of TMS370 functionality.
  - Capability to upload and download code.
  - Capability to execute programs and software routines, and to single-step executable instructions.
  - Software breakpoints to halt program execution at selected addresses.
  - Wire-wrap prototype area.
  - Reverse assembler.

MS is a trademark of Microsoft Corporation. **VMS,** **DEC**, and **VAX** are trademarks of Digital Equipment Corporation.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V_{CC} (see Note 6) ..... – 0.6 to 7 V
Input voltage range, All pins except MC ..... – 0.6 to 7 V
MC ..... – 0.6 to 14 V
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{CC}$) ..... ±20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{CC}$) ..... ±20 mA
Continuous output current per buffer, I_O ($V_O = 0$ to V_{CC}) (see Note 5) ..... ±10 mA
Maximum I_{CC} current ..... 170 mA
Maximum I_{SS} current ..... –170 mA
Continuous power dissipation ..... 500 mW
Storage temperature range ..... – 65°C to 150 °C

† Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTES: 5. Electrical characteristics are specified with all output buffers loaded with the specified I_O current. Exceeding the specified I_O current in any buffer may affect the levels on other buffers.

recommended operating conditions

PARAMETER			MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage (see Note 6)		4.5	5	5.5	V
V_{CC}	RAM data-retention supply voltage (see Note 7)		3		5.5	V
V_{IL}	Low-level input voltage	All pins except MC	V_{SS}		0.8	V
		MC, normal operation	V_{SS}		0.3	
V_{IH}	High-level input voltage	All pins except MC, XTAL2/CLKIN, and $\overline{RESET}$	2		V_{CC}	V
		MC/Write Protect Override (WPO)	11.7		13	
		XTAL2/CLKIN	0.8 V_{CC}		V_{CC}	
		$\overline{RESET}$	0.7 V_{CC}		V_{CC}	
T_A	Operating free-air temperature	A version	– 40		85	°C
		L version	0		70	°C

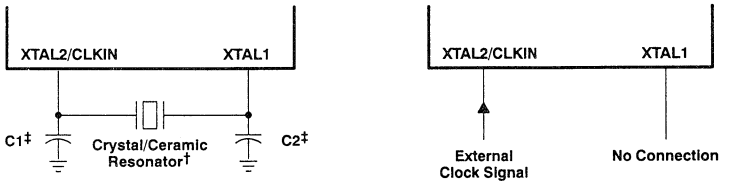
NOTES: 6. Unless otherwise noted, all voltages are with respect to V_{SS}.
7. To guarantee RAM data retention form 3 V to 4.5 V, $\overline{RESET}$ must be externally asserted and released only while V_{CC} is within the recommended operating range of 4.5 V to 5.5 V.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OL} Low-level digital output voltage		I_{OL} = 1.4 mA			0.4	V
V_{OH} High-level output voltage		I_{OH} = – 50 μA	0.9 V_{CC}			V
		I_{OH} = – 2 mA	2.4			
I_I Input current	MC	0 V ≤ V_I ≤ 0.3 V			10	μA
		0.3 V < V_I ≤ 13 V			650	
	I/O pins	0 V ≤ V_I ≤ V_{CC}			± 10	μA
I_{OL} Low-level output current		V_{OL} = 0.4 V	1.4			mA
I_{OH} High-level output current		V_{OH} = 0.9 V_{CC}	– 50			μA
		V_{OH} = 2.4 V	– 2			mA
I_{CC} Supply current (Operating mode) Osc Power bit = 0 (see Note 10)		(see Notes 8 and 9) CLKIN = 20 MHz		20	36	mA
		(see Notes 8 and 9) CLKIN = 12 MHz		13	25	
		(see Notes 8 and 9) CLKIN = 2 MHz		5	11	
I_{CC} Supply current (Standby mode) Osc Power bit = 0 (see Note 11)		(see Notes 8 and 9) CLKIN = 20 MHz		10	17	mA
		(see Notes 8 and 9) CLKIN = 12 MHz		6.5	11	
		(see Notes 8 and 9) CLKIN = 2 MHz		2	3.5	
I_{CC} Supply current (Standby mode) Osc Power bit = 1 (see Note 12)		(see Notes 8 and 9) CLKIN = 12 MHz		4.5	8.6	mA
		(see Notes 8 and 9) CLKIN = 2 MHz		1.5	3.0	
I_{CC} Supply current (Halt Mode)		(see Note 8) XTAL2/CLKIN < 0.2 V		1	30	μA

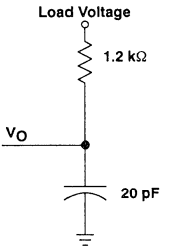
NOTES: 8. Single chip mode, ports configured as inputs, or outputs with no load. All inputs ≤ 0.2 V or ≥ V_{CC} – 0.2 V.
9. XTAL2/CLKIN is driven with an external square wave signal with 50% duty cycle and rise and fall times less than 10 ns. Currents may be higher with a crystal oscillator. At 20 MHz this extra current = .01 mA × (total load capacitance + crystal capacitance in pF).
10. Maximum operating current for TMS370Cx10 = 1.4 (CLKIN) + 8 mA.
11. Maximum standby current for TMS370Cx10 = 0.75 (CLKIN) + 2 mA.
12. Maximum standby current for TMS370Cx10 = 0.56 (CLKIN) + 1.9 mA. (Osc power bit valid only from 2 MHz to 12 MHz.)

Recommended Crystal/Clock Connections



† The crystal/ceramic resonator frequency is four times the reciprocal of the system clock period.
‡ The values of C1 and C2 should be the values recommended by the crystal/ceramic resonator manufacturer.

Typical Output Load Circuit§



Case 1: $V_O = V_{OH}$ = 2.4 V; Load Voltage = 0 V
Case 2: $V_O = V_{OL}$ = 0.4 V; Load Voltage = 2.1 V

§ All measurements are made with the pin loading as shown unless otherwise noted. All measurements are made with XTAL2/CLKIN driven by an external square wave signal with a 50% duty cycle and rise and fall times less than 10 ns unless otherwise stated.

timing parameter symbology

Timing parameter symbols have been created in accordance with JEDEC Standard 100. In order to shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

AR	Array	S	Slave mode
B	Byte	SIMO	SPISIMO
CI	XTAL2/CLKIN	SOMI	SPISOMI
CO	CLKOUT	SPC	SPICLK
PGM	Program		

Lowercase subscripts and their meanings are:

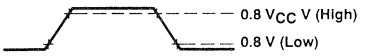
c	cycle time (period)	su	setup time
d	delay time	v	valid time
f	fall time	w	pulse duration (width)
r	rise time	x	oscillator

The following additional letters are used with these meanings:

H	High
L	Low
V	Vallid

PARAMETER MEASUREMENT INFORMATION

All timings are measured between high and low measurement points as indicated in the figures below.



XTAL2/CLKIN Measurment Points

General Measurement Points

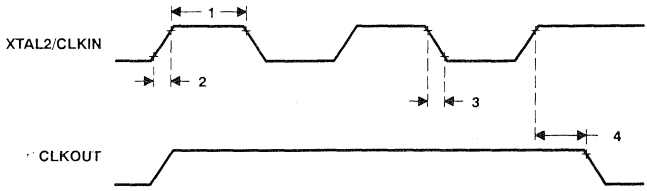
external clocking requirements†

NO.	PARAMETER		MIN	NOM	MAX	UNIT
1	$t_{w(CI)}$	XTAL2/CLKIN pulse duration (see Note 13)	20			ns
2	$t_{r(CI)}$	XTAL2/CLKIN rise time			30	ns
3	$t_{f(CI)}$	XTAL2/CLKIN fall time			30	ns
4	$t_{d(CIH-COL)}$	Delay time, XTAL2/CLKIN rise to CLKOUT fall			100	ns
	CLKIN	Crystal operating frequency	2		20	MHz

† For V_{IL} and V_{IH}, refer to "Recommended Operating Conditions".

NOTE 13: This pulse may be either a high pulse, as illustrated below, which extends from the earliest valid high to the final valid high in an XTAL2/CLKIN cycle, or a low pulse, which extends from the earliest valid low to the final valid low in an XTAL2/CLKIN cycle.

external clock timing

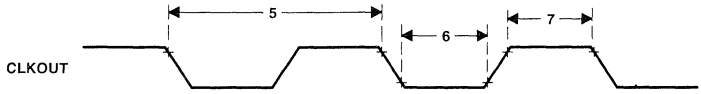


switching characteristics and timing requirements ‡

NO.	PARAMETER		MIN	MAX	UNIT
5	t_c	CLKOUT (system clock) cycle time	200	2000	ns
6	$t_{w(COL)}$	CLKOUT low pulse duration	$0.5t_c$-20	$0.5t_c$	ns
7	$t_{w(COH)}$	CLKOUT high pulse duration	$0.5t_c$	$0.5t_c$+20	ns

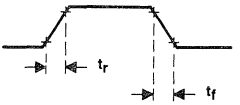
‡ t_c = system clock cycle time = 4/CLKIN.

CLKOUT timing



general purpose output signal switching time requirements

		MIN	NOM	MAX	UNIT
t_r	Rise time			45	ns
t_f	Fall time			45	ns



recommended EEPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w((PGM)B}$	Programming signal pulse duration to insure valid data is stored (byte mode)	10			ms
tw(PGM)AR	Programming signal pulse duration to insure valid data is stored (array mode)	20			ms

recommended EPROM operating conditions for programming

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.75	5.5	6.0	V
V_{PP}	Supply voltage at MC pin	12	12.5	13	V
I_{PP}	Supply current at MC pin during programming (V_{PP} = 13 V)		35	50	mA
CLKIN	Operating crystal frequency	2		20	MHz

recommended EPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(IEPGM)}$	Initial programming pulse (see Note 14)	0.95	1	1.05	ms
$t_{w(FEPGM)}$	Final programming pulse	2.85		78.75	ms

NOTE 14: Programming pulse is active when both EXE (EPCTL.0) and VPPS (EPCTL.6) are set.

Serial Peripheral Interface (SPI) Timing

SPI master external timing characteristics†

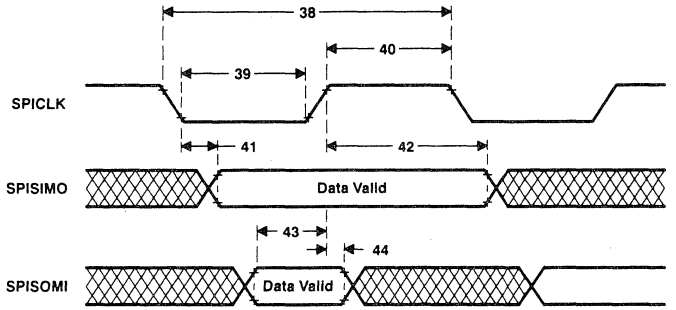
NO.	PARAMETER		MIN	MAX	UNIT
38	$t_{C(SPC)}$	SPICLK cycle time	$2t_C$	$256t_C$	ns
39	$t_{w(SPCL)}$	SPICLK-low pulse duration	$t_C - 45$	$0.5t_{C(SPC)}+45$	ns
40	$t_{w(SPCH)}$	SPICLK-high pulse duration	$t_C - 45$	$0.5t_{C(SPC)}+45$	ns
41	$t_{d(SPCL-SIMOV)}$	Delay time, SPISIMO valid after SPICLK low (Polarity =1)	− 50	50	ns
42	$t_{v(SPCH-SIMO)}$	SPISIMO data valid after SPICLK high (Polarity =1)	$t_{w(SPCH)} - 50$		ns

SPI master external timing requirements†

NO.			MIN	MAX	UNIT
43	$t_{su(SOMI-SPCH)}$	SPISOMI setup time to SPICLK high (Polarity = 1)	$0.25t_C + 150$		ns
44	$t_{v(SPCH-SOMI)}$	SPISOMI data valid afterSPICLK high (Polarity = 1)	0		ns

† t_C = system clock cycle time = 4/CLKIN.

SPI master external timing



NOTE 15: The diagram above is for Polarity = 1. SPICLK is inverted from above diagram when Polarity = 0.

SPI slave external timing characteristics†

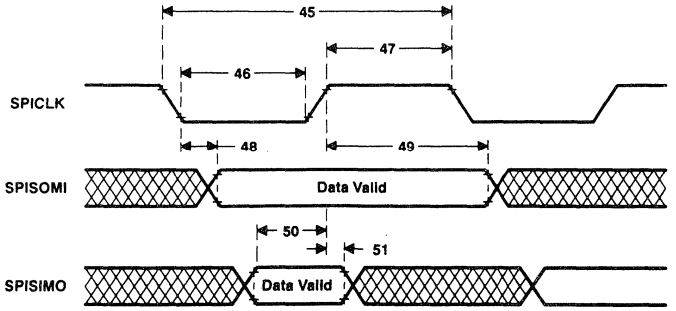
NO.	PARAMETER		MIN	MAX	UNIT
48	$t_{d(SPCL-SOMIV)S}$	Delay time, SPISOMI valid after SPICLK low (Polairty = 1)		$3.25t_c + 125$	ns
49	$t_{v(SPCH-SOMI)S}$	SPISOMI data valid after SPICLK high (Polarity =1)	$t_{w(SPCH)S}$		ns

SPI slave external timing requirements†

NO.	PARAMETER		MIN	MAX	UNIT
45	$t_{c(SPC)S}$	SPICLK cycle time	$8t_c$		ns
46	$t_{w(SPCL)S}$	SPICLK-low pulse duration	$4t_c - 45$	$0.5t_{c(SPC)S}+45$	ns
47	$t_{w(SPCH)S}$	SPICLK high pulse duration	$4t_c - 45$	$0.5t_{c(SPC)S}+45$	ns
50	$t_{su(SIMO-SPCH)S}$	SPISIMO setup time to SPICLK high (Polarity = 1)	0		ns
51	$t_{v(SPCH-SIMO)S}$	SPISIMO data valid after SPICLK high (Polarity = 1)	$3t_c + 100$		ns

† t_c = system clock cycle time = 4/CLKIN.

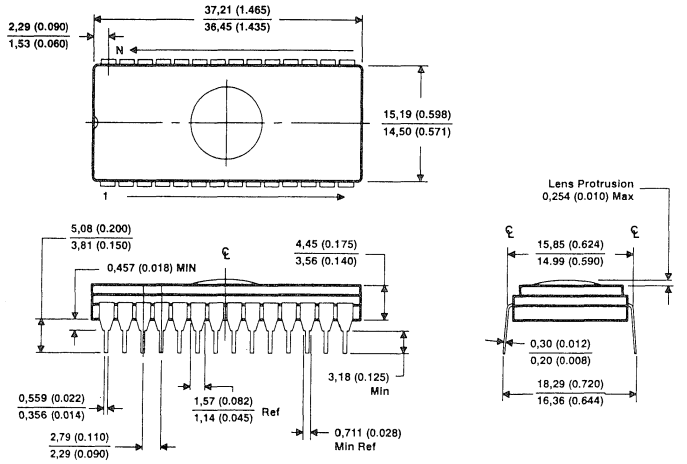
SPI slave external timing



NOTES: 16. The diagram above is for Polarity =1. SPICLK is inverted from above diagram when Polarity = 0.
17. As a slave, the SPICLK pin is used as the input for the serial clock, which is supplied from the network master.

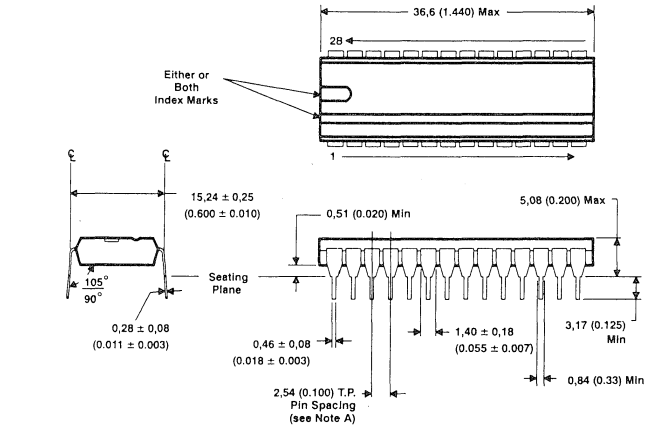
MECHANICAL DATA

28-pin ceramic leaded chip carrier package (J suffix)



ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHICALLY IN INCHES

28-pin plastic leaded chip carrier package (N suffix)

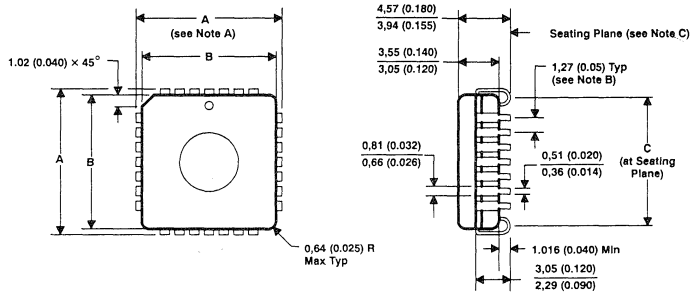


ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHICALLY IN INCHES

NOTE A: Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.

MECHANICAL DATA

FZ cerquad chip carrier package

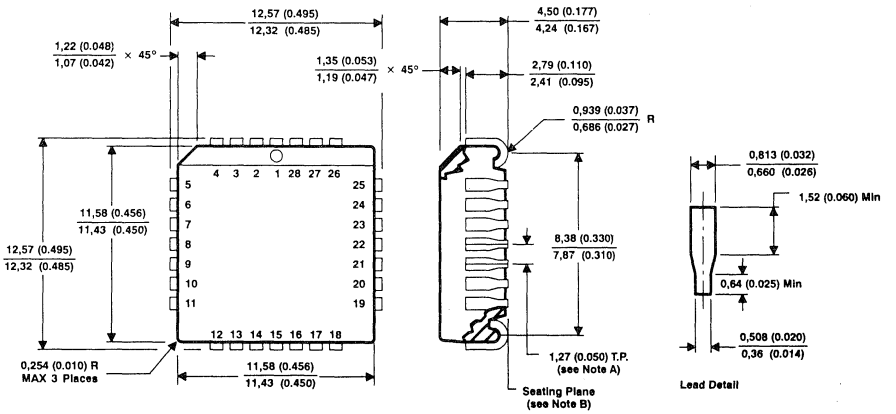


		A		B		C	
		MIN	MAX	MIN	MAX	MIN	MAX
M0-087AA	28	12,32 (0.485)	12,57 (0.495)	10,92 (0.430)	11,56 (0.455)	10,41 (0.410)	10.92 (0.430)
M0-087AB	44	17,40 (0.685)	17,65 (0.695)	16,00 (0.630)	16,64 (0.655)	15,49 (0.610)	16,00 (0.630)
—	68	25,02 (0.985)	25,27 (0.995)	23,62 (0.930)	24,26 (0.955)	23.11 (0.910)	23,62 (0.930)

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHICALLY IN INCHES

NOTES: A. Center line of center pin each side is within 0,10 (0.004) of package centerline as determined by dimension B.
B. Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
C. The lead contact points are planar with 0,15 (0.006).

28-pin plastic leaded chip carrier package (FN suffix)



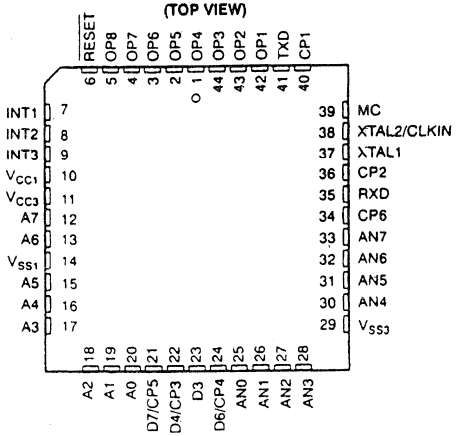
ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHICALLY IN INCHES

NOTES: A. Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
B. The lead contact points are planar within 0,101(0.004).

FEBRUARY 1991

- TMS370 Configured Microcontroller
  - PACT Module
  - A/D Module
  - Data EEPROM Module
- CMOS, EEPROM, EPROM, and A/D Technologies on a Single Device
  - EEPROM Programming Via Single 5-V Supply
  - EPROM Programming Via External Supply
- Flexible Operating Features
  - Power Reduction STANDBY and HALT Modes
  - Temperature– 40°C to 85°C
  - Input Clock Frequency 2 MHz to 20 MHz
  - Voltage (V_{CC}) 5 V ± 10%
- Internal System Memory Configurations
  - 8K-Byte On-Chip Program Memory
  - Program EPROM (TMS370C732)
  - Data EEPROM, 256 Bytes
  - 256 Bytes Static RAM, with 128 Bytes Usable by the PACT Module; All 256 Bytes Usable as Registers
- Programmable Acquisition and Control Timer (PACT) Module
  - Input Capture on up to 6 Pins, 4 of Which May Have a Programmable Prescaler
  - One Input Capture Pin Can Drive an 8-Bit Event Counter
  - Up to 8 Timer-Driven Outputs
  - Interaction Between Event Counter and Timer Activity
  - 18 Independent Interrupt Vectors
  - Watchdog with Selectable Time-Out Period
  - Mini SCI
- Eight-Channel 8-Bit A/D Converter
- 250 mA Typical Latch-Up Immunity at 25°C
- ESD Protection Exceeds 2,000 V per MIL-STD-883C Method 3015

FN AND FZ PACKAGES
(TOP VIEW)



- 36 CMOS/TTL Compatible I/O Pins
  - 14 Bidirectional Pins, 13 Input Pins, and 9 Output Pins
- 44-Pin Plastic Leaded Chip Carrier (PLCC FN Suffix), or 44-Pin Ceramic Leaded Chip Carrier (CLCC FZ Suffix)
- Flexible Interrupt Handling
  - Two S/W Programmable Interrupt Levels
  - Global and Individual Interrupt Masking
  - Programmable Rising or Falling Edge Detect
- TMS370 Series Compatibility
  - Register-to-Register Architecture
  - 256 General-Purpose Registers
  - 14 Powerful Addressing Modes
- PC-Based Workstation Development Support Emphasizes Productivity, Featuring:
  - Realtime In-Circuit Emulation
  - Symbolic Debug
  - Extensive Breakpoint/Trace Capability
  - Software Performance Analysis
  - Multi-Window User Interface
  - EEPROM/EPROM Programming

TEXAS INSTRUMENTS

description

The TMS370C732 device is a member of the TMS370 family of single-chip configurable 8-bit microcontrollers that include the PACT module. The TMS370 family provides cost-effective realtime system control through VLSI integration of advanced on-chip memory and peripheral function modules.

The TMS370C732 device contains the following modules:

256 bytes static RAM (usable as registers), 128 bytes of which are dual port.

Programmable Acquisition and Control Timer (PACT).

Eight-channel eight-bit A/D converter.

8k bytes EPROM of program memory.

256 bytes DATA EEPROM.

This family is implemented using high-performance silicon-gate CMOS technology. The low operating power, wide operating temperature range, and high noise immunity of CMOS technology coupled with the high performance and extensive on-chip peripheral functions make the TMS370C732 devices attractive in system designs for automotive electronics, industrial control, computer peripheral control, and motor control.

Uses of the TMS370C732 include prototyping capabilities, and a low-volume alternative to the mask ROM devices in applications where program constraints are likely to change periodically.

The TMS370C732 provides two power reduction modes (STANDBY and HALT) for the applications where low power consumption is critical. Both modes stop all CPU activity (i.e., no instructions are executed). In STANDBY mode, the internal oscillator, the PACT counter, and PACT's first Command/Definition entry remain active. This allows the PACT module to wake the device out of STANDBY mode. In HALT mode, all device activity is stopped. The device retains all RAM data and peripheral configuration bits throughout both power reduction modes.

The TMS370C732 features advanced register-to-register architecture that allows direct arithmetic and logical operations without requiring an accumulator (e.g., ADD R24, R47; add the contents of register 24 to the contents of register 47 and store the result in register 47). The TMS370C732 instruction set is fully compatible with other TMS370 family members, allowing easy transition between members.

The TMS370 family provides the system designer with an economical, efficient solution to realtime control applications. The TMS370 family eXtended Development System (XDS) solves the challenge of efficiently developing the software and hardware required to design the TMS370 into an ever-increasing number of complex applications. The TMS370 family XDS communicates via a standard RS-232-C interface with an existing personal computer to form a PC-DOS hosted workstation, using the PC's editors and software utilities already familiar to the designer. The TMS370 family XDS emphasizes compatibility and ease-of-use through extensive use of menus and screen windowing so that a system designer can begin developing software with minimum training. Precise realtime in-circuit emulation and extensive symbolic debug and analysis tools ensure efficient software and hardware implementation as well as reduced time-to-market.

The TMS370 family provides the systems designer a complete solution:

TMS370Cx32 mask ROM devices

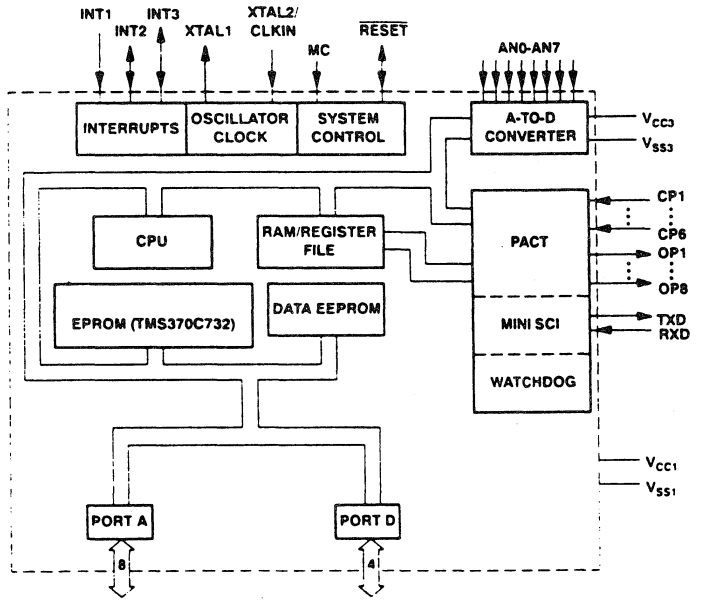
TMS370C732 8K-byte program EPROM device

TMS370 family PACT XDS/22 for applications development

Comprehensive product documentation

Customer hotline support

functional block diagram



NOTE Three of Port D's four I/O buffers (D4, D6, and D7) are internally connected to three of the PACT module's inputs (CP3, CP4, and CP5). The actual pins are D4:CP3, D6:CP4, and D7:CP5.

pin descriptions

PIN NAME	PIN NO	I/O	DESCRIPTION
A0	20	I/O	Port A is a general purpose bidirectional port
A1	19	I/O	
A2	18	I/O	
A3	17	I/O	
A4	16	I/O	
A5	15	I/O	
A6	13	I/O	
A7	12	I/O	
			Port D is a general purpose bidirectional port
D3	23	I/O	Also configurable as CLKOUT (see Note 1).
D4 CP3	22	I/O	PACT input capture 3 (see Note 2)
D6 CP4	24	I/O	PACT input capture 4 (see Note 2).
D7 CP5	21	I/O	PACT input capture 5 (see Note 2).
INT1	7	I	External interrupt (non-maskable or maskable)/general purpose input pin.
INT2	8	I/O	External maskable interrupt input/general purpose bidirectional pin.
INT3	9	I/O	External maskable interrupt input/general purpose bidirectional pin.
CP1	40	I	PACT input capture pin 1.
CP2	36	I	PACT input capture pin 2.
CP6	34	I	PACT input capture pin 6. External Event input pin (for event counter).
TXD	41	O	PACT SCI transmit output pin.
RXD	35	I	PACT SCI receive input pin.
OP1	42	O	PACT output pin 1.
OP2	43	O	PACT output pin 2.
OP3	44	O	PACT output pin 3.
OP4	1	O	PACT output pin 4.
OP5	2	O	PACT output pin 5.
OP6	3	O	PACT output pin 6.
OP7	4	O	PACT output pin 7.
OP8	5	O	PACT output pin 8.
AN0	25	I	A/D analog input (AN0 — AN7) or positive reference pins (AN1 — AN7).
AN1	26	I	
AN2	27	I	
AN3	28	I	The analog port may be individually programmed as general purpose input pins if not used as A/D
AN4	30	I	converter analog input or positive reference input.
AN5	31	I	
AN6	32	I	
AN7	33	I	
$\overline{\text{RESET}}$	6	I/O	System reset bidirectional pin. As input it initializes microcontroller, as open-drain output it indicates an internal failure was detected by the watchdog or oscillator fault circuit.
MC	39	I	Microcomputer mode control input pin; also enables EEPROM write protection override (WPO) mode.
XTAL2/ CLKIN	38	I	Internal oscillator crystal input/External clock source input.
XTAL1	37	O	Internal oscillator output for crystal.
V_{CC1}	10		Positive supply voltage for digital logic and digital I/O pins.
V_{SS1}	14		Ground reference for digital logic and digital I/O pins.
V_{CC3}	11		A/D converter positive supply voltage and optional positive reference input.
V_{SS3}	29		A/D converter ground supply and low reference input pin.

NOTES: 1. D3 may be configured as CLKOUT by appropriately programming the DPORT1 and DPORT2 registers.
2. These digital I/O buffers are internally connected to some of the PACT module's input capture pins. This allows the microcontroller to read the level on the input capture pin, or if the Port D pin is configured as an output, to generate a capture. Be careful to leave the Port D pin configured as an input if the corresponding input capture pin is being driven by external circuitry.

memory map

The TMS370 family architecture is based on the Von Neumann architecture, where the program memory and data memory share a common address space. All peripheral input/output is memory mapped in this same common address space. As shown in Figure 1, the TMS370 provides memory-mapped RAM, ROM, EEPROM, EPROM, input/output pins, and peripheral functions.

The peripheral file contains all input/output port control, peripheral status and control, EEPROM/EPROM memory programming, and system-wide control functions. The peripheral file is located between 1010h to 104Fh and is logically divided into 4 Peripheral File Frames of 16 bytes each. Each on-chip peripheral is assigned to a separate peripheral frame through which control and data information is passed. The TMS370C732 has four peripheral and system control frames assigned to Peripheral File Frames 1, 2, 4 and 7, addresses 1010h —101Fh, 1020h —102Fh, 1040h —104Fh, and 1070h — 107Fh.

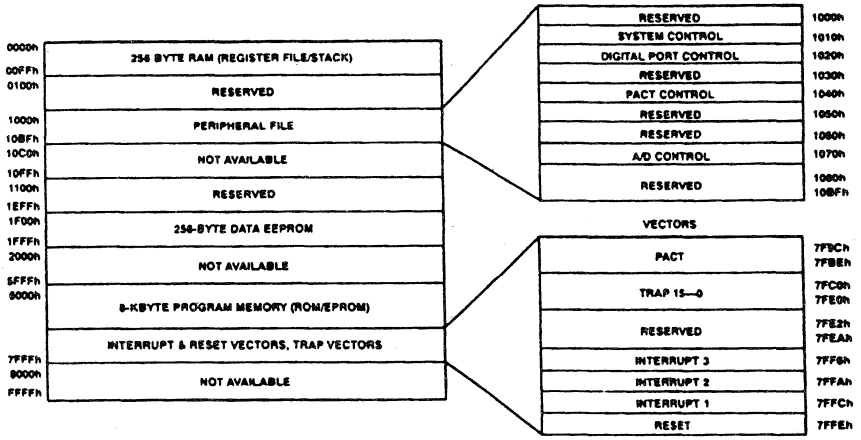


FIGURE 1. TMS370C732 MEMORY MAP

memories

RAM/register file

The TMS370C732 has 256 bytes of static RAM, which serve as both the CPU register file and general-purpose memory. The RAM is treated as registers by the instruction set and is referenced as R0 through R255. The first two registers, R0 and R1, are also called the A and B registers, respectively. The stack is located in the RAM, and operates as a last-in first-out read/write memory. It is used to store the return address on subroutine calls and the contents of the status register at time of interrupt. Accessing this memory as registers is performed in one system clock cycle (t_c), while general-purpose memory access is performed in two system clock cycles.

Instructions may be executed from RAM. This versatility enables the internal RAM to be used for functions such as programming the on-chip EEPROM. The user may load external programs or data into the RAM by incorporating a simple loader in the program memory.

The upper 128 bytes of the register files may be used by the PACT module to contain module commands and definitions as well as timer values. Any RAM not used by PACT may be used as additional CPU registers or as general-purpose memory.

data EEPROM

The TMS370C732 EPROM has 256 bytes of on-chip electrically erasable programmable ROM (EEPROM), addressed as 256 consecutive bytes mapped from locations 1F00h to 1FFFh. The data EEPROM provides nonvolatile programmable storage for items such as calibration constants and configuration information for personalization of a generic program algorithm used in specific end applications. The data EEPROM supports bit, byte, and block write/erase modes. Instructions may be executed from data EEPROM, providing additional program space and the ability to patch program algorithms by placing a branch table for volatile routines in data EEPROM.

The data EEPROM uses the 5-V V_{CC} supply voltage and provides the programming voltage via an internal dedicated generator, eliminating the need for an external high-voltage programming source. The dedicated voltage generator optimizes the programming voltage characteristics, increasing the reliability as well as extending the write/erase endurance of the array.

Programming control and status monitoring are performed through the data EEPROM control register (DEECTL) in the peripheral file. An EEPROM write/erase operation is performed in the following sequence:

1. Perform normal memory write to the target EEPROM location.
2. Write to DEECTL control register to select WRITE1/WRITE0 and set the EXE bit to 1.
3. Wait for program time to elapse ($t_{w(PGM)B}$ or $t_{w(PGM)AR}$).
4. Write to DEECTL control register to set EXE bit to 0.

The WRITE1/WRITE0 control bit selects whether the zeros or the ones in the data byte are to be programmed into the selected EEPROM location. For example, a WRITE1 operation will program ones into all bit positions within the EEPROM byte that have ones in the data byte, while bits that are zero in the data byte will not affect the EEPROM contents. The WRITE1 operation effectively performs a logical OR of the information previously stored on the EEPROM byte with the data byte. The WRITE0 operation effectively performs a logical AND of the data byte. The EEPROM programming algorithm may use this bit-programming capability to optimize the life of the EEPROM.

When a data value cannot be achieved by writing only zeros or only ones into the EEPROM byte, a WRITE1 followed by a WRITE0 will program any data value into the EEPROM byte, regardless of the previous data stored at that location.

All unprotected bytes within the data EEPROM array may be programmed during a single EEPROM programming cycle by setting the ARRAY PROG bit of DEECTL to 1 at the start of the programming cycle.

Data EEPROM read accesses are performed as normal memory read operations in two system clock cycles. A memory read cycle to any EEPROM location while EXE = 1 returns the value currently being written to the EEPROM. Following an EEPROM write operation, the EEPROM voltages must stabilize prior to performing an EEPROM read operation. The BUSY flag indicates the status of the EEPROM voltage. When set, the EEPROM is not ready for a read operation. The BUSY flag is reset to 0 by the EEPROM control logic when 128 system clock cycles have elapsed following the EXE bit being set to 0. If an EEPROM read operation is performed while BUSY = 1, automatic WAIT states will be generated until BUSY = 0, and then the read operation will be performed.

Bytes within the data EEPROM may be protected from inadvertent overwriting of critical information. As shown in Figure 2, the 8-bit write protect register (WPR), located at 1F00h within the data EEPROM, provides write protection for the 256-byte data EEPROM, segmenting the array into eight blocks of 32 bytes each. Each of these 32-byte blocks may be individually write- and erase-protected by setting the corresponding bit to 1 in the WPR. Since the WPR resides in the array in BLK0, the WPR may also be write-protected, thereby increasing the system reliability by preventing bytes from being reprogrammed. Bytes left unprotected may be written to by the normal EEPROM programming sequence. The write protection override (WPO) mode overrides the write protection of all blocks in the data EEPROM, and enables data to be written to any location in the data EEPROM, regardless of the WPR contents. Enter the WPO mode by placing 12 V on the MC pin. The WPO mode is typically used in

a service environment to update the protected EEPROM contents. The 12 V input level on the MC pin to enter the WPO mode is not normally present in an application except in a service environment; therefore, the data integrity of the program is ensured during normal operation.

To guarantee that the contents of nonprotected EEPROM locations are not corrupted, the CPU (device) must be in reset when the supply voltage is not within the recommended operating conditions.

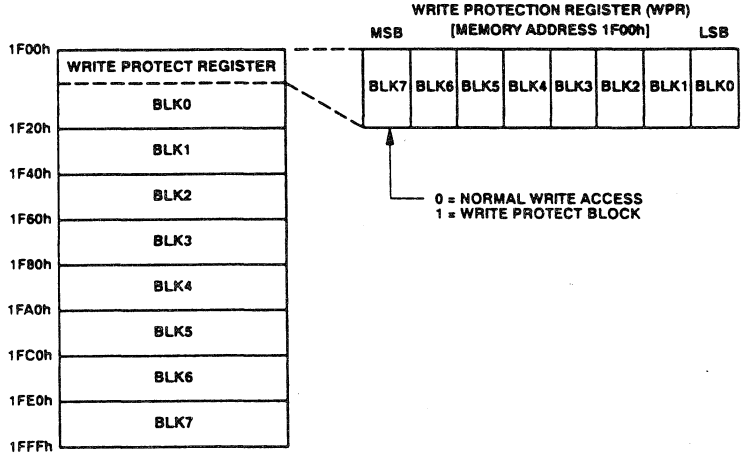


FIGURE 2. WRITE PROTECTION BITS

program EPROM

The program EPROM of the TMS370C732 is an 8K ultraviolet-light-erasable, electrically programmable read-only memory, addressed as consecutive bytes mapped form location 6000h to 7FFFh. It provides application performance identical to the TMS370Cx32 mask ROM devices. Program instructions are read from the program EPROM in two system clock cycles, providing the prototype capability of the mask program ROM.

An external supply (V_{PP}) is needed at the MC pin to provide the necessary voltage V_{PP} for programming. Programming is controlled through register EPCTL in the peripheral file.

Before programming, the TMS370C732 EPROM must be erased by exposing the device through the transparent window to high-intensity ultraviolet light (wavelength 2537angstroms). The recommended minimum exposure dose (UV intensity × exposure time) is 15 watt-seconds per square centimeter. A typical 12-milliwatt-per-square-centimeter, filterless UV lamp will erase the device in 21 minutes. The lamp should be located about 2.5 centimeters above the chip during erasure. After erasure, the entire array is in logic 1 state. A programmed 0 can be erased only by exposure to ultraviolet light. It should be noted that normal ambient light contains the correct wavelength for erasure. Therefore, when using the TMS370C732, the window should be covered with an opaque label. All devices are erased to logical 1 upon delivery from the factory.

CAUTION

Exposing the EPROM module to ultraviolet light may also cause erasure in any EEPROM module. Any data stored in the EEPROM must be reprogrammed after exposure to UV light.

Programming 0 to the EPROM is controlled by the EPCTL register via the EXE bit and the V_{PPS} bit. The EXE bit initiates EPROM programming when set and disables programming when cleared. The V_{PPS} bit connects the programming voltage V_{PP} at the MC pin to the EPROM module. V_{PPS} (EPCTL.6) and EXE (EPCTL.0) should be set separately, and the V_{PPS} bit should be set at least two microseconds before the EXE bit is set. After programming, the application programming should wait for four microseconds before any read attempt is made. The programming operation (see Figure 3) is performed in the following suggested sequence:

1. Supply the programming voltage to the MC pin.
2. Write to EPCTL register to set the V_{PPS} bit to 1.
3. Perform normal memory write register to the target EPROM location.
4. Write to EPCTL register to set the EXE bit register to 1. (Wait at least two microseconds after step 2.)
5. Wait for program time register to elapse (one millisecond).
6. Write to EPCTL register to clear the EXE bit (leave V_{PPS} set to 1).
7. Read the byte being programmed; if correct data is not read, repeat steps 4 through 6 *X* times up to a maximum of 25.
8. Write to EPCTL register to set the EXE bit to 1 for Final programming.
9. Wait for program time to elapse (3*X* milliseconds duration).
10. Write to EPCTL register to clear the EXE and V_{PPS} bits.

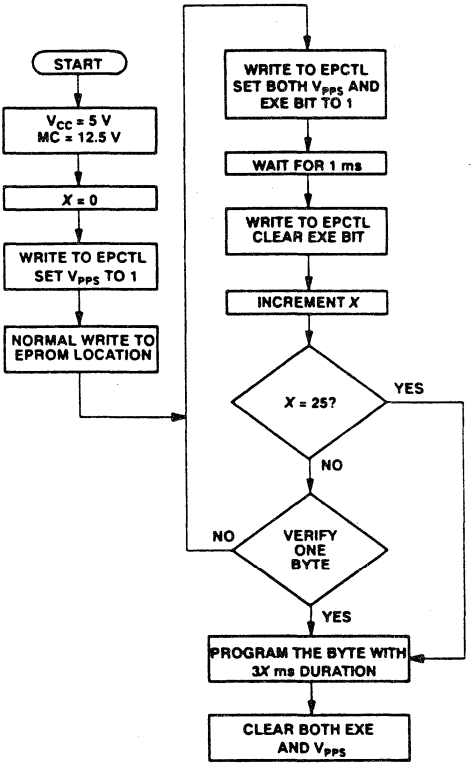


FIGURE 3. EPROM PROGRAMMING OPERATION

An external power supply at V_{PP}, I_{PP} (30 mA), is required for programming operation. Programming voltage V_{PP} is supplied via the MC pin. This also automatically puts the microcontroller in the write protection override (WPO) mode. Programming voltage may be applied via the MC pin anytime after reset and remain at V_{PP} after programming (after the EXE bit is cleared). Applying programming voltage while $\overline{\text{RESET}}$ is active will put the microcontroller in a reserved mode, where programming operation is inhibited.

write protect of program EPROM

To override the EPROM write protection, the V_{PP} voltage must be applied to the MC pin and the V_{PPS} bit (EPCTL.6) must be set. This dual requirement ensures that the program EPROM will not accidentally be overwritten during data EEPROM operations when V_{PP} is applied to the MC pin. Data EEPROM may be programmed when the V_{PPS} bit is set.

central processing unit

The central processing unit (CPU) of the TMS370 series is an enhanced version of the TMS7000 Family CPU. The enhancements include additional user instructions such as integer divide and conditional jump instructions based on the overflow status bit, and addressing modes such as stack-pointer-relative addressing for subroutine parameter passing. The efficient register-to-register architecture of the TMS7000 family has been carried over to the TMS370 series to avoid the conventional accumulator bottleneck. The complete TMS370 series instruction set is summarized in the TMS370 Instruction Overview.

In addition to the interpretation and execution of the user program, the CPU performs the functions of bus protocol generation and interrupt priority arbitration. While the CPU is implemented independent of the memory, input/output, and peripheral modules, it performs the central system control function through communications with these on-chip modules and external memory and peripherals.

The TMS370 series CPU registers accessible to the programmer are shown in Figure 4. The register file consists of 256 general-purpose registers, R0 through R255, implemented in on-chip RAM, and is used by the CPU for general-purpose 16-bit source and destination operands, index registers, and indirect addressing. The first two registers, R0 and R1, are also called registers A and B and are used by the CPU as general-purpose registers or for implied operands. The program counter (PC) contains the address of the next instruction to be executed. The stack pointer (SP) contains the address of the last or top entry on the stack, which is located in the on-chip register file. The status register (ST) contains four bits that reflect the outcome of the instruction just executed, and two bits that control the masking of the interrupt priority chains.

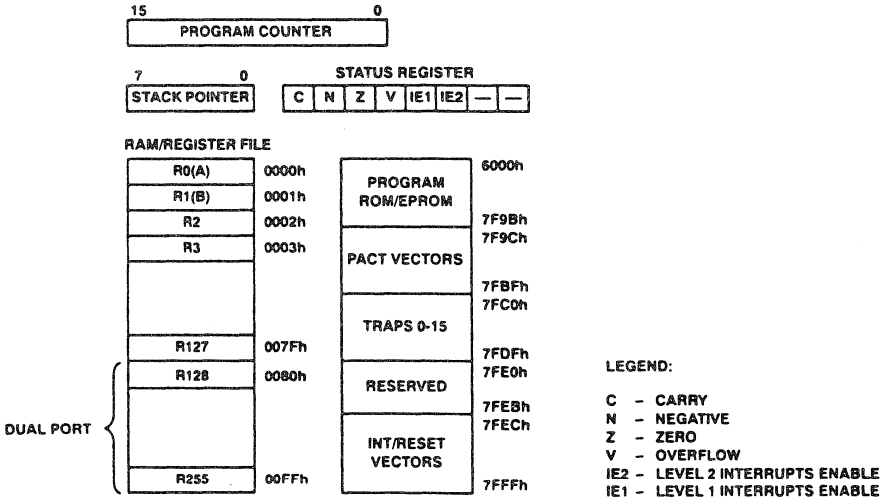
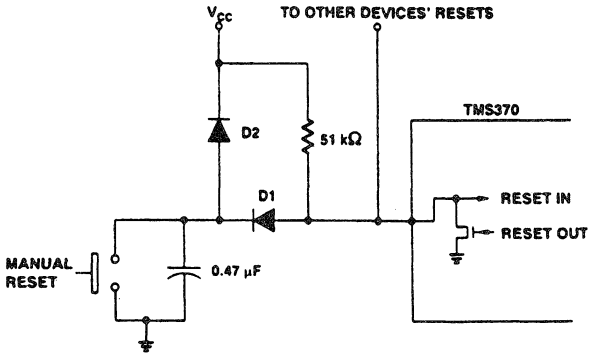


FIGURE 4. PROGRAMMER'S MODEL

system resets

The TMS370C732 has three possible reset sources: a low input to the $\overline{\text{RESET}}$ pin, a PACT watchdog timer timeout, or a programmable oscillator fault failure. The $\overline{\text{RESET}}$ pin, an input/output pin, initiates TMS370C732 hardware initialization and ensures an orderly software startup. A low level input of at least one clock cycle initiates the reset sequence. The microcontroller is held in reset until the $\overline{\text{RESET}}$ pin goes inactive (high). If the $\overline{\text{RESET}}$ input signal is low for less than eight system clock cycles, the TMS370C732 will hold the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The $\overline{\text{RESET}}$ pin must be activated by the application at powerup, which can be accomplished by an external input or an RC powerup reset circuit. Recall that the only operating mode, microcomputer, is determined by a low voltage level applied to the MC pin during reset.



NOTE: Low voltage detect circuit to protect against EEPROM corruption is shown in the *TMS370 Family Data Manual*.

FIGURE 5. TYPICAL RESET CIRCUIT

The watchdog timer provides system integrity by detecting a program that has become lost or is not executing as expected. A system reset is generated if the watchdog timer is not properly re-initialized by a specific software sequence, if the re-initialization does not occur before the watchdog timer times out. The watchdog timer timeout initiates the TMS370C732 reset sequence and drives the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. Watchdog control bits can be initialized only following a powerup reset. The timer section discusses additional information on the watchdog timer and its configurations.

The oscillator fault circuit provides the means to monitor failures of the oscillator input signal (XTAL2/CLKIN). This function may be disabled under software control by clearing the OSC FLT RST ENA bit of SCCR2. If the oscillator input signal frequency remains above the 90% point of the minimum operating frequency (CLKIN), the oscillator input is guaranteed not to be activated. However, if the oscillator input is lost or its frequency falls below 20 kHz and the oscillator fault reset is enabled, the TMS370C732 is reset and the external $\overline{\text{RESET}}$ pin is driven low.

When an oscillator input failure occurs, the internal clocks are stopped and $\overline{\text{RESET}}$ is held active until the oscillator input frequency is greater than 100 kHz typical (500 kHz maximum). During a HALT mode, the oscillator fault circuitry will be disabled.

After the occurrence of a reset, the program can interrogate the status bits (shown in Table 1) to determine the source, of the reset in order to take appropriate action. If none of the sources indicated in Table 1 caused the reset, then the $\overline{\text{RESET}}$ pin was pulled low by external hardware or the PACT module's watchdog.

TABLE 1. RESET SOURCES

REGISTER	ADDRESS	PF	BIT #	CONTROL BIT	SOURCE OF RESET
SCCR0	1010h	P010	7	COLD START	Cold start reset
SCCR0	1010h	P010	4	OSC FLT FLAG	Oscillator out of range

The $\overline{\text{RESET}}$ pin can be pulled low at any time during operation to start the reset sequence immediately. The sequence of events during reset is as follows:

1. Initialize CPU registers: ST = 00h, SP = 01h.
2. Initialize registers A and B to 00h (no other RAM is changed).
3. Read the contents of 7FFFh and store in the PC low byte (PCL).
4. Read the contents of 7FFEh and store in the PC high byte (PCH).
5. Start user program execution with an opcode fetch from the address pointed to by the PC.

The reset takes 21 cycles from the time $\overline{\text{RESET}}$ is released to the first opcode fetch in the microcomputer mode.

When the Watchdog overflow or the Oscillator Fault detection circuit generates a reset, the $\overline{\text{RESET}}$ pin is pulled low in order to reset other external components in the system.

During a reset, RAM contents (except for Register A and Register B) are unchanged and the majority of the peripheral file bits are set to 0 with the exception of the bits shown in Table 2.

TABLE 2. CONTROL-BIT STATES FOLLOWING RESET

REGISTER	CONTROL BIT	COLD RESET MICROCOMPUTER	WARM RESET MICROCOMUTER
SCCR0	COLD START OSC FLT FLAG	1 0	See Note 1. See Note 1.
SCCR2	OSC FLT RST ENA	1	1
ADSTAT	AD READY	1	1
PACT	PACT TXRDY	1	1

NOTE: State determined by cause of reset. See bit descriptions in the *TMS370 Family Data Manual*.

Interrupts

The TMS370 family software-programmable interrupt structure supports flexible on-chip and external interrupt configurations to meet realtime interrupt-driven application requirements. The hardware interrupt structure incorporates two priority levels as shown in Figure 6. Interrupt level 1 has a higher priority than interrupt level 2. The two priority levels can be independently masked by the global interrupt mask bits (IE1 and IE2) of the status register.

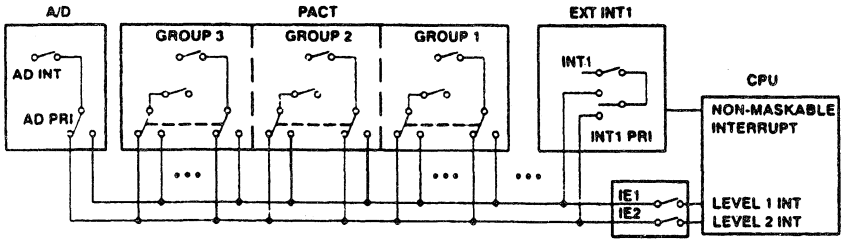


FIGURE 6. INTERRUPT CONTROL

Each system interrupt is independently configured on either the high or low priority chain by the application program during system initialization. Within each interrupt chain, the interrupt priority is fixed by the position of the module on the system interrupt chain. Arbitration between the two priority levels is performed within the CPU. Arbitration within each of the priority chains is performed within the peripheral modules to support interrupt expansion to future modules. Pending interrupts are serviced upon completion of current instruction execution, depending on their interrupt mask and priority conditions.

The TMS370C732 has 23 hardware system interrupts as shown in the Table 3. Each system interrupt has a dedicated interrupt vector located in program memory through which control is passed to the interrupt service routines. All of the interrupt sources are individually maskable by local interrupt enable control bits in the associated peripheral file. Each interrupt source flag bit is individually readable for software polling or to determine which interrupt source generated the associated system interrupt.

Nineteen of the system interrupts are generated by on-chip peripheral functions, and three external interrupts are supported. Software configuration of the external interrupts is performed through the INT1, INT2, and INT3 control registers in peripheral file frame 1. Each external interrupt is individually software configurable for input polarity (rising or falling) for ease of system interface. External interrupt INT1 is software configurable as either a maskable or nonmaskable interrupt. When INT1 is configured as nonmaskable, it cannot be masked by the individual or global mask bits. Recall that the INT1 NMI bit is protected during nonprivileged operation and therefore should be configured during the initialization sequence following reset. To maximize pin flexibility, external interrupts INT2 and INT3 can be software configured as general-purpose input/output pins if the interrupt function is not required (INT1 can be similarly configured as an input pin).

TABLE 3. INTERRUPT VECTOR SOURCES

MODULE	VECTOR ADDRESS	INTERRUPT SOURCE	INTERRUPT FLAG BIT	SYSTEM INTERRUPT	MODULE PRIORITY[1]	PRIORITY IN GROUP[1]
PACT	7F9Ch, 7F9Dh	PACT SCI TXINT	PACT TXRDY	PTXINT		2
(Group 2)	7F9Eh, 7F9Fh	PACT SCI RXINT	PACT RXRDY	PRXINT		1
PACT	7FA0h, 7FA1h	PACT Cmd/Def Entry 0	CMD/DEF INT 0 FLAG	CDINT0		1
(Group 3)	7FA2h, 7FA3h	PACT Cmd/Def Entry 1	CMD/DEF INT 1 FLAG	CDINT1		2
	7FA4h, 7FA5h	PACT Cmd/Def Entry 2	CMD/DEF INT 2 FLAG	CDINT2		3
	7FA6h, 7FA7h	PACT Cmd/Def Entry 3	CMD/DEF INT 3 FLAG	CDINT3	7	4
	7FA8h, 7FA9h	PACT Cmd/Def Entry 4	CMD/DEF INT 4 FLAG	CDINT4		5
	7FAAh, 7FABh	PACT Cmd/Def Entry 5	CMD/DEF INT 5 FLAG	CDINT5		6
	7FACh, 7FADh	PACT Cmd/Def Entry 6	CMD/DEF INT 6 FLAG	CDINT6		7
	7FAEh, 7FAFh	PACT Cmd/Def Entry 7	CMD/DEF INT 7 FLAG	CDINT7		8
PACT	7FB0h, 7FB1h	PACT Circular Buffer (Half/Full)	BUFFER HALF/FULL INT FLAG	BUFINT		1
(Group 1)	7FB2h, 7FB3h	PACT CP6 Edge	CP6 INT FLAG	CP6INT		2
	7FB4h, 7FB5h	PACT CP5 Edge	CP5 INT FLAG	CP5INT		3
	7FB6h, 7FB7h	PACT CP4 Edge	CP4 INT FLAG	CP4INT	5	4
	7FB8h, 7FB9h	PACT CP3 Edge	CP3 INT FLAG	CP3INT		5
	7FBAh, 7FBBh	PACT CP2 Edge	CP2 INT FLAG	CP2INT		6
	7FBCh, 7FBDh	PACT CP1 Edge	CP1 INT FLAG	CP1INT		7
	7FBEh, 7FBFh	PACT Default Timer Overflow	DEFTIM OVRFL INT FLAG	POVRFL INT		8
A/D	7FECh, 7FEDh	A/D Conversion Complete	AD INT FLAG	ADINT	0	1
INT 3	7FF8h, 7FF9h	External INT3	INT3 FLAG	INT3	4	1
INT 2	7FFAh, 7FFBh	External INT2	INT2 FLAG	INT2	3	1
INT 1	7FFCh, 7FFDh	External INT1	INT1 FLAG	INT1	2	1
RESET	7FFEh, 7FFFh	External RESET	COLD START	RESET	1	1
		Watchdog Overflow				
		Oscillator Fault Detect	OSC FLT FLAG			

[1] 1 is the highest priority.

NOTE The In-Circuit Emulator with PACT (PACT XDS/22) has the PACT module as the lowest priority interrupt, while the device family (TMS370Cx32) has the A/D interrupt at lowest priority.

privileged operation

The TMS370C732 is designed with significant flexibility to enable the designer to software-configure the system and with peripherals to meet the requirements of a broad variety of applications. The non-privileged mode of operation ensures the integrity of the system configuration once defined for an end application. Following a hardware reset, the TMS370C732 operates in the privileged mode, where all peripheral file registers have unrestricted read/write access and the application program will configure the system during the initialization sequence following reset. As the last step of system initialization, the PRIVILEGED DISABLE bit of SCCR2 will be set to 1, entering the non-privileged mode and disabling write operations to specific configuration control bits within the peripheral file. The following system configuration bits are write-protected during the non-privileged mode and must be configured by software prior to exiting the privileged mode.

TABLE 4. PRIVILEGED-MODE CONFIGURATION BITS

REGISTER	BIT
SCCR0	OSC POWER
SCCR1	MEMORY DISABLE AUTOWAIT DISABLE
SCCR2	PRIVILEGE DISABLE POWERDOWN/IDLE HALT/STANDBY INT NMI OSC FLT RST ENA
ADPRI	AD PRIORITY
PACTPRI	PACT GROUP 1 PRIORITY PACT GROUP 2 PRIORITY PACT GROUP 3 PRIORITY PACT STEST PACT MODE SELECT PACT WD PRESCALE SELECT 1 PACT WD PRESCALE SELECT 2
PACTSCR	FAST MODE SELECT PACT PRESCALE SELECT 3 PACT PRESCALE SELECT 2 PACT PRESCALE SELECT 1 PACT PRESCALE SELECT 0

The only way to change these bits after leaving the privileged mode is to reset the processor and then program the control registers. The write protect override (WPO) used for the EEPROM has no effect on the privileged bits.

oscillator fault

The processor contains a system of circuits to monitor the oscillator operation and to detect and contain major oscillator problems. This enhances processor and system reliability and aids in system recovery caused by a temporary fault. Programmable bits allow the user the option of incorporating or deleting some features of these circuits to match the application.

The circuit stops the processor whenever it detects an out of range oscillator operation. The Oscillator Fault Detection circuitry consists of:

1. Amplitude detector: Detects if the oscillator signal has a proper voltage level.
2. Frequency detector: Senses when the oscillator frequency goes too low. The oscillator fault detection circuit will always trigger below 20 kHz and never above 500 kHz.

The oscillator circuitry is designed to delay operation of the device until a stable clock signal is received. This protects the part against slow crystal startup times coming out of a halt mode or after an oscillator fault when the input clock may not be operating at the correct voltage range. The circuitry holds device operation until the input clock signal is within the required voltage range.

The Oscillator Fault Reset Enable bit (OSC FLT RST ENA) allows the user to determine what action the processor will take when the oscillator goes out of range. When active, the processor pulls the $\overline{\text{RESET}}$ pin low for at least eight cycles, causing external devices to reset along with the processor. When inactive, the processor enters a pseudo-halt state and waits for a reset.

The OSC FLT RST ENA bit defaults to the active state after a reset. This allows the processor to generate reset pulses until the oscillator operates within the correct range.

After reset, the program can check the Oscillator Fault flag (OSC FLT FLAG) along with the Cold Start flag to help determine the source of the reset. A reset does not clear these flags.

low-power operating modes

The STANDBY and HALT low-power modes significantly reduce power consumption by reducing or stopping the activity of the various on-chip peripherals when processing is not required. Each of the low-power modes is entered by executing the IDLE instruction when the POWERDOWN/IDLE bit in SCCR2 has been set to 1. The HALT/STANDBY bit in SCCR2 controls which low-power mode is entered.

In the STANDBY mode (HALT/STANDBY=0), all CPU activity and most peripheral module activity is stopped; however, the oscillator, internal clocks, the PACT counter, and the first PACT command entry remain active in all the modules. System processing is suspended until a qualified interrupt (hardware $\overline{\text{RESET}}$, external interrupt, or PACT interrupt) is detected.

The HALT MODE stops all internal clocks, which stops processing and provides the lowest power consumption.

The following information is preserved throughout both the STANDBY and HALT modes: RAM (register file), CPU registers (stack pointer, program counter, and status register), I/O pin direction and output data, and status registers of all on-chip peripheral functions. Since all CPU instruction processing is stopped during the STANDBY and HALT modes, the clocking of the watchdog timer is inhibited.

peripheral file frame 1

Peripheral File Frame 1 contains system configuration and control functions and registers for controlling EEPROM programming. The privileged bits are shaded in the Peripheral File Frames.

PERIPHERAL FILE FRAME 1: SYSTEM CONFIGURATION AND CONTROL REGISTERS

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1010h	P010	COLD START	OSC POWER	PF AUTO WAIT	OSC FLT FLAG	MC PIN WPO	MC PIN DATA	—	UP/$\overline{\mu}$C MODE	SCCR0
1011h	P011	—	—	—	AUTOWAIT DISABLE	—	MEMORY DISABLE	—	—	SCCR1
1012h	P012	HALT/ STANDBY	PWRDWN/ IDLE	OSC FLT RST ENA	BUS STEST	CPU STEST	OSC FLT DISABLE	INT1 NMI	PRIVILEGE DISABLE	SCCR2
1013h to 1016h	P013 to P016	RESERVED								
1017h	P017	INT1 FLAG	INT1 PIN DATA	—	—	—	INT1 POLARITY	INT1 PRIORITY	INT1 ENABLE	INT1
1018h	P018	INT2 FLAG	INT2 PIN DATA	—	INT2 DATA DIR	INT2 DATA OUT	INT2 POLARITY	INT2 PRIORITY	INT2 ENABLE	INT2
1019h	P019	INT3 FLAG	INT3 PIN DATA	—	INT3 DATA DIR	INT3 DATA OUT	INT3 POLARITY	INT3 PRIORITY	INT3 ENABLE	INT3
101Ah	P01A	BUSY	—	—	—	—	AP	W1W0	EXE	DEECTL
101Bh	P01B	RESERVED								
101Ch										EPCTL (TMS370C732)
101Dh to 101Fh	P01D to P01F	RESERVED								

Peripheral File Frame 2 contains the digital I/O pin configuration and control registers. The following figure details the specific addresses, registers, and control bits within this Peripheral File Frame.

PERIPHERAL FILE FRAME 2: DIGITAL PORT CONTROL REGISTERS

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1020h	P020	RESERVED								APORT1
1021h	P021	PORT A CONTROL REGISTER 2 (must be 0)								APORT2
1022h	P022	PORT A DATA								ADATA
1023h	P023	PORT A DIRECTION								ADIR
1024h to 102Bh	P024 to P02B	RESERVED								
102Ch	P02C	0	0	0	0	X	0	0	0	DPORT1
102Dh	P02D	0	0	0	0	X	0	0	0	DPORT2
102Eh	P02E	X	X	X[2]	X	X	X[2]	X[2]	X[2]	DDATA
102Fh	P02F	X[1]	X[1]	0[2]	X[1]	X	0[2]	0[2]	0[2]	DDIR

[1] Output may cause PACT capture
[2] This port pin not available: read data is indeterminate

PORT CONFIGURATION REGISTERS SET-UP

a = PORTX CONTROL REGISTER 1
b = PORT X CONTROL REGISTER 2
c = DATA
d = DIRECTION

PORT	PIN	abcd 00x0	abcd 00q1	abcd 01xx
A	0 — 7	Data in	Out q	Do not use
D	4,6,7	Data in	Out q	Do not use
D	3	Data in	Out q	CLKOUT

NOTES 1 Each bit controls the corresponding pin; for example, bit 6 controls Port pin 6. Each pin is individually configurable.
2 Only register combination 00xx is defined for TMS370C732, except for pin D3, which can be configured as CLKOUT.

PACT MODULE

features

Input capture functions on six input pins, including four pins (CP3 to CP6) with a programmable prescaler.

Timer-driven outputs on eight pins.

Configurable timer overflow rates for different functions.

One 8-bit event counter driven by CP6.

Up to 20-bit timer capability.

Interaction between event counter and timer activity.

Register-based organization, allowing single-cycle accesses to parameters.

18 independent interrupt vectors with two priority levels.

Integrated, configurable Watchdog with selectable time-out period.

Mini Serial Communications Interface with independent setup of baud rate for receive and transmit lines.

overview

A simplified functional block diagram of the PACT module is shown in Figure 7. Each of these blocks will be discussed in the following pages.

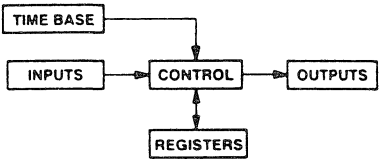


FIGURE 7. PACT BLOCK DIAGRAM

time base

The time base section of PACT is very similar to that used in traditional timers. The microcontroller system clock is routed to a prescaler that feeds a hardware counter. As seen in Figure 8, the prescale section consists of a 4-bit prescaler and an optional divide-by-8 circuit. The hardware counter is 20 bits wide.

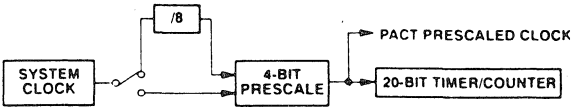


FIGURE 8. TIME BASE BLOCK DIAGRAM

inputs

Figure 9 shows a block diagram of the input capture section of PACT. PACT has six input capture pins. How these pins are used depends on the PACT operating mode. In mode A, two pins cause captures to dedicated capture registers and the other four cause captures to a circular buffer. In mode B, four pins cause captures to dedicated capture registers and the other two cause captures to the circular buffer. All input pin captures are 32 bits long and consist of the 20-bit hardware timer, the 8-bit event counter, and 4 extra bits used to identify the pin that caused the capture in the circular buffer. Captures can be set to occur on the falling, rising, or both edges of the input signal. Input pin CP6 also runs the 8-bit event counter.

Capture pins CP3 through CP6 may be prescaled with a divide value from 1 to 8. Each of these four pins has its own edge counter, but the maximum count value (1– 8) before an actual capture occurs must be the same for all four pins.

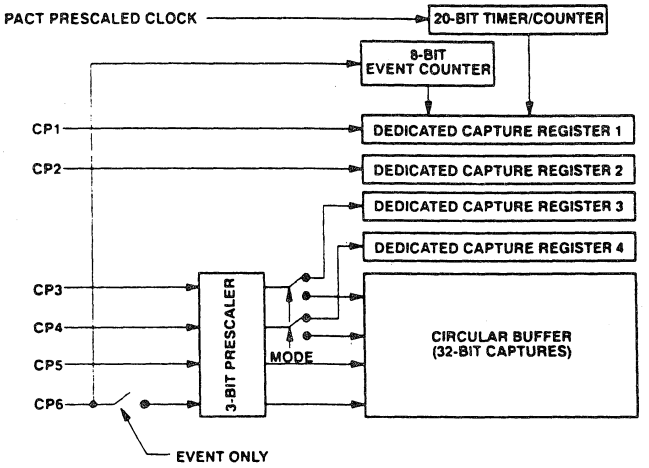


FIGURE 9. INPUT CAPTURE BLOCK DIAGRAM

memory organization

One of the major differences between the PACT module and standard timers is the location of the capture registers. The capture registers are actually locations in a dual-port RAM. These locations can be read or written by the CPU. They are automatically written to by PACT when the appropriate feature is enabled. The dual-ported RAM is 128 bytes long as shown in Figure 10.

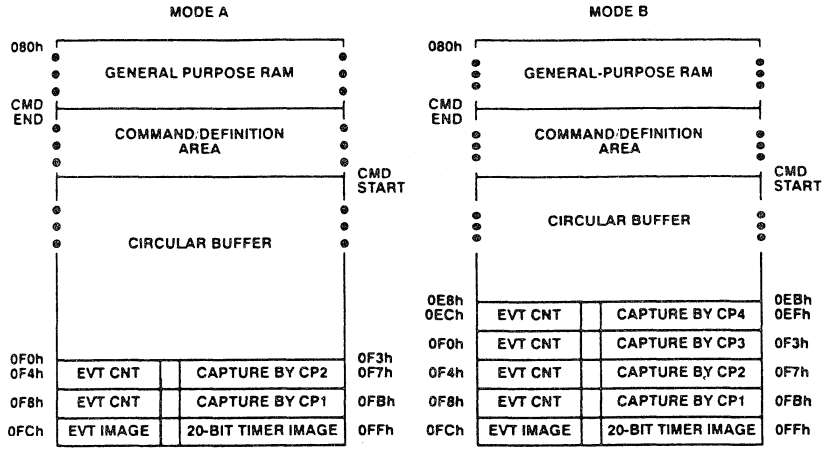


FIGURE 10. DUAL-PORTED RAM ORGANIZATION

The architecture of PACT does not care where the dual-port RAM is located in the memory map, but by convention it is located in the register file from 0080h to 00FFh. This allows maximum access speed to these registers by the CPU when using the register address mode. The locations 0FCh – 0FFh of this dual-port memory contain an image of the 20-bit default time, and an image of the 8-bit event counter. Since they are images, or more precisely, copies of the actual hardware registers, they can be over written by the application software. However, they will be rewritten every time the PACT module receives another prescaled clock. The 32-bit capture registers are directly before the timer and event counter images.The circular buffer is directly before the capture registers. The length of the circular buffer is defined by the user. Five bits in the peripheral file define the start of the Command/Definition area which in effect also defines the size of the circular buffer.

Since it takes several system clock periods for the CPU to read a 20-bit timer capture value, it is possible for an additional capture to occur while reading the original capture. The user's program can detect this situation by clearing the capture flag in the peripheral file before reading the capture value and then verifying that the flag has not been set again after the read is complete. If the flag was set again, the value read may be invalid and should be reread.

The memory map in Figure 11 is a typical implementation of PACT. Note that there are three areas of memory that are used by the PACT module. The 128 bytes of dual-port RAM contain the capture registers, the circular buffer and a command/definition area. The peripheral frame contains the hardware registers used for initial setup. The interrupt vectors must be setup by the programmer.

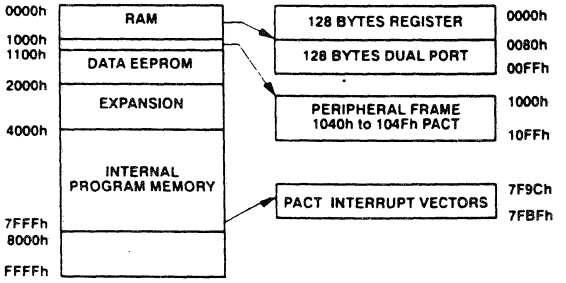


FIGURE 11. RAM ORGANIZATION

control and outputs

The control and output section of PACT is the most powerful part of this timer. The controller acts like a state machine. The controller is started when it receives a rising edge from the PACT prescaled clock. The controller reads its commands (or state microcode) from the command definition part of the dual-port RAM. The 8-bit event counter and the 16 least significant bits of the 20-bit default counter are also input into the controller for use in comparisons. The outputs from the controller are used to set or clear the 8 output pins. The prescaled clock from the PACT time base is only used to start the controller. The controller steps through its commands as fast as it can, using the system clock phases for synchronization. The controller must step through all of the commands in the command/definition area before the next rising edge of the prescaled clock. The next prescaled clock increments the 20-bit default counter and starts the whole process over.

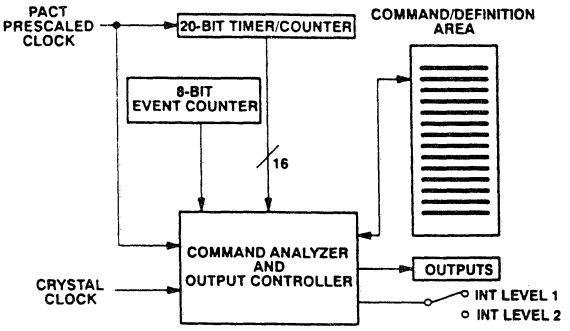


FIGURE 12. OUTPUT CONTROL SECTION

standard compare command

All commands or definitions in the command/definition area are 32 bits long. The simplest command is the standard compare command. The purpose of the standard compare command is to set or clear an output pin whenever the timer/counter is equal to a certain value. As seen in Figure 13, the standard compare command is made up of a 16-bit compare value, 3 bits to select one of the 8 output pins, some bits to select what action to take, and a few bits to distinguish this command from the others.

The possible actions that this command can cause are:

- Set or clear the chosen output pin when the counter matches the compare value,
- Do the opposite action (clear or set) when the 16 least significant bits of the counter are equal to zero,
- Generate an interrupt when the compare value is reached.

A block diagram of the standard compare command is shown below. This diagram shows the information contained in the command. For more information or actual bit definitions refer to *TMS370 Family Data Manual*.

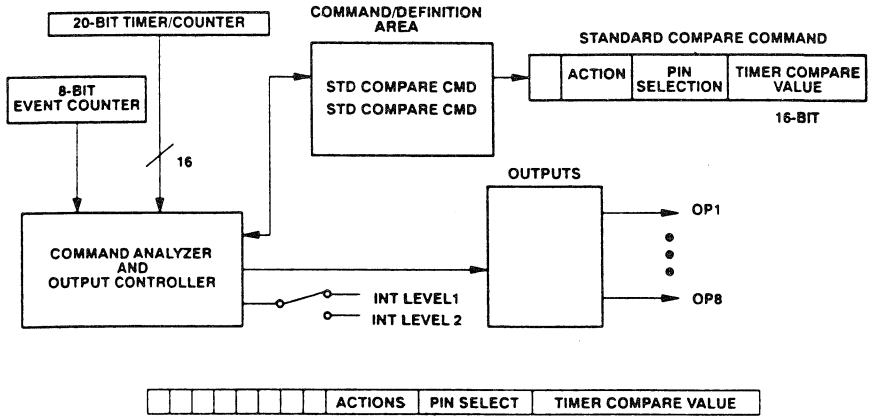
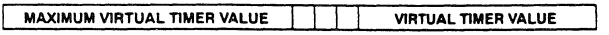


FIGURE 13. STANDARD COMPARE COMMAND

With a single standard compare command we can make a simple pulse width modulated output. Assume that we want a PWM output with an initial duty cycle of 75%. Using one standard compare command, set the timer compare value to 04000h (1/4 the overflow rate), set the actions to cause an output pin to go high when the count is equal to the compare value and then low again when the least significant16 bits of the counter are zero. The duty cycle can be varied by changing the 16-bit compare value. The signal can be inverted by selecting clear on compare equal, as opposed to set on compare equal.

virtual timers

The way to vary the period of the PWM is with the use of a virtual timer. Keep in mind that the command/definition area is implemented in RAM. Figure 14 shows the virtual timer definition and its implementation. The virtual timer definition consists of 16 bits that are read, incremented and rewritten on each tic of the PACT clock. Also within the virtual timer definition are 13 bits that define a maximum value. When the virtual timer reaches this maximum value it is reset to zero. A block diagram of the virtual timer definition is shown below. This diagram shows the information contained in the definition.



The command/definition area of Figure 14 shows two standard compare commands, a virtual timer definition and a third standard compare command. Assume that we are using a 200-ns internal system clock and we are prescaling the PACT clock with divide by five so that each PACT clock tic is one microsecond. The first two standard compare commands are used to generate PWM signals of variable duty cycle with a period of 65,536 prescaled clock tics (65.536 ms). If we want the third PWM to have a period of one millisecond, then we set up the virtual timer with a maximum value of one thousand. When the controller sees the timer definition, it will increment the virtual timer and then use the virtual timer value for future comparisons. The third standard compare command will generate a PWM of variable duty cycle with a period of one millisecond.

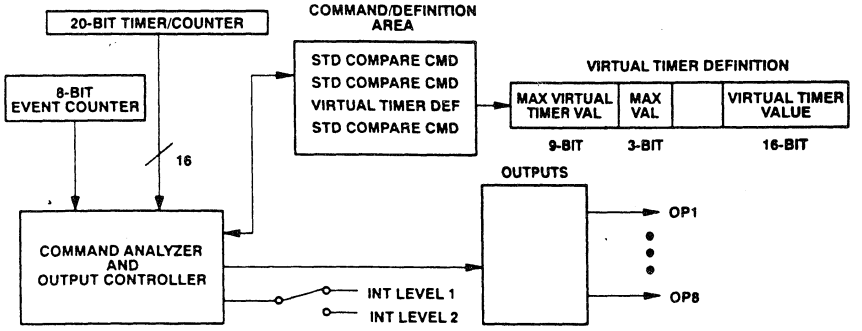


FIGURE 14. VIRTUAL TIMER IMPLEMENTATION

Combinations of standard compare commands and virtual timers can be used to create complex repeating waveforms. Multiple standard compare commands can be used on a single output pin to create multiple pulses of different duration.

Virtual timers are also used to provide periodic interrupts to the processor.

double event compare command

Actions can also be taken based on comparisons to the 8-bit event counter. Since all commands are 32 bits wide, the double event compare command actually defines two event compare values and the actions that can be performed, based on each value. The allowed actions based on event compare 1 matching the event counter are:

- Set or reset the selected output pin (OP1– 8)
- Interrupt
- Generate a 32-bit capture into the circular buffer

The allowed actions based on event compare 2 matching the event counter are:

- Set or reset the selected output pin (OP1– 8)
- Interrupt
- Generate a 32-bit capture into the circular buffer
- Reset the 20-bit default timer

Because of synchronization, these actions will occur two or three prescaled clock cycles after the input edge that incremented the event counter. A block diagram of the double event command is shown below. This diagram shows the information contained in the command.

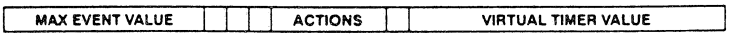
So far we can manipulate output lines based on time values or based on the number of external events. There is an additional virtual timer definition that allows us to manipulate output lines based on a combination of the event counter and time.

offset timer definition–time from last event

The offset timer definition–time from last event creates a 16-bit virtual timer that is cleared on each occurence of an event on pin CP6. This definition also sets an event counter maximum, so that the event counter is reset after reaching this maximum value. The offset timer definition may generate the following actions:

- Generate an interrupt when the maximum event count is reached,
- Store the 16-bit virtual timer in the circular buffer on each event,
- Store the 20-bit default timer and 8-bit event counter in the circular buffer when the maximum event count is reached,
- Reset the 20-bit hardware default timer when the maximum event count is reached.

A block diagram of the offset timer definition is shown below. This diagram shows the information contained in the command.

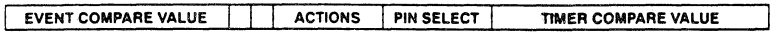


conditional compare command

There is also a special compare command that has a timer compare value and an event compare value. Both of these values must match for the defined action to take place. Usually, a series of these commands follows an offset timer definition–time from last event and provides output pulses on different pins, based on the event count and an elapsed time from the event. The actions that may be generated by the conditional compare command are:

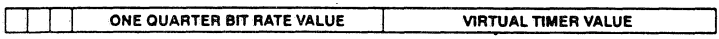
- Generate an interrupt when both conditions are met:
  1. The event compare value equals the event counter.
  2 The timer compare value equals the last defined timer.
- Set or clear one of seven output pins (OP1– OP7) when the following two conditions are met:
  1. The event compare value equals the event counter.
  2 The timer compare value equals the last defined timer.

The same actions described above may be enabled on the event counter reaching the event compare value plus one without regard to the timer compare value. This allows for the case where the next event occurs before the delay period specified by the timer compare value was reached. A block diagram of the conditional compare command is shown below. This diagram shows the information contained in the command.



baud rate timer definition

The last item that can be put into the command/definition area of the PACT module is a baud rate virtual timer. This virtual timer runs the serial communications port built into the PACT module. Set up the maximum timer value for one-quarter bit period of the desired baud rate. Separate timers can be defined for transmit and receive. For more information on the PACT SCI see the *TMS370 Family Data Manual*.



watchdog

At power-up the Watchdog is always enabled with the lower time-out period (bit 9 of default timer).

A Watchdog originated reset is normally generated when a software-selected bit of the default timer toggles. There are three options that determine the Watchdog time-out period plus a disable watchdog code:

WD2	WD1	OPTIONS
0	0	Reset when Bit 9 of default timer toggles
0	1	Reset when Bit 15 of default timer toggles
1	0	Reset when Bit 19 of default timer toggles
1	1	Disable Watchdog

These bits are resident in peripheral frame 4 at address 104Fh. They are available to the programmer only in privileged mode immediately after powerup.

Once a period has been selected by the above means, the alternate key bytes, 55h and AAh (55h first) must be written to location 104Eh (peripheral frame 4) to avoid a Watchdog-originated reset being issued. The only exception to this occurs when the default counter is being cleared by the PACT module. In this case a Watchdog-originated reset will occur whenever the default timer is cleared, unless the correct keyword (55h/AAh) has been written since the previous clear.

The Watchdog is stopped in standby mode and halt mode.

serial communications interface (SCI)

The SCI works as a simplified full duplex UART, transmitting 8-bit words with a fixed format of one start and one stop bit. If parity transmission is required, the parity bit must be calculated by the CPU and placed in the transmit buffer as part of the 8-bit word. Parity reception is facilitated by the parity result bit. This bit allows the processor to check for parity errors by comparing the bit (SCICTLP.5) against 0 or 1 for even or odd parity. Hence, there is no parity error bit to be checked by the processor.

There is no overrun detection. The PACT SCI has a shift register and a buffer register. This gives the program a full data byte reception time to read the previous byte before it is overwritten.

During reception the start bit is detected on the falling edge and then sampled again in the center of the bit to avoid false detection. All other bits are sampled once at their centers. If at least one stop bit is not detected when it is expected, the framing error flag (P045.3) is set. This bit will remain set until cleared by reset, SCI software reset, or by writing a zero to it.

The software selectable baud rates (RX and TX) may be different. Receive and transmit operations may be stopped or started by using the control bits within the baud rate definition command.

The data being received and transmitted is accessed in the same peripheral frame (4) as are the control bits. Received data is held at RXBUFP (1046h); transmitted data, at TXBUFP (1047h).

PACT block diagram

Figure 15 is a block of the PACT module. It shows the major functional blocks, inputs, and outputs. This module is controlled not only by the peripheral file but also by the defined areas/contents of the dual-port RAM. The dual-port RAM is located between 0080h and 00FFh (inclusive) on the memory map.

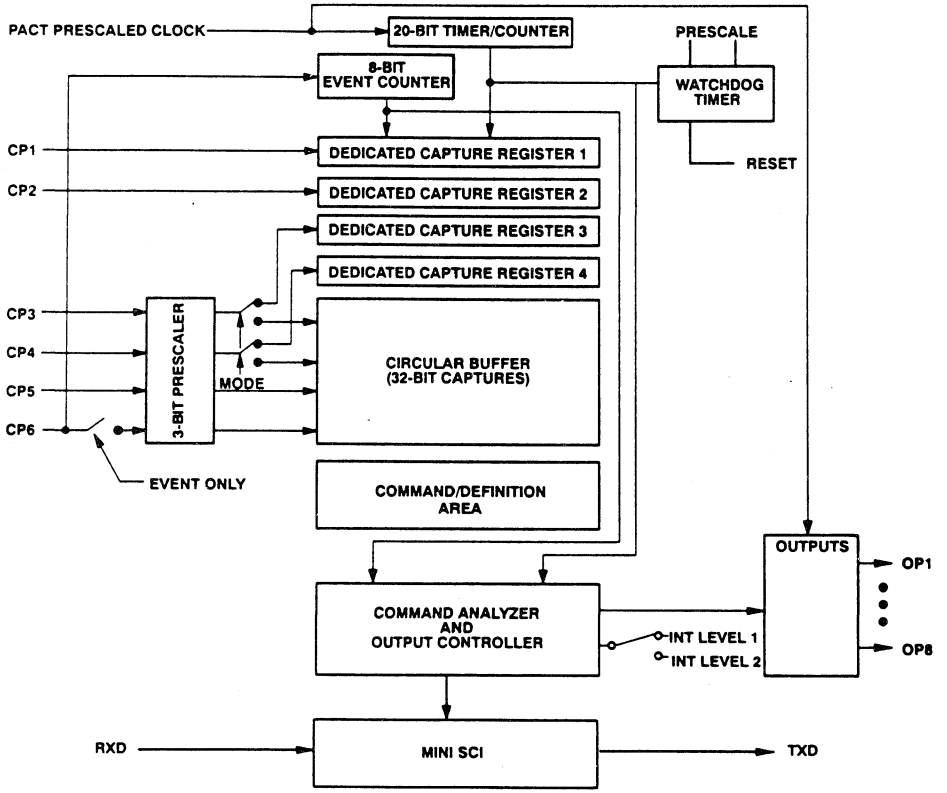


FIGURE 15. PACT BLOCK DIAGRAM

PERIPHERAL FILE FRAME 4: PACT CONTROL REGISTER

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1040h	P040	DEFTIM OVRFL INT ENA	DEFTIM OVRFL INT FLAG	CMD/DEF AREA ENA	FAST MODE SELECT	PACT PRESCALE SELECT 3	PACT PRESCALE SELECT 2	PACT PRESCALE SELECT 1	PACT PRESCALE SELECT 0	PACTSCR
1041h	P041	CMD/DEF AREA INT ENA	---	CMD/DEF AREA START BIT 5	CMD/DEF AREA START BIT 4	CMD/DEF AREA START BIT 3	CMD/DEF AREA START BIT 2	---	---	CDSTART
1042h	P042	---	CMD/DEF AREA END BIT 6	CMD/DEF AREA END BIT 5	CMD/DEF AREA END BIT 4	CMD/DEF AREA END BIT 3	CMD/DEF AREA END BIT 2	---	---	CDEND
1043h	P043	---	---	BUFFER POINTER BIT 5	BUFFER POINTER BIT 4	BUFFER POINTER BIT 3	BUFFER POINTER BIT 2	BUFFER POINTER BIT 1	---	BUFPTR
1044h	P044	RESERVED								
1045h	P045	PACT RXRDY	PACT TXRDY	PACT PARITY	PACT FE	PACT SCI RX INT ENA	PACT SCI TX INT ENA	---	PACT SCI SW RESET	SCICTLP
1046h	P046	PACT RXDT7	PACT RXDT6	PACT RXDT5	PACT RXDT4	PACT RXDT3	PACT RXDT2	PACT RXDT1	PACT RXDT0	RXBUFP
1047h	P047	PACT TXDT7	PACT TXDT6	PACT TXDT5	PACT TXDT4	PACT TXDT3	PACT TXDT2	PACT TXDT1	PACT TXDT0	TXBUFP
1048h	P048	PACT OP8 STATE	PACT OP7 STATE	PACT OP6 STATE	PACT OP5 STATE	PACT OP4 STATE	PACT OP3 STATE	PACT OP2 STATE	PACT OP1 STATE	OPSTATE
1049h	P049	CMD/DEF INT 7 FLAG	CMD/DEF INT 6 FLAG	CMD/DEF INT 5 FLAG	CMD/DEF INT 4 FLAG	CMD/DEF INT 3 FLAG	CMD/DEF INT 2 FLAG	CMD/DEF INT 1 FLAG	CMD/DEF INT 0 FLAG	CDFLAGS
104Ah	P04A	CP2 INT ENA	CP2 INT FLAG	CP2 CAPT RISING EDGE	CP2 CAPT FALLING EDGE	CP1 INT ENA	CP1 INT FLAG	CP1 CAPT RISING EDGE	CP1 CAPT FALLING EDGE	CPCTL1
104Bh	P04B	CP4 INT ENA	CP4 INT FLAG	CP4 CAPT RISING EDGE	CP4 CAPT FALLING EDGE	CP3 INT ENA	CP3 INT FLAG	CP3 CAPT RISING EDGE	CP3 CAPT FALLING EDGE	CPCTL2
104Ch	P04C	CP6 INT ENA	CP6 INT FLAG	CP6 CAPT RISING EDGE	CP6 CAPT FALLING EDGE	CP5 INT ENA	CP5 INT FLAG	CP5 CAPT RISING EDGE	CP5 CAPT FALLING EDGE	CPCTL3
104Dh	P04D	BUFFER HALF/ FULL INT ENA	BUFFER HALF/ FULL INT FLAG	INPUT CAPT PRESCALE SELECT 3	INPUT CAPT PRESCALE SELECT 2	INPUT CAPT PRESCALE SELECT 1	CP6 EVENT ONLY	EVENT COUNTER SW RESET	OP SET / CLR SELECT	CPPRE
104Eh	P04E	WATCHDOG RESET KEY								WDRST
104Fh	P04F	PACT STEST	---	PACT GROUP PRIORITY	PACT GROUP 2 PRIORITY	PACT GROUP 3 PRIORITY	PACT MODE SELECT	PACT WD PRESCALE SELECT 1	PACT WD PRESCALE SELECT 0	PACTPRI

analog-to-digital converter

The 8-bit analog-to-digital (A/D) converter provides the designer with eight multiplexed analog input channels. The A/D converter has internal sample and hold circuitry and uses a successive approximation conversion technique. The accuracy of the A/D conversion process is increased by providing separate analog position supply and analog ground input pins (V_{CC3} and V_{SS3}). The V_{SS3} pin also provides the low reference voltage input for the conversion process.

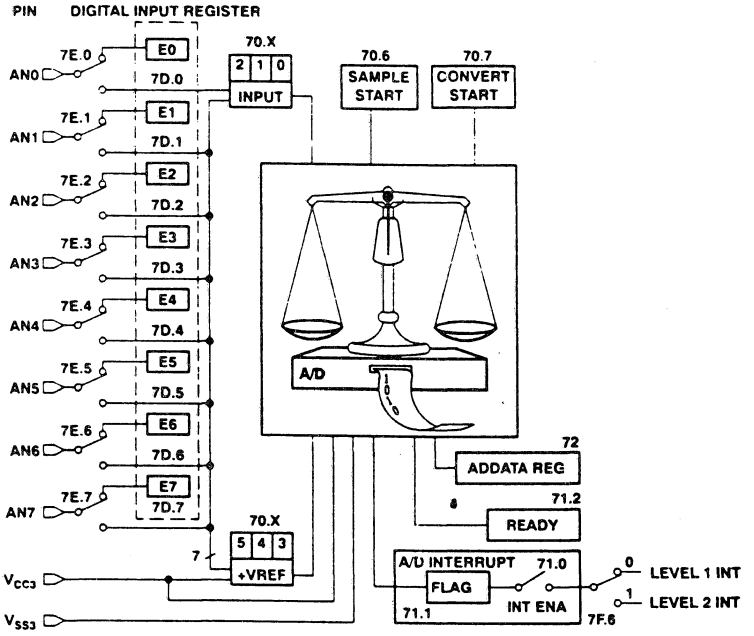


FIGURE 16. A/D CONVERTER BLOCK DIAGRAM

The A/D converter high reference voltage input is software selectable as one of eight positive reference inputs, as shown in the following table. The A/D conversion process is ratiometric, using V_{SS3} and the software-selected high-reference voltage input as the limits for the selected analog input channel. An input voltage equal to or greater than the high reference input converts to FFh (full scale) with no overflow. An input voltage equal to or less than V_{SS3} converts to 00h. Ratiometric conversions allow analog inputs to be scaled against selected high reference inputs to achieve the greatest accuracy.

A/D INPUT			ANALOG INPUT CHANNEL
SEL2	SEL1	SEL0	
0	0	0	AN0
0	0	1	AN1
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4
1	0	1	AN5
1	1	0	AN6
1	1	1	AN7

REFERENCE VOLTAGE			HIGH REFERENCE INPUT
SEL2	SEL1	SEL0	
0	0	0	V_{CC3}
0	0	1	AN1
0	1	0	AN2
0	1	0	AN3
1	0	0	AN4
1	0	1	AN5
1	1	0	AN6
1	1	1	AN7

To read an A/D channel:

1. Write to the ADCTL peripheral file control register to:
   a. Select the high reference voltage input (ADCTL.5-3).
   b. Select the analog input channel for conversion (ADCTL.2-0).
   c. Set the SAMPLE START bit to 1 (ADCTL.6).
2. Wait for the sample time to elapse.
3. Set the CONVERT START bit (ADCTL.7) and leave SAMPLE START bit (ADCTL.6) set.
4. Wait for either the interrupt flag to be set or the A/D interrupt to occur.
5. Read the conversion value form ADDATA when AD INT FLAG is set to 1 or the A/D interrupt occurs.
6. Clear the interrupt flag (ADSTAT.1).

To provide the designer with the flexibility to optimize the A/D conversion process with both high- and low-impedance sources, the sample time is independently defined by the application program. At the completion of the sample time, the conversion is initiated by setting the CONVERT START and SAMPLE START bits to 1. Eighteen clock cycles after the CONVERT START bit is set to 1, the CONVERT START and SAMPLE START bits will both be set to 0 by the A/D converter, indicating the conversion has started and the analog input signal can be removed. The AD READY bit is set to 0 by the A/D converter to indicate a conversion is in progress. The conversion is complete 164 system clock cycles after it is initiated by setting the CONVERT START bit to 1, and the result is located in the ADDATA result register. Upon completion of the conversion, the AD INT FLAG bit is set, and if the AD INT ENA bit is set to 1, an interrupt will be asserted.

The A/D converter has eight bits of resolution with absolute accuracy of plus or minus one LSB, with (High Reference Voltage – V_{SS3}) = 5 V. This translates to an absolute accuracy of 20 mV.

To maximize I/O control capability, all analog input pins not used for an analog input or high reference voltage input may be individually configured as general-purpose digital input pins. The control and input data values are contained in the ADENA and ADIN peripheral file control registers.

PERIPHERAL FILE FRAME 7: A-TO-D CONTROL REGISTERS

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1070h	70	CONVERT START	SAMPLE START	REF VOLT SELECT 2	REF VOLT SELECT 1	REF VOLT SELECT 0	AD INPUT SELECT 2	AD INPUT SELECT 1	AD INPUT SELECT 0	ADCTL
1071h	71	---	---	---	---	---	AD READY	AD INT FLAG	AD INT ENA	ADSTAT
1072h	72	A-TO-D CONVERSION DATA REGISTER								ADDATA
1073h to 107Ch	73 to 7C	RESERVED								
107Dh	7D	PORT E DATA INPUT REGISTER								ADIN
107Eh	7E	PORT E DATA ENABLE REGISTER								ADENA
107Fh	7F	AD STEST	AD PRIORITY	AD ESPEN	---	---	---	---	---	ADPRI

instruction set

The TMS370 family instruction set consists of 73 instructions that control input, output, data manipulations, data comparisons, and program flow. The instruction set is supported with 14 addresssing modes to provide the flexibility to optimize programs to the user's applications. For example, the MOV instruction has 27 operand combinations supported by its addressing modes.

ADDRESSING MODE	EXAMPLE		OPERATION
GENERAL			
Implied	LDSP		(B) → (SP)
Register	MOV	R5,R4	(0005) → (0004)
Peripheral	MOV	P025,A	(1025) → A
Immediate	ADD	#123,R23	123 + (03) → (PC)
PC Relative	JMP	offset	PCN + offset → (PC)
Stack Pointer Relative	MOV	2(SP),(A)	(2 + (SP)) → (A)
EXTENDED			
Absolute Direct	MOV	A1234	(A) → (1234)
Absolute Indexed	MOV	1234(B),A	(1234) + (B)) → (A)
Absolute Indirect	MOV	@R4,A	((R3:R4)) → (A)
Absolute Offset Indirect	MOV	12(R4),A	(12 + (R3:R4)) → (A)
Relative Direct	JMPL	1234	PCN + 1234 → (PC)
Relative Indexed	JMPL	1234(B)	PCN + 1234 + (B) → (PC)
Relative Indirect	JMPL	@R4	PCN + (R3:R4) → (PC)
Relative Offset Indirect	JMPL	12(R4)	PCN + 12 + (R3:R4) → (PC)

PCN = 16-bit address of next instruction.
(x) = Contents of memory at address x.
((x)) = Contents of memory location designated by contents at address x

The CPU controls instruction execution by executing microinstructions from a dedicated control memory at a rate of one microinstruction per internal system clock cycle, t_c. The number of system clock cycles required to execute one assembly language instruction varies depending on the instruction complexity, operand addressing mode, and number of wait states. Instruction execution times are stated in terms of the number of integral system clock cycles per instruction. Instruction execution times vary from 5 to 63 internal system clock cycles, with most instructions requiring less than 10 cycles to complete.

Similarly, the number of bytes of program memory required to store an instruction will vary with instruction complexity and addressing mode. TMS370 instructions require from one to five bytes of program memory space, with most instructions occupying one or two bytes.

SYMBOL	DEFINITION	SYMBOL	DEFINITION
s	Source Operand	d/D	Destination Operand (8-bit/16-bit)
A	Register A or R0 in Register File	B	Register B or R1 in Register File
Rs	Source Register in Register File	Rd	Destination Register in Register File
Ps	Source Register in Peripheral File	Pd	Destination Register in Peripheral File
Rps	Source Register Pair (Rn, Rn-1)	Rpd	Destination Register Pair (Rn, Rn-1)
Rp	General Purpose Register Pair	label	16-bit Label
iop8	8-bit Immediate Operand	iop16	16-bit Immediate Operand
off8	8-bit Signed Offset (label – PCN)	off16	16-bit Signed Offset
PC	Program Counter	PCN	16-bit Address of Next Instruction
SP	Stack Pointer	ST	Status Register
#	Immediate Operand	@	Indirect Addressing Operand
C	Status Register Carry Bit	XADDR	16-bit Address
( )	Contents of	→	Is Assigned To

TABLE 5. TMS370 FAMILY INSTRUCTION OVERVIEW

MNEMONIC		OPCODE	BYTES	CYCLES t_c	STATUS C N Z V	OPERATION DESCRIPTION
ADC	B,A	69	1	8	x x x x	(s) + (d) + (C_I) → (d)
	Rs,A	19	2	7		Add the source, destination, and carry bit together.
	Rs,B	39	2	7		Store at the destination address.
	Rs,Rd	49	3	9		
	#iop8,A	29	2	6		
	#iop8,B	59	2	6		
	#iop8,Rd	79	3	8		
ADD	B,A	68	1	8	x x x x	(s) + (d) → (d)
	Rs,A	18	2	7		Add the source and destination operands at the destination address.
	Rs,B	38	2	7		
	Rs,Rd	48	3	9		
	#iop8,A	28	2	6		
	#iop8,B	58	2	6		
	#iop8,Rd	78	3	8		
AND	A,Pd	83	2	9	0 x x 0	(s) AND (d) → (d)
	B,A	63	1	8		AND the source and destination operands together and store at the destination address.
	B,Pd	93	2	9		
	Rs,A	13	2	7		
	Rs,B	33	2	7		
	Rs,Rd	43	3	9		
	#iop8,A	23	2	6		
	#iop8,B	53	2	6		
	#iop8,Rd	73	3	8		
	#iop8,Pd	A3	3	10		
BR	label	8C	3	9	- - - -	XADDR → (PC)
	@Rp	9C	2	8		Branch to the destination address.
	label(B)	AC	3	11		
	off8(Rp)	F4 EC	4	16		
BTJO	A,Pd,off8	86	3	10	0 x x 0	If (s) AND (d) ≠ 0,
See Note 3	B,A,off8	66	2	10		then PCN + offset → (PC)
	B,Pd,off8	96	3	10		If the AND of the source and destination operands ≠ 0 (corresponding 1 bits) the PC will add the offset, and the jump will be taken.
	Rs,A,off8	16	3	9		
	Rs,B,off8	36	3	9		
	Rs,Rd,off8	46	4	11		
	#iop8,A,off8	26	3	8		
	#iop8,B,off8	56	3	8		
	#iop8,Rd,off8	76	4	10		
	#iop8,Pd,off8	A6	4	11		
BTJZ	A,Pd,off8	87	3	10	0 x x 0	If (s) AND (not d) ≠ 0
See Note 3	B,A,off8	67	2	10		then (PCN) + offset → (PC)
	B,Pd,off8	97	3	10		If any 1 in the source corresponds to a 0 in the destination, the PC adds the offset and the jump is taken.
	Rs,A,off8	17	3	9		
	Rs,B,off8	37	3	9		
	Rs,Rd,off8	47	4	11		
	#iop8,A,off8	27	3	8		
	#iop8,B,off8	57	3	8		
	#iop8,Rd,off8	77	4	10		
	#iop8,Pd,off8	A7	4	11		
CALL	label	8E	3	13	- - - -	Push PC MSB, PC LSB,
	@Rp	9E	2	12		XADDR → (PC)
	label(B)	AE	3	15		
	off8(Rp)	F4 EE	4	20		

NOTE 3: Add two to the cycle count if a jump is taken.

Legend:

0 Status Bit always cleared.
1 Status Bit always set.
x Status Bit cleared or set on results.
- Status Bit not affected.

TABLE 5. TMS370 FAMILY INSTRUCTION OVERVIEW (CONTINUED)

MNEMONIC		OPCODE	BYTES	CYCLES t_c	STATUS C N Z V	OPERATION DESCRIPTION
CALLR	label	8F	3	15	- - - -	Call Relative Push PC MSB, PC LSB, PCN + (XADDR) → (PC)
	@Rp	9F	2	14		
	label(B)	AF	3	17		
	off8(Rp)	F4 EF	4	22		
CLR	A	B5	1	8	0 0 1 0	0 → (d) Clear the destination operand.
	B	C5	1	8		
	Rd	D5	2	6		
CLRC		B0	1	9	0 x x 0	0 → (C) Clears the carry bit. N and Z bits set on the result of A
CMP	label,A	8D	3	11	x x x x	Compare; (d) – (s) computed. Set flags on the result of the source operand subtracted from the destination operand. Operands are not affected by operation.
	@Rp,A	9D	2	10		
	label(B),A	AD	3	13		
	off8(Rp),A	F4 ED	4	18		
	off8(SP),A	F3	2	8		
	B,A	6D	1	8		
	Rs,A	1D	2	7		
	Rs,B	3D	2	7		
	Rs,Rd	4D	3	9		
	#iop8,A	2D	2	6		
	#iop8,B	5D	2	6		
	#iop,Rd	7D	3	8		
CMPBIT	Rname	75	3	8	0 x x 0	Complement Bit; invert the bit
	Pname	A5	3	10		
COMPL	A	BB	1	8	x x x 0	Two's complement; 00h – (s) → (d)
	B	CB	1	8		
	Rd	DB	2	6		
DAC	B,A	6E	1	10	x x x x	(s) + (d) + (C) → (d) (BCD) The source, destination, and the carry bit are added, and the BCD sum is stored at the destination address.
	Rs,A	1E	2	9		
	Rs,B	3E	2	9		
	Rs,Rd	4E	3	11		
	#iop8,A	2E	2	8		
	#iop8,B	5E	2	8		
	#iop8,Rd	7E	3	10		
DEC	A	B2	1	8	x x x x	(d) – 1 → (d) Decrement destination operand by 1.
	B	C2	1	8		
	Rd	D2	2	6		
DINT		F0 00	2	6	0 0 0 0	0 → (ST)(global interrupt enable bits) 0 → IE1, 0 → IE2.
DIV	Rs,A	F4 F8	3	55— 63 14	0 x x 0 1 1 1 1	A:B/Rs → A(= quotient),B(= remainder) Integer divide, 16 by 8 bit. Overflow detected.
DJNZ See Note 3	A,off8	BA	2	10	- - - -	(d) – 1 → (d), If (d) ≠ 0, then PCN + offset → (PC) Decrement and jump if not 0.
	B,off8	CA	2	10		
	Rd off8	DA	3	8		
DSB	B,A	6F	1	10	x x x x	(d) – (s) – 1 + (C) → (d) (BCD) The source operand is subtracted from the destination; this sum is then reduced by 1 and the carry bit is then added to it. The result is stored as a BCD number.
	Rs,A	1F	2	9		
	Rs,B	3F	2	9		
	Rs,Rd	4F	3	11		
	#iop8,A	2F	2	8		
	#iop8,B	5F	2	8		
	#iop8,Rd	7F	3	10		

NOTE 3 Add two to the cycle count if a jump is taken

Legend:

0 Status Bit always cleared
1 Status Bit always set
x Status Bit cleared or set on results
- Status Bit not affected

TABLE 5. TMS370 FAMILY INSTRUCTION OVERVIEW (CONTINUED)

MNEMONIC		OPCODE	BYTES	CYCLES t_c	STATUS C N Z V	OPERATION DESCRIPTION
EINT		F0 0C	2	6	0 0 0 0	0Ch → (ST)(global interrupt enable bit) 1 → IE1, 1 → IE2.
EINTH		F0 04	2	6	0 0 0 0	04h → (ST)(high priority global interrupt enable bit) 1 → IE1, 0 → IE2
EINTL		F0 08	2	6	0 0 0 0	08h → (ST)(low priority global interrupt enable bit) 0 → IE1, 1 → IE2
IDLE		F6	1	6	- - - -	(PC) → (PC) until interrupt (PC) + 1 → (PC) after return from interrupt. Stops µC execution until an interrupt. Entry to low power modes.
INC	A B Rd	B3 C3 D3	1 1 2	8 8 6	x x x x	(d) + 1 → (d) Increase the destination operand by 1.
INCW	#off8,Rp	70	3	11	x x x x	(Rp) + offset → (Rp) Add 8-bit signed offset to register pair.
INV	A B Rd	B4 C4 D4	1 1 2	8 8 6	0 x x 0	NOT(d) → (d) 1's complement the destination operand.
JBIT0 (1)	Rd,off8 Pd,off8	77 A7	4 4	10 11	0 x x 0	Jump If Bit = 0
JBIT1 (1)	Rd,off8 Pd,off8	76 A6	4 4	10 11	0 x x 0	Jump If Bit = 1
JMP	off8	00	2	7	- - - -	PCN + off8 → (PC) Jump unconditionally using an 8-bit offset.
JMPL	label @Rp label (B) off8(Rp)	89 99 A9 F4 E9	3 2 3 4	9 8 11 16	- - - -	PCN + D → (PC) Jump unconditionally using a 16-bit offset.
Jcnd See Note 3	JC JEQ JG JGE JHS JL JLE JLO JN JNC JNE JNV JNZ JP JPZ JV JZ	03 02 0E 0D 0B 09 0A 0F 01 07 06 0C 06 04 05 08 02	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	- - - -	Conditional jump on: Carry Jump Equal Greater Than, signed Greater Than or Equal, signed Higher or Same, unsigned Less Than, signed Less Than or Equal, signed Lower Value, unsigned Negative, signed No Carry Jump Not Equal No Overflow, signed Not Zero Positive, signed Positive or Zero, signed Overflow, signed Zero
LDSP		FD	1	7	- - - -	(B) → (SP) Load stack pointer with contents of register B.
LDST	#iop8	F0	2	6	x x x x	(s) → (ST) Load ST Register.

NOTE 3: Add two to the cycle count if a jump is taken

Legend:

0 Status Bit always cleared.
1 Status Bit always set.
x Status Bit cleared or set on results.
- Status Bit not affected.

TABLE 5. TMS370 FAMILY INSTRUCTION OVERVIEW (CONTINUED)

MNEMONIC		OPCODE	BYTES	CYCLES t_c	STATUS C N Z V	OPERATION DESCRIPTION
MOV	A,B	C0	1	9	0 x x 0	(s) → (d) Replace the destination operand with the source operand.
	A,Rd	D0	2	7		
	A,Pd	21	2	8		
	A,label	8B	3	10		
	A,@Rp	9B	2	9		
	A,label(B)	AB	3	12		
	A,off8(Rp)	F4 EB	4	17		
	A,off8(SP)	F2	2	7		
	Rs,A	12	2	7		
	Rs,B	32	2	7		
	label,A	8A	3	10		
	@Rp,A	9A	2	9		
	label(B),A	AA	3	12		
	off8(Rp),A	F4 EA	4	17		
	off8(SP),A	F1	2	7		
	B,A	62	1	8		
	B,Rd	D1	2	7		
	B,Pd	51	2	8		
	Rs,Rd	42	3	9		
	Rs,Pd	71	3	10		
	Ps,A	80	2	8		
	Ps,B	91	2	8		
	Ps,Rd	A2	3	10		
	#iop8,A	22	2	6		
	#iop8,B	52	2	6		
	#iop8,Rd	72	3	8		
	#iop8,Pd	F7	3	10		
MOVW	Rps,Rpd	98	3	12	0 x x 0	(s) → (Rd–1:Rd) Copy the source register word to the destination register pair.
	#iop16,Rpd	88	4	13		
	#iop16(B),Rpd	A8	4	15		
	off8(Rs),Rpd	F4 E8	5	20		
MPY	B,A	6C	1	47	0 x x 0	(s) x (d) → (A:B) Multiply the source and destination operands; store the result in Registers A (MSB) and B (LSB).
	Rs,A	1C	2	46		
	Rs,B	3C	2	46		
	Rs,Rd	4C	3	48		
	#iop8,A	2C	2	45		
	#iop8,B	5C	2	45		
	#iop8,Rn	7C	3	47		
NOP		FF	1	7	- - - -	No operation
OR	A,Pd	84	2	9	0 x x 0	(s) OR (d) → (d) Logically OR the source and destination operands, and store the results at the destination address.
	B,A	64	1	8		
	B,Pd	94	2	9		
	Rs,A	14	2	7		
	Rs,B	34	2	7		
	Rs,Rd	44	3	9		
	#iop8,A	24	2	6		
	#iop8,B	54	2	6		
	#iop8,Rd	74	3	8		
	#iop8,Pd	A4	3	10		
POP	A	B9	1	9	0 x x 0	((SP)) → (d) (SP) –1 → (SP)
	B	C9	1	9		
	Rd	D9	2	7		
	ST	FC	1	8	x x x x	

NOTE 3: Add two to the cycle count if a jump is taken.

Legend:

0 Status Bit always cleared.
1 Status Bit always set.
x Status Bit cleared or set on results.
– Status Bit not affected.

TABLE 5. TMS370 FAMILY INSTRUCTION OVERVIEW (CONTINUED)

MNEMONIC		OPCODE	BYTES	CYCLES t_c	STATUS C N Z V	OPERATION DESCRIPTION
PUSH	A	B8	1	9	0 x x 0	(SP) + 1 → (SP) (s) → ((SP))
	B	C6	1	9		Copy the operand onto the stack
	Rd	D8	2	7		Copy the status register onto the stack
	ST	FB	1	8		
RL	A	BE	1	8	x x x 0	Bit(n) → Bit(n + 1) Bit(7) → Bit(0) and Carry
	B	CE	1	8		
	Rd	DE	2	6		
RLC	A	BF	1	8	x x x 0	Bit(n) → Bit(n + 1) Carry → Bit(0) Bit(7) → Carry
	B	CF	1	8		
	Rd	DF	2	6		
RR	A	BC	1	8	x x x 0	Bit(n + 1) → Bit(n) Bit(0) → Bit(7) and Carry
	B	CC	1	8		
	Rd	DC	2	6		
RRC	A	BD	1	8	x x x 0	Bit(n + 1) → Bit(n) Carry → Bit(7) Bit(0) → Carry
	B	CD	1	8		
	Rd	DD	2	6		
RTI		FA	1	12	x x x x	Pop PCL, PCH, POP ST Return From Interrupt
RTS		F9	1	9	- - - -	Pop PCL, PCH
SBB	B,A	6B	1	8	x x x x	(d) – (s) – 1 + (C) → (d) Subtract with borrow Destination minus source minus 1 plus carry; stored at the destination address.
	Rs,A	1B	2	7		
	Rs,B	3B	2	7		
	Rs,Rd	4B	3	9		
	#iop8,A	2B	2	6		
	#iop8,B	5B	2	6		
	#iop8,Rd	7B	3	8		
SBIT0	Rd	73	3	8	0 x x 0	Set Bit to 0
	Pd	A3	3	10		
SBIT1	Rd	74	3	8	0 x x 0	Set Bit to 1
	Pd	A4	3	10		
SETC		F8	1	7	1 0 1 0	Axh → (ST) Set the carry bit. IE1 and IE2 unchanged.
STSP		FE	1	8	- - - -	(SP) → (B) Copy the SP into Register B.
SUB	B,A	6A	1	8	x x x x	(d) – (s) → (d) Store the destination operand minus the source operand at the destination address.
	Rs,A	1A	2	7		
	Rs,B	3A	2	7		
	Rs,Rd	4A	3	9		
	#iop8,A	2A	2	6		
	#iop8,B	5A	2	6		
	#iop8,Rd	7A	3	8		
SWAP	A	B7	1	11	0 x x 0	s(7-4,3-0) → d(3-0,7-4) Swap the operand's high and low nibbles.
	B	C7	1	11		
	Rd	D7	2	9		
TRAP	n	EF–E0	1	14	- - - -	Vector n → (PC), n = 0 → 15 Trap to Subroutine; Push PCN. Trap 0 = EF.
TST	A	B0	1	9	0 x x 0	Test; Set flags from register.
	B	C6	1	10		

OTE 3: Add two to the cycle count if a jump is taken.

:gend:

Status Bit always cleared.
Status Bit always set.
Status Bit cleared or set on results.
Status Bit not affected.

TABLE 5. TMS370 FAMILY INSTRUCTION OVERVIEW (CONCLUDED)

MNEMONIC		OPCODE	BYTES	CYCLES t_c	STATUS C N Z V	OPERATION DESCRIPTION
XCHB	A	B6	1	10	0 x x 0	(B) ←→ (Rn) Swap the contents of Register B with (Rn).
	B	C6	1	10		
	Rd	D6	2	8		
XOR	A,Pd	85	2	9	0 x x 0	(s) XOR (d) → (d) Logically exclusive-OR the source and destination operands, store at the destination address.
	B,A	65	1	8		
	B,Pd	95	2	9		
	Rs,A	15	2	7		
	Rs,B	35	2	7		
	Rs,Rd	45	3	9		
	#iop8,A	25	2	6		
	#iop8,B	55	2	6		
	#iop8,Rd	75	3	8		
	#iop8,Pd	A5	3	10		

Legend:
0 Status Bit always cleared.
1 Status Bit always set.
x Status Bit cleared or set on results.
– Status Bit not affected.

development system support

The TMS370 family development support tools include an Assembler, a Linker, an In-Circuit emulator (XDS – eXtended Development Support), and an EEPROM/UVEPROM programmer that supports standard EPROM and all TMS370 devices. All of the tools work closely together using any MS ™ DOS-basedPC.

- Assembler/Linker (Part Number TMDS3740810-02 (DOS)/TMDS3740210-08 (VMS))
  - — Extensive macro capability.
  - — High-speed operation.
  - — Format conversion utilities available for popular formats.
- PACT XDS/22 (eXtended Development Support) In-Circuit Emulator (Part Number TMDS3762211)
  - — PC-based, window/function-key-oriented user interface for ease of use and a rapid learning environment.
  - — Sophisticated breakpoint and trace support for software debugging and software/hardware integration. Provides 2047 qualified trace samples, up to 32 breakpoints, sequential breakpoint triggering, and counters.
  - — Symbolic debugging.
  - — Provides a realtime testbed for software development before target circuitry is available.
  - — Timing analysis to evaluate and enhance system performance. Provides identification of critical routines and instructions with two timers controlled by the full capabilities of the breakpoint qualification support.
  - — Disassembly of code and trace history, including symbols.
  - — Eight-line logic probe to expand the breakpoint and trace visibility.
  - — Logic analyzer output interface to support circuit-level debugging.
- EEPROM/UVEPROM Programmer (Part Number TMDS3760510)
  - — PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
  - — Supports all TMS370 series devices to provide rapid target prototyping capability.
  - — Also programs TMS2732, TMS2764, TMS27128, and TMS27256 NMOS and CMOS EPROMs.
  - — Plug-in personality boards and loadable parameters to support future packages and devices.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V_{CC} (See Note 4) ..... - 0.3 to 7 V
Input voltage ranage, All pins except MC ..... 0.3 to V_{CC} + 0.3 V
MC ..... 0.3 to 14 V
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{CC}$) ..... ±20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{CC}$) ..... ±20 mA
Continuous output current per buffer, I_O (V_O = 0 to V_{CC}) ..... ±10 mA
Maximum source current, I_{CC} ..... 170 mA
Maximum drain current, I_{SS} ..... 170 mA
Continuous power dissipation ..... 800 mW
Storage temperature range ..... – 65°C to 150 °C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 4: All voltage values are with respect to V_{SS}.

recommended operating conditions

			MIN	NOM	MAX	UNIT
V_{CC1}	Supply voltage (see Note 5)		4.5	5	5.5	V
V_{CC1}	RAM data-retention supply voltage (see Note 6)		3		5.5	V
V_{CC3}	Analog supply voltage (see Note 5)		4.5	5	5.5	V
V_{IL}	Low-level input voltage	All pins except MC	V_{SS}		0.8	V
		MC, normal operation	V_{SS}		0.3	
V_{IH}	High-level input voltage	All pins except MC, XTAL2/CLKIN, and $\overline{RESET}$	2		V_{CC}	V
		MC/Write Protect Override (WPO)	11.7		13	
		XTAL2/CLKIN	0.8 V_{CC}		V_{CC}	
		$\overline{RESET}$	0.7 V_{CC}		V_{CC}	
	Mode control voltage on MC pin (see Note 7)	EEPROM write protect override (WPO)	11.7	12	13	V
		Microcomputer	V_{SS}		0.3	
		EPROM programming voltage (V_{PP})	12	12.5	13	
T_A	Operating free-air temperature	A version	– 40		85	°C
		L version	0		70	°C

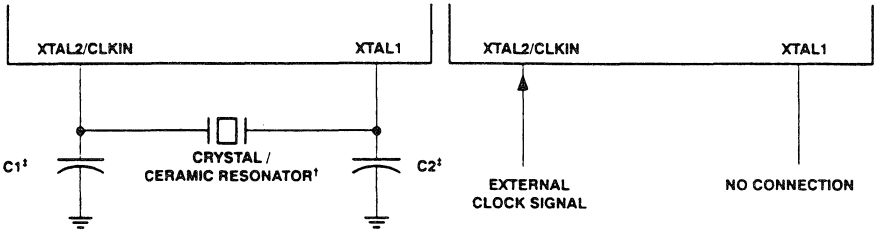
NOTES: 5. Unless otherwise noted, all voltages are with respect to V_{SS}.
6. To guarantee RAM data-retention from 3.0 V to 4.5 V, $\overline{RESET}$ must be externally asserted and released only while V_{CC} is within the recommended operating range of 4.5 V to 5.5 V.
7. The WPO mode may be selected anytime a sufficient voltage is present on the MC pin.

electrical characteristics over recommended operating free–air temperature range

PARAMETER			TEST CONDITIONS		MIN	NOM	MAX	UNIT
V_{OL}	Low–level output voltage		I_{OL} = –1.0 mA				0.4	V
V_{OH}	High–level output voltage		I_{OH} = –50 µA		0.9 V_{CC}			V
			I_{OH} = –2 mA		2.4			V
I_I	Input current	MC pin	0 V ≤ V_I ≤ 0.3 V				10	µA
			0.3 V ≤ V_I ≤ 13 V				650	µA
			12 V ≤ V_I ≤ 13 V				50	µA
		I/O pins	0 V ≤ V_I ≤ V_{CC}				±10	µA
I_{OL}	Low–level output current		V_{OL} = 0.9 V_{CC}		1.0			mA
I_{OH}	High–level output current		V_{OH} = 0.9 V_{CC}		–50			µA
			V_{OH} = 2.4 V		–2			mA
I_{CC}	Supply current (operating mode) Osc Power bit = 0		Operating mode (notes 8 & 9)	20 MHz		35	45	mA
				12 MHz		25	35	mA
				2 MHz		10	14	mA
I_{CC}	Supply current (standby mode) Osc Power bit = 0		Standby mode (notes 8 & 9)	20 MHz		12	17	mA
				12 MHz		8	13	mA
				2 MHz		3	4	mA
I_{CC}	Supply current (halt mode)		Halt mode (note 8) CLKIN 0.2 V			15	40	µA

† Input current Ipp will be of maximum of 50 mA only when programming EPROM.

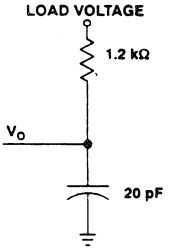
NOTES: 8. Single chip mode, ports configured as inputs, or outputs with no load. All inputs ≤ 0.2 V or ≥ V_{CC} – 0.2 V.
9. XTAL2/CLKIN is driven with an external square wave signal with 50% duty cycle and rise and fall times less than 10 ns. Currents may be higher with a crystal oscillator. At 20 MHz this extra current = 0.01 mA × (total load capacitance + crystal capacitance in pF).



† The crystal/ceramic resonator frequency is four times the reciprocal of the system clock period.
‡ The values of C1 and C2 are typically 15 pF. See manufacturer's recommendations for ceramic resonators.

FIGURE 17. RECOMMENDED CRYSTAL/CLOCK CONNECTIONS

PARAMETER MEASUREMENT INFORMATION

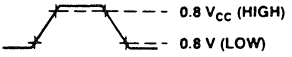


Case 1: $V_O = V_{OH} = 2.4$ V; Load Voltage = 0 V
Case 2: $V_O = V_{OL} = 0.4$ V; Load Voltage = 2.1 V

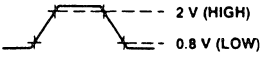
§ All measurements are made with the pin loading as shown unless otherwise noted. All measurements are made with XTAL2/CLKIN driven by an external square wave signal with a 50% duty cycle and rise and fall times less than 10 ns unless otherwise stated.

FIGURE 18. TYPICAL OUTPUT LOAD CIRCUIT§

All timings are measured between high and low measurement points as indicated in the figures below.



XTAL2/CLKIN MEASUREMENT POINTS



GENERAL MEASUREMENT POINTS

timing parameter symbology

Timing parameter symbols have been created in accordance with JEDEC Standard 100. In order to shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

AR	Array	CO	CLKOUT
B	Byte	PGM	Program
CI	XTAL2/CLKIN		

Lowercase subscripts and their meanings are:

c	cycle time (period)	r	rise time
d	delay time	su	setup time
f	fall time	v	valid time
h	hold time	w	pulse duration (width)

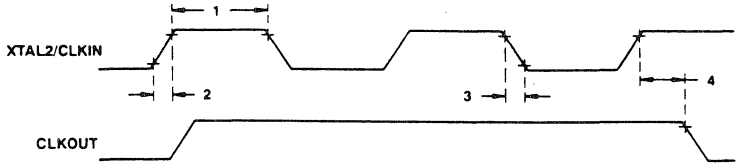
The following additional letters are used with these meanings:

H	High	V	Valid
L	Low		

external clocking requirements

NO.			MIN	NOM	MAX	UNIT
1	$t_{w(CI)}$	XTAL2/CLKIN pulse duration (see Note 10)	20			ns
2	$t_{r(CI)}$	XTAL2/CLKIN rise time			30	ns
3	$t_{f(CI)}$	XTAL2/CLKIN fall time			30	ns
	CLKIN	Crystal operating frequency	2		20	MHz

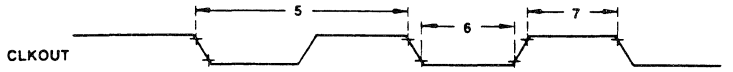
NOTE 10: This pulse may be either a high pulse, as illustrated, which extends from the earliest valid high to the final valid high in an XTAL2/CLKIN cycle, or a low pulse, which extends from the earliest valid low to the final valid low in an XTAL2/CLKIN cycle.



switching characteristics and timing requirements ‡

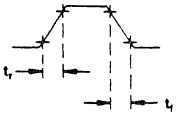
NO.	PARAMETER		MIN	MAX	UNIT
4	$t_{d(CIH-COL)}$	Delay time, XTAL2/CLKIN rise to CLKOUT fall		100	ns
5	t_c	CLKOUT (system clock) cycle time	200	2000	ns
6	$t_{w(COL)}$	CLKOUT low pulse duration	$0.5t_c$-20	$0.5t_c$	ns
7	$t_{w(COH)}$	CLKOUT high pulse duration	$0.5t_c$	$0.5t_c$+20	ns

‡ t_c = system clock cycle time = 4/CLKIN.



general purpose output signal switching timing requirements

PARAMETER			MIN	NOM	MAX	UNIT
t_r	Rise time	INT2, INT3			45	ns
t_f	Fall time	INT2, INT3			45	ns



recommended EPROM operating conditions for programming

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.75		5.5	V
V_{PP}	Supply voltage at MC pin	12	12.5	13	V
I_{PP}	Supply current at MC pin during programming (V_{PP} = 13 V)		30	50	mA
CLKIN	Operating crystal frequency	2		20	MHz

recommended EPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(IEPGM)}$	Initial programming pulse (see Note 11)	0.95	1	1.05	ms
$t_{w(FEPGM)}$	Final programming pulse	2.85		78.75	ms

NOTE 11: Programming pulse is active when both EXE (EPCTL.0) and V_{PPS} (EPCTL.6) are set.

recommended EEPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(PGM)B}$	Programming duration to insure valid data is stored (byte mode)	10			ms
$t_{w(PGM)AR}$	Programming duration to insure valid data is stored (array mode)	20			ms

A/D converter

The A/D Converter has a separate power bus for its analog circuitry. These pins are referred to as V_{CC3} and V_{SS3}. The purpose is to enhance A/D performance by preventing digital switching noise of the logic circuitry which may be present on V_{SS} and V_{CC} from coupling into the A/D analog stage. All A/D specifications will be given with respect to V_{SS3} unless otherwise noted.

Resolution ..... 8 bits (256 values)
Monotonic ..... Yes
Output conversion code ..... 00h to FFh (00 for $V_I \leq V_{SS3}$; FF for $VI \geq V_{ref}$)
Conversion time (excluding sample time) ..... $164t_c$

recommended operating conditions

		MIN	NOM	MAX	Unit
V_{CC3}	Analog supply voltage	4.5	5	5.5	V
		$V_{CC}-0.3$		$V_{CC}+0.3$	V
V_{SS3}	Analog ground	$V_{SS}-0.3$		$V_{SS}+0.3$	V
V_{ref}	Non-V_{CC3} reference (see Note 13)	2.5	V_{CC3}	$V_{CC3}+0.1$	V
	Analog input for conversion	V_{SS3}		V_{ref}	V

NOTE 12: V_{ref} must be stable, within ± 1/2 LSB of the required resolution, during the entire conversion time.

operating characteristics over full ranges of recommended operating conditions

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
Absolute accuracy (see Note 10)		V_{CC3} = 5.5 V, V_{ref} = 5.1 V			±1	LSB
Differential/integral linearity error (see Notes 13 and 14)		V_{CC3} = 5.5 V, V_{ref} = 5.1 V			±0.5	LSB
I_{CC3}	Analog supply current	Converting			2	mA
		Non-converting			5	µA
I_I	Inputcurrent,AN0 — AN7	$0 V \leq V_I \leq 5.5 V$			2	µA
V_{ref}	Input charge current				1	mA
Z_{ref}	Source impedance of V_{ref}	XTAL2/CLKIN ≤ 12 MHz			24	kΩ
		12 MHz < XTAL2/CLKIN ≤ 20 MHz			10	kΩ

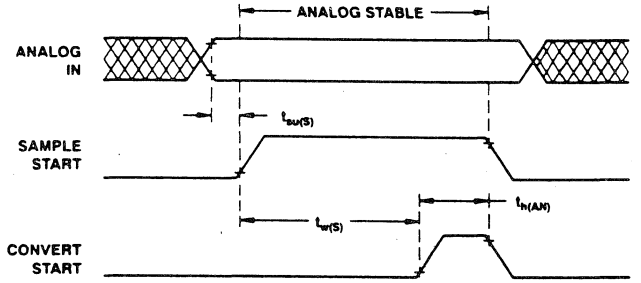
NOTES: 13. Absolute resolution = 20 mV. At V_{ref} = 5 V, this is 1 LSB. As V_{ref} decreases, LSB size decreases and thus absolute accuracy and differential/integral linearity errors in terms of LSBs increases.
14 Excluding quantization error of 1/2 LSB.

The A/D module allows complete freedom in design of the sources for the analog inputs. The period of the sample time is user-defined such that high impedance sources can be accommodated without penalty to low-impedance sources. The sample period begins when the SAMPLE START bit of the A/D Control Register (ADCTL) is set to 1. The end of the signal sample period occurs when the conversion bit (CONVERT START) of the ADCTL is set to 1. After a hold time, the converter will reset the SAMPLE START and CONVERT START bits, signaling that a conversion has started and the analog signal can be removed.

analog timing requirements

		MIN	NOM	MAX	UNIT
$t_{su(S)}$	Analog input setup to sample command	0			ns
$t_{h(AN)}$	Analog input hold from start of conversion	$18t_c$			ns
$t_{w(S)}$	Duration of sample time per kilohm of source impedance (see Note 15)	1			µs/kΩ

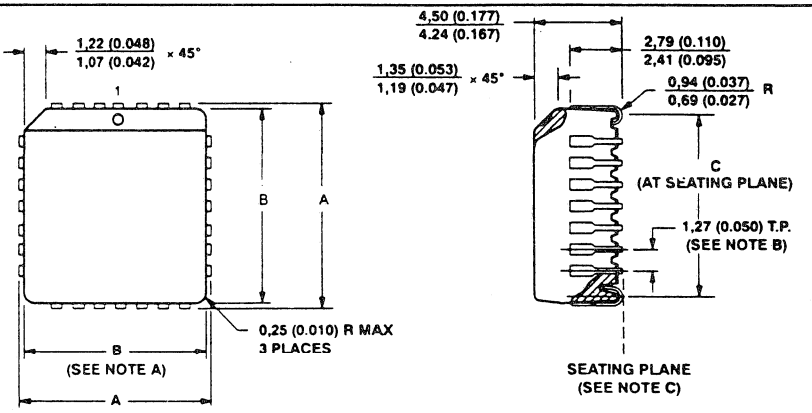
NOTE 15. The value given is valid for a signal with a source impedance greater than 1 kΩ. If the source impedance is less than 1 kΩ, use a minimum sampling time of 1 µs.



MECHANICAL DATA

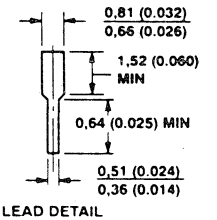
plastic-leaded chip carrier package (FN suffix)

Each of these chip carrier packages consists of a circuit mounted on a lead frame and encapsulated within an electrically nonconductive plastic compound. The compound withstands soldering temperatures with no deformation, and circuit performance characteristics remain stable when the devices are operated in high-humidity conditions. The packages are intended for surface mounting on solder lands on 1,27 (0.050) centers. Leads require no additional cleaning or processing when used in soldered assembly.



JEDEC OUTLINE	NO OF TERMINALS	A MIN	A MAX	B MIN	B MAX	C MIN	C MAX
MO-047AA	20	9,78 (0.385)	10,03 (0.395)	8,89 (0.350)	9,04 (0.356)	7,87 (0.310)	8,38 (0.330)
MO-047AB	28	12,32 (0.485)	12,57 (0.495)	11,43 (0.450)	11,58 (0.456)	10,41 (0.410)	10,92 (0.430)
MO-047AC	44	17,40 (0.685)	17,65 (0.695)	16,51 (0.650)	16,66 (0.656)	15,49 (0.610)	16,00 (0.630)
MO-047AE	68	25,02 (0.985)	25,27 (0.995)	24,13 (0.950)	24,33 (0.956)	23,11 (0.910)	23,62 (0.930)
MO-047AF	84	30,10 (1.185)	30,35 (1.195)	29,41 (0.150)	21,41 (1.158)	27,69 (0.090)	28,70 (1.130)

All dimensions and notes for the specified JEDEC outline apply



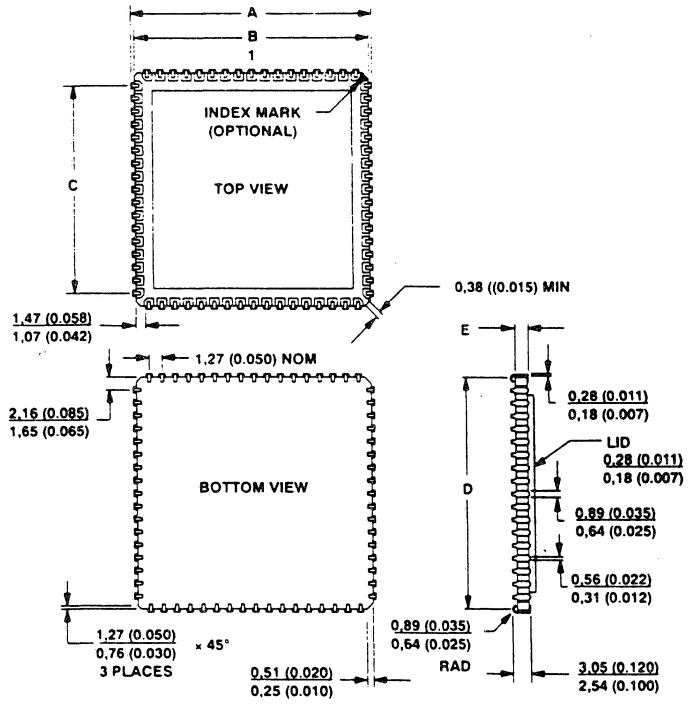
LEAD DETAIL

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTES A Centerline of center pin each side is within 0,10 (0.004) of package centerline as determined by dimension B.
B Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
C The lead contact points are planar within 0,10 (0.004).

MECHANICA DATA

ceramic leaded chip carrier package (FJ suffix)



DIM / PINS	A		B		C		D		E	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
44	17,27 (0.680)	17,78 (0.700)	16,28 (0.641)	16,74 (0.659)	12,57 (0.495)	12,88 (0.507)	15,49 (0.610)	16,51 (0.650)	1,91 (0.075)	2,16 (0.085)
68	24,89 (0.980)	25,40 (0.100)	23,88 (0.940)	24,51 (0.965)	20,19 (0.795)	20,52 (0.808)	23,11 (0.910)	24,13 (0.950)	1,91 (0.075)	2,41 (0.095)

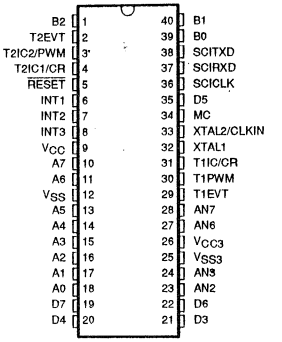
ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

INDEX

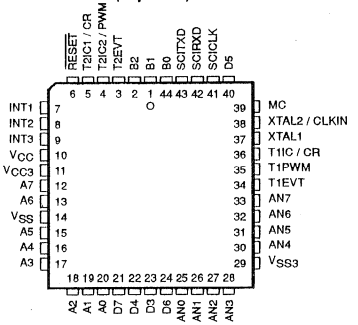
SPNS016 — OCTOBER 1991

- CMOS/EEPROM/EPROM Technologies on a Single Device
  - — Mask ROM Devices for High Volume Production
  - — One-Time Programmable (OTP) Devices for Low Volume Production
  - — Reprogrammable EPROM Devices for Prototyping Purposes
- Flexible Operating Features
  - — Power-Reduction STANDBY and HALT Modes
  - — Commercial and Industrial Temperature Ranges
  - — Input Clock Frequency 2 MHz to 20 MHz
  - — Voltage (V_{CC}) 5 V ± 10%
- Internal System Memory Configurations
  - — On-Chip Program Memory Versions
  - — ROM, 4K-Bytes or 8K-Bytes
  - — EPROM, 8K-Bytes
  - — Data EEPROM, 256-Bytes
  - — Static RAM, 256-Bytes Usable as Registers
- Eight Bit A/D Converter
  - — 4 Channels In 40-Pin Packages
  - — 8 Channels In 44-Pin Packages
- Two 16-Bit General Purpose Timers
  - — Software Configurable as Two 16-Bit Event Counters, or Two 16-Bit Pulse Accumulators, or Three 16-Bit Input Capture Functions, or Four Compare Registers, or Two Self-Contained PWM Functions
  - — Software Programmable Input Polarity
  - — One Timer Has an 8-Bit Prescaler, Providing a 24-Bit Realtime Timer
- On-Chip 24-Bit Watchdog Timer
- Serial Communications Interface (SCI)
  - — Asynchronous and Isosynchronous Modes
  - — Full Duplex, Double Buffered Rx and Tx
  - — Two Multiprocessor Communications Formats
- CMOS/TTL Compatible I/O Pins
  - — All Peripheral Function Pins Software Configurable for Digital I/O
  - — 40 Pin Plastic and Ceramic Dual-In-Line Package
    - – 27 Bidirectional, 5 Input Pins
  - — 44 Pin Plastic and Ceramic Leaded Chip Carrier Package
    - – 27 Bidirectional, 9 Input Pins

J, N and N2 Packages
(Top View)



FN and FZ Packages
(Top View)



- Flexible Interrupt Handling
  - — Two S/W Programmable Interrupt Levels
  - — Global and Individual Interrupt Masking
  - — Programmable Rising or Falling Edge Detect
- TMS370 Series Compatibility
  - — Register-to-Register Architecture
  - — 256 General-Purpose Registers
  - — 14 Powerful Addressing Modes
- PC-Based Workstation Development Support Emphasizes Productivity, Featuring:
  - — C-Compiler Support
  - — Realtime In-Circuit Emulation
  - — Symbolic Debug
  - — Extensive Breakpoint/Trace Capability
  - — Software Performance Analysis
  - — Multi-Window User Interface
  - — EEPROM/EPROM Programming

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

description

The TMS370C040, TMS370C042, TMS370C340, TMS370C342, TMS370C642, and TMS370C742 devices are members of the TMS370 family of single-chip 8-bit microcontrollers. The TMS370 family provides cost-effective realtime system control through VLSI integration of advanced peripheral function on-chip memory configurations.

The TMS370Cx4x family is implemented using high-performance silicon-gate CMOS technology. The low operating power, wide operating temperature range, and noise immunity of CMOS technology coupled with the high performance and extensive on-chip peripheral functions make the TMS370Cx4x devices attractive in system designs for automotive electronics, industrial motor, computer peripheral control, and telecommunications.

Unless otherwise noted, the term TMS370Cx4x refers to the TMS370C040, TMS370C042, TMS370C340, TMS370C342, TMS370C642 and TMS370C742 for all peripheral function modules available on those devices. All TMS370Cx4x devices contain a minimum of the following on-chip peripheral modules:

- 256-bytes RAM (usable as registers)
- 8-channel (for 44 pin device) or 4-channel (for 40 pin device) 8-bit Analog-to-Digital converter (A/D)
- Serial communications interface function (SCI)
- Two 24-bit general-purpose timers, one of which can be used as a Watchdog timer
- One 16-bit general-purpose timer

The following table provides an overview of the various memory configurations and operating modes of the TMS370Cx4x devices.

Memory Configurations

DEVICES	PROGRAM MEMORY (BYTES) ROM	PROGRAM MEMORY (BYTES) EPROM	DATA MEMORY (BYTES) RAM	DATA MEMORY (BYTES) EEPROM	PIN/PACKAGES
TMS370C040	4K	—	256	256	44 / FN-PLCC 40 / N-DIP 40 / N2-DIP
TMS370C042	8K	—	256	256	44 / FN-PLCC 40 / N-DIP 40 / N2-DIP
TMS370C340	4K	—	256	—	44 / FN-PLCC 40 / N-DIP 40 / N2-DIP
TMS370C342	8K	—	256	—	44 / FN-PLCC 40 / N-DIP 40 / N2-DIP
TMS370C642	—	8K	256	—	44 / FN-PLCC 40 / N-DIP 40 / N2-DIP
TMS370C742	—	8K	256	256	44 / FN-PLCC 40 / N-DIP 40 / N2-DIP

† System evaluators and development tools are for use only in a prototype environment and their reliability has not been characterized.

The 4K-bytes and 8K-bytes of mask-programmable ROM in the TMS370C040, TMS370C042, TMS370C340 and TMS370C342 are replaced in the TMS370C642 and TMS370C742 with 8K-bytes of EPROM while all other available memory and on-chip peripherals are identical, with the exception of no data EEPROM on the TMS370C340, TMS370C342, and TMS370C642 devices. OTP (TMS370C642 and TMS370C742) devices and the reprogrammable device (TMS370C742FZ) are the two versions of the TMS370Cx4x family that have 8K of EPROM program memory with a superset of the memory and peripherals of all the other family members with the same pin-out.

TMS370C642 and TMS370C742 (OTP) devices are in plastic packages and can be programmed one time. This is an effective microcomputer to use for immediate production updates for other members of the TMS370Cx4x family or for low volume production runs that cannot satisfy minimum volume or cycle time for the low-cost mask ROM devices.

The TMS370C742FZ is a windowed ceramic package to allow reprogramming of the program EPROM memory during the development prototyping phase of design. These TMS370C742FZ devices allow quick updates to breadboards and prototype systems while iterating initial designs.

The TMS370Cx4x provides two power reduction modes (STANDBY and HALT) for applications where low power consumption is critical. Both modes stop all CPU activity (i.e., no instructions are executed). In the STANDBY mode the internal oscillator, the general purpose timer, and the SCI receiver start bit detection remain active. In the HALT mode, all device activity is stopped. The device retains all RAM data and peripheral configuration bits throughout both powerdown modes.

The TMS370Cx4x features advanced register-to-register architecture that allows arithmetic and logical operations without requiring an accumulator (e.g., ADD r24, r47; add the contents of register 24 to the contents of register 47 and store the result in register 47). The TMS370Cx4x family is fully instruction-set-compatible, allowing easy transition between members.

The TMS370Cx4x family offers an 8-channel with 8-bit accuracy Analog-to-Digital converter for the 44-pin PLCC packages and also offers 4-channel Analog-to-Digital for the 40-pin DIP packages. The 33-µs conversion time at 20 MHz and the variable sample period, combined with selectable positive reference voltage sources, turn real-world analog signals into digital data.

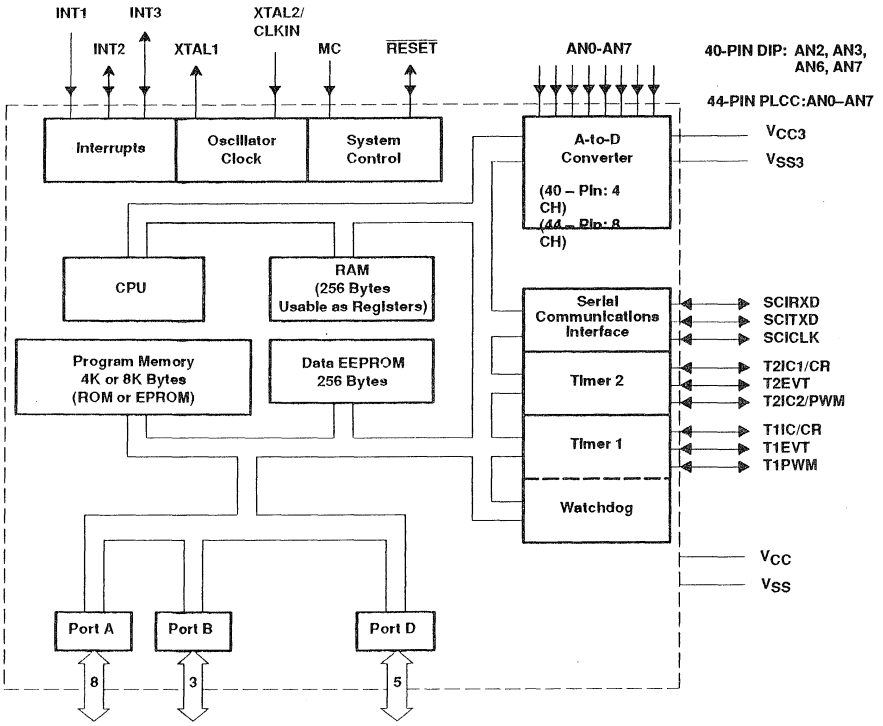
The Serial Communications Interface (SCI) module is a built-in serial interface that can be programmed to be asynchronous or isosynchronous to give three methods of serial communications. The SCI allows standard RS-232-C communications with other common data transmission equipment. The CPU takes no part in serial communications except to write data to be transmitted to a register and to read received data from a register.

The TMS370Cx4x family provides the system designer with very economical, efficient solutions to real-time control applications. The TMS370 family eXtended Development System (XDS) solves the challenge of efficiently developing the software and hardware required to design the TMS370Cx4x into an ever-increasing number of complex applications. The application source code can be written in Assembly and C-language, and the output code can be generated by the linker. The TMS370 family XDS communicates via a standard RS-232-C interface with an existing personal computer to form a PC-DOS hosted workstation. This allows the use of the PC's editors and software utilities already familiar to the designer. The TMS370 family XDS emphasizes ease-of-use through extensive use of menus and screen windowing so that a system designer can begin developing software with minimum training. Precise realtime in-circuit emulation and extensive symbolic debug and analysis tools ensure efficient software and hardware implementation as well as reducing the time-to-market cycle.

The TMS370Cx4x family together with the TMS370 family XDS for applications development, the TMS370C742FZ reprogrammable devices, comprehensive product documentation, and customer support, provide a complete solution for the needs of the system designer.

The TMS370Cx4x family members are available in three package types: 44-pin PLCC (FN), 40-pin DIP (N) and 40-pin shrink DIP (N2). The shrink DIP package is the same width as the standard 40-pin DIP (0.600″) but the space between pins is reduced from 0.100″ to 0.070″. This results in a 30% board space savings yielding a 40-pin package the size of a standard 28-pin DIP.

functional block diagram



Terminal Functions

PIN NO. DIP (40)	PIN NO. PLCC (44)	I/O	DESCRIPTION
A0	20	I/O	Port A pins are general purpose bidirectional I/O ports.
A1	19	I/O	
A2	18	I/O	
A3	17	I/O	
A4	16	I/O	
A5	15	I/O	
A6	13	I/O	
A7	12	I/O	
B0	44	I/O	Port B pins are general purpose bidirectional I/O ports.
B1	1	I/O	
B2	2	I/O	
			Port D pins are general purpose bidirectional I/O ports.
D3	23	I/O	D3 also configurable as CLKOUT (see Note 1)
D4	22	I/O	
D5	40	I/O	
D6	24	I/O	
D7	21	I/O	
AN0/E0	25	I	A/D analog input channels or positive reference pins.
AN1/E1	26	I	
AN2/E2	27	I	
AN3/E3	28	I	
AN4/E4	30	I	
AN5/E5	31	I	
AN6/E6	32	I	Any A/D channel may be programmed as general purpose input pins (E port) if not
AN7/E7	33	I	used as an analog input or reference channel.
V_{CC3}	11		A/D converter positive supply voltage and optional positive reference input pin.,
V_{SS3}	29		A/D converter ground supply and low reference input pin.
INT1	7	I	External non-maskable or maskable interrupt/General purpose input pin.
INT2	8	I/O	External maskable interrupt input/General purpose bidirectional pin.
INT3	9	I/O	External maskable interrupt input/General purpose bidirectional pin.
T1IC/CR	36	I/O	Timer 1 input Capture/Counter Reset input pin/General purpose bidirectional pin.
T1PWM	35	I/O	Timer 1 Pulse-Width-Modulation output pin/General purpose bidirectional pin.
T1EVT	34	I/O	Timer 1 External Event input pin/general purpose bidirectional pin.
T2IC1/CR	5	I/O	Timer 2 input capture/counter reset input pin/General purpose bidirectional pin.
T2IC2/PWM	4	I/O	Timer 2 input capture 2/PWM output pin/General purpose bidirectional pin.
T2EVT	3	I/O	Timer 2 External Event input pin/General purpose bidirectional pin.
SCITXD	43	I/O	SCI Transmit Data Output pin/General purpose bidirectional pin.
SCIRXD	42	I/O	SCI Receive Data input pin/General purpose bidirectional pin.
SCICLK	41	I/O	SCI bidirectional Serial Clock pin/General purpose bidirectional pin.
$\overline{RESET}$	6	I/O	System reset bidirectional pin. As input it initializes microcontroller, as open-drain output it indicates an internal failure was detected by the Watchdog or Oscillator Fault circuit.
MC	39	I	Mode control input pin; enables the EEPROM Write Protection Override (WPO) mode.
XTAL1	37	I	Internal oscillator output for crystal
XTAL2/CLKIN	38	O	Internal oscillator crystal input/external clock source input
V_{CC}	10		Positive supply voltage.
V_{SS}	14		Ground reference.

NOTE 1: Each pin associated with Interrupt 2, Interrupt 3, Timer 1, Timer 2 and SCI functional blocks may be individually programmed as a general purpose bidirectional pin if it is not used for its primary block function. Unused A/D input channel pins may be programmed as a general purpose input only pins. D3 may be configured as CLKOUT by appropriately programming the DPORT1 and DPORT2 registers.

memory map

The TMS370 family architecture is based on the Von Neumann architecture, where the program memory and data memory share a common address space. All peripheral input/output is memory mapped in this same common address space. As shown in Figure 1, the TMS370 family provides memory-mapped RAM, ROM, EEPROM, EPROM, input/output pins, and peripheral functions.

The peripheral file contains all input/output port control, peripheral status and control, EPROM memory programming, and system-wide control functions. The peripheral file is located between 1010h to 107Fh and is logically divided into 6 Peripheral File Frames of 16 bytes each. Each on-chip peripheral is assigned to a separate frame through which peripheral control and data information is passed. The TMS370Cx4x has 5 peripheral frames and a system control frame assigned to Peripheral File Frames 1, 2, 4, 5, 6, and 7.

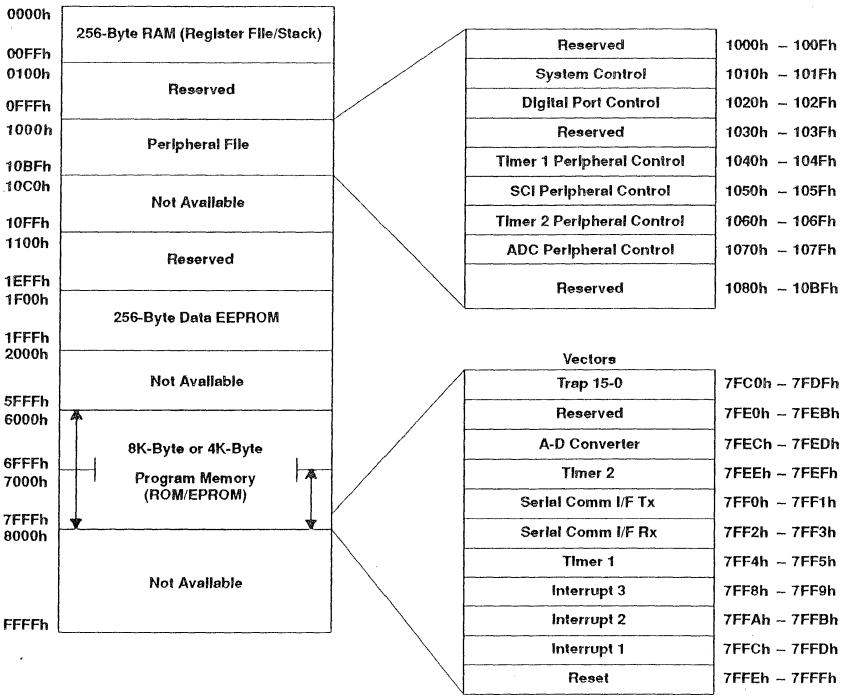


Figure 1. TMS370Cx4x Memory Map

memories

RAM/register file

The TMS370Cx4x has 256 bytes of static RAM, which serve as both the CPU register file and general-purpose memory. The RAM is treated as registers by the instruction set and is referenced as R0 through R255. The first two registers, R0 and R1, are also called the A and B registers, respectively. The stack is located in the RAM, and operates as a last-in first-out read/write memory. It is used to store the return address on subroutine calls and the status register during interrupts. Accessing this memory as registers is performed in one system clock cycle (t_c), while general-purpose memory access is performed in two system clock cycles.

Instructions may be executed from RAM. This versatility enables the internal RAM to be used for functions such as microcontroller self-test, diagnostics, or system test of the end application. The user may load external programs or data into the RAM by incorporating a simple bootstrap loader in the program memory.

data EEPROM

The TMS370Cx4x family has 256 bytes of on-chip Electrically Erasable Programmable ROM (EEPROM), addressed as 256 consecutive bytes mapped from locations 1F00h to 1FFFh. The data EEPROM provides nonvolatile programmable storage for items such as calibration constants and configuration information for personalization of a generic program ROM/EPROM algorithm for use in specific end applications. The data EEPROM supports bit, byte, and block write/erase modes. Instructions may be executed from the data EEPROM, providing additional program space and the ability to patch algorithms by placing a branch table for volatile routines in the data EEPROM.

The data EEPROM uses the 5-V V_{CC} supply voltage and provides the programming voltage via an internal dedicated generator, eliminating the need for an external high-voltage programming source. The dedicated voltage generator optimizes the programming voltage characteristics, increasing the reliability as well as extending the write/erase endurance of the array.

Programming control and status monitoring are performed through the data EEPROM control register (DEECTL) in the peripheral file. An EEPROM write/erase operation is performed in the following sequence:

1. Perform normal memory write to the target EEPROM location.
2. Write to DEECTL control register to select WRITE1/WRITE0 and set the EXECUTE (EXE) bit to 1.
3. Wait for program time to elapse [$t_{w(PGM)B}$ or $t_{w(PGM)AR}$].
4. Write to DEECTL control register to set the EXECUTE (EXE) bit to 0.

The WRITE1/WRITE0 control bit selects whether the zeros or the ones in the data byte are to be programmed into the selected EEPROM location. For example, a WRITE1 operation will program ones into all bit positions within the EEPROM byte that have ones in the data byte, while bits that are zero in the data byte will not affect the EEPROM contents. The WRITE1 operation effectively performs a logical OR of the information previously stored on the EEPROM byte with the data byte. The WRITE0 operation effectively performs a logical AND between these two bytes. Single bit programming within an EEPROM byte is performed by writing only the zeros or ones of the data byte. The EEPROM programming algorithm may use this bit-programming capability to optimize the life of the EEPROM.

When a data value cannot be achieved by writing only zeros or only ones into the EEPROM byte, a WRITE1 followed by a WRITE0 will program any data value into the EEPROM byte, regardless of the previous data stored at that location.

Data EEPROM read accesses are performed as normal memory read operations in two system clock cycles. A memory read cycle to any EEPROM location while EXECUTE = 1 returns the value currently being written to the EEPROM. Following an EEPROM write operation, the EEPROM voltages must stabilize prior to performing an EEPROM read operation. The BUSY FLAG indicates the status of the EEPROM voltage. When set, the EEPROM is not ready for a read operation. The BUSY FLAG is reset to 0 by the EEPROM control logic

when 128 system clock cycles have elapsed following the EXECUTE bit being set to 0. If an EEPROM read operation is performed while BUSY = 1, automatic WAIT states will be generated until BUSY = 0, and then the read operation will be performed.

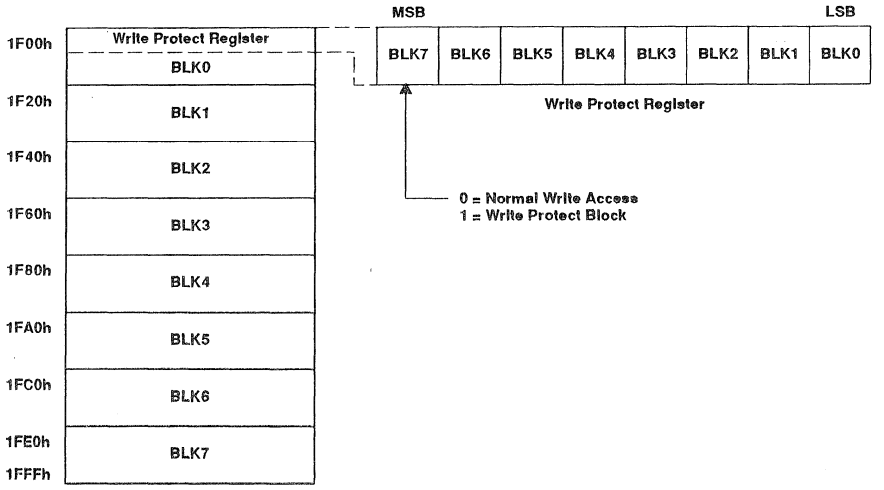


Figure 2. Write Protect Register for TMS370x4x Devices With A 256-Byte Data EEPROM Array

Bytes within the data EEPROM may be protected from inadvertent overwriting of critical information. As shown in Figure 2, the 8-bit Write Protect Register (WPR), located at 1F00h within the data EEPROM, provides write protection for the 256-byte data EEPROM, segmenting the array into eight blocks of 32 bytes each. Each of these 32-byte blocks may be individually write- and erase-protected by setting the corresponding bit to 1 in the appropriate WPR. Since the WPRs reside in the array in BLK0, the WPR may also be write-protected, thereby increasing the system reliability by preventing bytes from being reprogrammed. Bytes left unprotected may be written to by the normal EEPROM programming sequence. The Write Protection Override (WPO) mode enables data to be written to any location in the data EEPROM, regardless of the WPR contents. Enter the WPO mode by placing 12-V on the MC pin. The WPO mode is typically used in a service environment to update the protected EEPROM contents.

All unprotected bytes within the data EEPROM array may be programmed during a single EEPROM programming cycle by setting the ARRAY PROG bit of DEECTL to 1 at the start of the programming cycle.

program ROM

The program ROM consists of 4K- to 16K-bytes of mask programmable read-only memory. The program ROM is used for permanent storage of data or instructions, with read operations performed in two system clock cycles. Memory addresses 7FECh through 7FFFh are reserved for interrupt and reset vectors. Trap vectors, used with TRAP0 through TRAP15 instructions, are located between addresses 7FC0h and 7FDFh. Programming of the mask ROM is performed at the time of device fabrication.

program EPROM (TMS370C642 and TMS370C742 only)

The program EPROM of the TMS370C642 and TMS370C742 is an 8K electrically programmable read-only memory, addressed as 8K consecutive bytes mapped from location 6000h to 7FFFh. It provides application performance identical to the TMS370Cx4x mask ROM devices. Program instructions are read from the program EPROM in two system clock cycles, providing the prototyping capability of the mask program ROM.

An external supply (V_{PP}) is needed at the MC pin to provide the necessary programming voltage (V_{PP}). Programming is controlled through a register (EPCTL) in the peripheral file.

The TMS370C642 and TMS370C742 each come in a plastic package and cannot be erased. They are one-time-programmable (OTP) devices. The TMS370C742 comes in ceramic package with a quartz window. Before programming, the TMS370C742's EPROM is erased by exposing the device through the transparent window to high-intensity ultraviolet light (wavelength 2537Å). The recommended minimum exposure dose (UV intensity × exposure time) is 15 W•s/cm^2. A typical 12-mW/cm^2, filterless UV lamp will erase the device in 21 minutes. The lamp should be located about 2.5 cm above the chip during erasure. After erasure, the entire array is in a logic 1 state. A programmed 0 can be erased to a 1 only by exposure to ultraviolet light. It should be noted that normal ambient light contains the correct wavelength for erasure. When using the TMS370C742, the window should be covered with an opaque label. All devices are erased to logic 1 when delivered from the factory.

CAUTION

Exposing the EPROM module to ultraviolet light may also cause erasure in any EEPROM module. Any useful data stored in the EEPROM must be reprogrammed after exposure to UV light.

Programming 0 to the EPROM is controlled by the EPCTL register via the EXE bit and the VPPS bit. The EXE bit initiates EPROM programming when set and disables programming when cleared. The VPPS bit connects the programming voltage V_{PP} at the MC pin to the EPROM module. VPPS (EPCTL.6) and EXE (EPCTL.0) should be set separately, and the VPPS bit should be set at least two microseconds before the EXE bit is set. After programming, the application programming should wait for four microseconds before any read attempt is made. The programming operation (see Figure 3) is performed in the following recommended sequence:

1. Supply the programming voltage to the MC pin.
2. Write to EPCTL register to set the VPPS bit to 1.
3. Perform normal memory write register to the target EPROM location.
4. Write to EPCTL register to set the EXE bit to 1. (Wait at least two microseconds after step 2).
5. Wait for program time to elapse (one millisecond).
6. Write to EPCTL register to clear the EXE bit (leave VPPS bit set to 1).
7. Read the byte being programmed; if correct data is not read, repeat steps 4 through 6 X times up to a maximum of 25.
8. Write to EPCTL register to set the EXE bit to 1 for final programming.
9. Wait for program time to elapse (3X milliseconds duration).
10. Write to EPCTL register to clear the EXE and VPPS bits.

An external power supply at V_{PP}, I_{PP} (30mA) is required for programming operations. Programming voltage V_{PP} is supplied via the MC pin. This also automatically puts the microcontroller in the Write Protection Override (WPO) mode. Programming voltage may be applied via the MC pin anytime after $\overline{\text{RESET}}$ and will remain at V_{PP} after programming (after the EXECUTE bit is cleared). Applying programming voltage while $\overline{\text{RESET}}$ is active will put the microcontroller in reserved mode, where programming operation is inhibited.

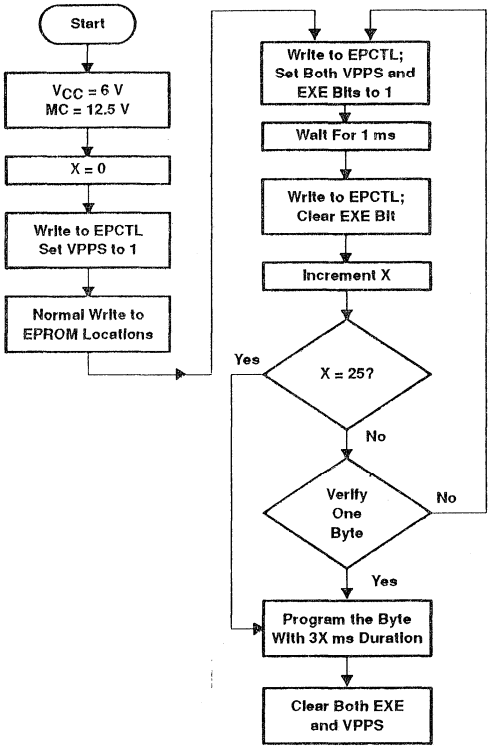


Figure 3. EPROM Programming Operation

write protection of program EPROM

To override the EPROM write protection, the V_{PP} voltage must be applied to the MC pin and the VPPS bit (EPCTL.6) must be set. This dual requirement ensures that the program EPROM will not accidentally be overwritten during the data EEPROM operations when V_{PP} is applied to the MC pin. The data EEPROM may be programmed when the VPPS bit is set.

central processing unit

The central processing unit (CPU) of the TMS370 series is an enhanced version of the TMS7000 Family CPU. The enhancements include additional user instructions such as integer divide, conditional jump instructions based on the overflow status bit, and addressing modes such as stack-pointer-relative addressing for subroutine parameter passing. The efficient register-to-register architecture of the TMS7000 family has been carried over to the TMS370 family, which avoids the conventional accumulator bottleneck. The complete TMS370 family instruction set is summarized in the table, *TMS370 Instruction Set Summary,* page 39.

In addition to the interpretation and execution of the user program, the CPU performs the functions of bus protocol generation and interrupt priority arbitration. While the CPU is implemented independent of the memory, input/output, and peripheral modules, it performs the central system control function through communications with these on-chip modules and external memory and peripherals.

The TMS370 family CPU registers accessible to the programmer are shown in Figure 4. The register file consists of 256 general purpose registers, R0 through R255 implemented in on-chip RAM, and is used by the CPU for general purpose 8- and 16-bit source and destination operands, index registers, and indirect addressing. The first two registers, R0 and R1, are also called registers A and B and are used by the CPU as general purpose registers or for implied operands. The program counter (PC) contains the address of the next instruction to be executed. The stack pointer (SP) contains the address of the last or top entry on the stack, which is located in the on-chip register file. The status register (ST) contains four bits that reflect the outcome of the instruction just executed, and two bits that control the masking of the interrupt priority chains.

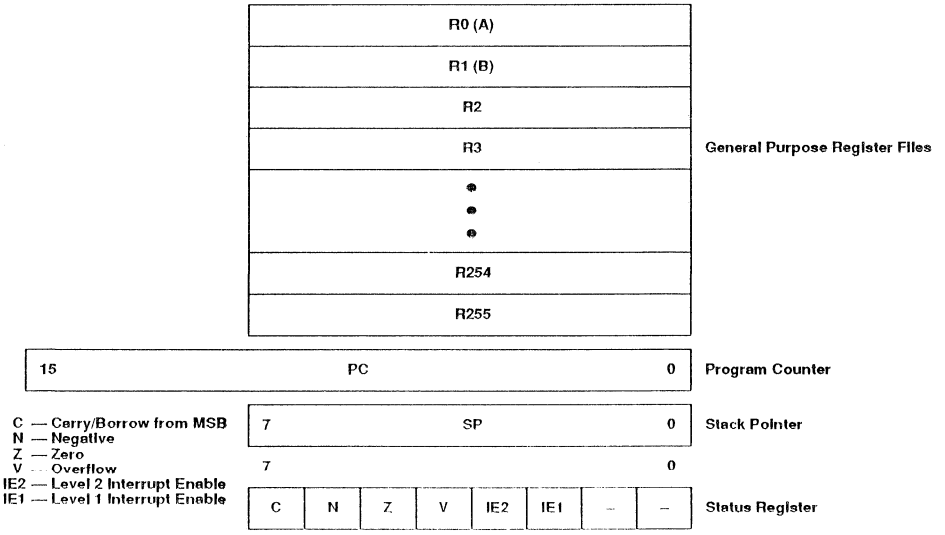


Figure 4. CPU Registers

system resets

The TMS370Cx4x has three possible reset sources: a low input to the $\overline{\text{RESET}}$ pin, a programmable watchdog timer timeout, or a programmable oscillator fault failure. The $\overline{\text{RESET}}$ pin, an input/output pin, initiates TMS370Cx4x hardware initialization and ensures an orderly software startup. A low-level input of at least 50 ns initiates the reset sequence. The microcontroller is held in reset until the $\overline{\text{RESET}}$ pin goes inactive (high). If the $\overline{\text{RESET}}$ input signal is low for less than eight system clock cycles, the TMS370Cx4x will hold the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The $\overline{\text{RESET}}$ pin must be activated by the application at power-up, which can be accomplished by an external input or an RC power-up reset circuit. Recall that the basic operating mode, microcomputer or microprocessor, is determined by the voltage level applied to the MC pin two cycles before the $\overline{\text{RESET}}$ pin goes inactive (high). The $\overline{\text{RESET}}$ pin can be asserted at any time during operation, resulting in an immediate initiation of the reset sequence.

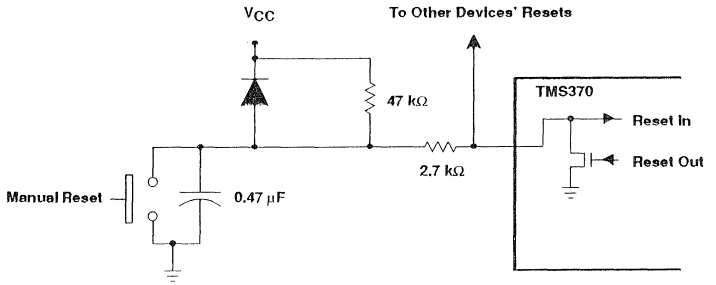


Figure 5. Typical Reset Circuit

The watchdog timer provides system integrity by detecting a program that has become lost or is not executing as expected. A system reset is generated if the watchdog timer is not properly re-initialized by a specific software sequence, or if the re-initialization does not occur before the watchdog timer times out. The watchdog timer timeout initiates the TMS370Cx4x reset sequence and drives the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The watchdog reset function is enabled by setting the WD OVRFL RST ENA bit of T1CTL2 to 1. Once the software enables the watchdog reset function, subsequent writes to the WD OVRFL RST ENA bit are ignored. Watchdog control bits can be initialized only following a power-up reset. The timer section discusses additional information on the watchdog timer and its configurations.

The oscillator fault circuit provides the means to monitor failures of the oscillator input signal (XTAL2/CLKIN). This function is enabled under software control by setting the OSC FLT RST ENA bit of SCCR2 to 1. If the oscillator input signal frequency remains above the 90% point of the minimum operating frequency (CLKIN), the oscillator input will not to be activated. However, if the oscillator input is lost or its frequency falls below 20 kHz and the oscillator fault reset is enabled, the TMS370Cx4x is reset and the external $\overline{\text{RESET}}$ pin is driven low.

Reset Sources

REGISTER	ADDRESS	PF	BIT NO.	CONTROL BIT	SOURCE OF RESET
SCCR0	1010h	P010	7	COLD START	Cold or Warm start reset.
SCCR0	1010h	P010	4	OSC FLT FLAG	Oscillator out of range.
T1CTL2	104Ah	P04A	5	WD OVRFL INT FLAG	Watchdog timer timeout.

When an oscillator input failure occurs, the internal clocks are stopped and $\overline{\text{RESET}}$ is held active until the oscillator input frequency is greater than 100 kHz typical. If the OSC FLT RST ENA bit of SCCR2 is set to 0, the fault-detection circuit independently sets the OSC FLT FLAG of SCCR0 without generating a system reset. The fault detection circuit can be disabled if the OSC FLT DISABLE bit is set to 1. The OSC FLT RST ENA bit is protected during non-privileged operation and therefore should be software configured during the initialization sequence following system reset. During the HALT mode the oscillator fault circuitry will be disabled.

During a microcontroller reset, the majority of the peripheral file bits are set to 0, with the exception of the bits shown in the following table. During all resets, the COLD START, OSC FLT FLAG, and the WD OVRFL FLAG are appropriately set by the active reset and may be interrogated by the program to determine the source of system reset. Registers A and B are set to zero during all resets. The other registers are not affected by a reset under power (warm reset).

Control Bit States Following Reset

REGISTER	CONTROL BIT	POWERUP MICROCOMPUTER	WARM RESET MICROCOMPUTER
SCCR0	µP/µC MODE MC PIN DATA COLD START OSC FLT FLAG	0 0 1 0	0 0 † †
T1CTL2	WD OVRFL FLAG	0	†
TXCTL	TX EMPTY	1	1
TXCTL	TXRDY	1	1
ADSTAT	AD READY	1	1

† Status bit corresponding to active reset source is set to 1.

Interrupts

The TMS370 family software-programmable interrupt structure supports flexible on-chip and external interrupt configurations to meet realtime interrupt-driven application requirements. The hardware interrupt structure incorporates two priority levels as shown in Figure 6. Interrupt level 1 has a higher priority than interrupt level 2. The two priority levels can be independently enabled by the global interrupt enable bits (IE1 and IE2) of the Status Register.

Each system interrupt is independently configured on either the high or low priority chain by the application program during system initialization. Within each interrupt chain, the interrupt priority is fixed by the position of the system interrupt. However, since each system interrupt is selectively configured on either the high or low priority interrupt chain, the application program can elevate any system interrupt to the highest priority. Arbitration between the two priority levels is performed within the CPU. Arbitration within each of the priority chains is performed within the peripheral modules to support interrupt expansion to future modules. Pending interrupts are serviced upon completion of current instruction execution, depending on their interrupt mask and priority conditions.

The TMS370Cx4x has nine hardware system interrupts as shown in the table on page 15. Each system interrupt has a dedicated interrupt vector located in program memory through which control is passed to the interrupt service routines. A system interrupt may have multiple interrupt sources (e.g., SCI RXNT has two interrupt

sources). All of the interrupt sources are individually maskable by local interrupt enable control bits in the associated peripheral file. Each interrupt source FLAG bit is individually readable for software polling or to determine which interrupt source generated the associated system interrupt.

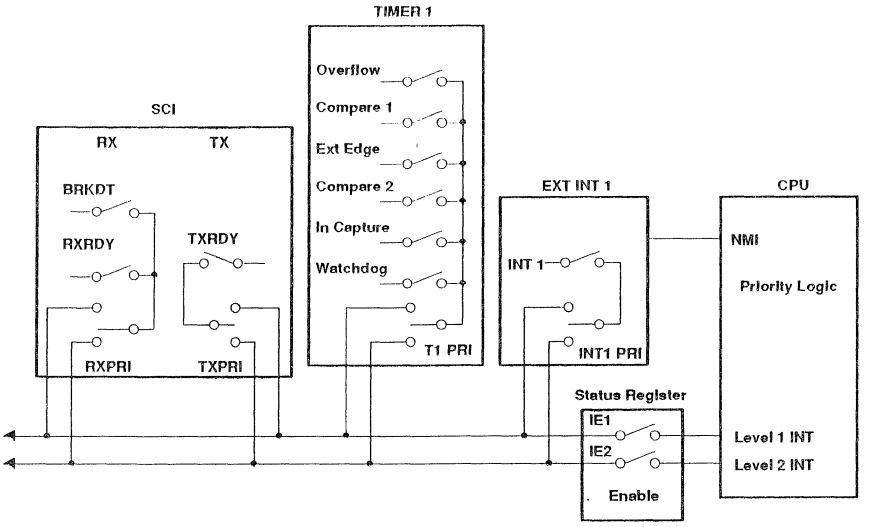


Figure 6. Interrupt Control

Five of the system interrupts are generated by on-chip peripheral functions, and three external interrupts are supported. Software configuration of the external interrupts is performed through the INT1, INT2, and INT3 control registers in peripheral file frame 1. Each external interrupt is individually software configurable for input polarity (rising or falling) for ease of system interface. External interrupt INT1 is software configurable as either a maskable or non-maskable interrupt. When INT1 is configured as non-maskable, it cannot be masked by the individual or global enable mask bits. Recall that the INT1 NMI bit is protected during non-privileged operation and therefore should be configured during the initialization sequence following reset. To maximize pin flexibility, external interrupts INT2 and INT3 can be software configured as general purpose input/output pins if the interrupt function is not required (INT1 can be similarly configured as an input pin).

Hardware System Interrupts

INTERRUPT SOURCE	INTERRUPT FLAG	SYSTEM INTERRUPT	VECTOR ADDRESS	PRIORITY§
External $\overline{\text{RESET}}$ Watchdog Overflow Oscillator Fault Detect	COLD START WD OVRFL INT FLAG OSC FLT FLAG	$\overline{\text{RESET}}$†	7FFEh, 7FFFh	1
External INT1	INT1 FLAG	INT1†	7FFCh, 7FFDh	2
External INT2	INT2 FLAG	INT2†	7FFAh, 7FFBh	3
External INT3	INT3 FLAG	INT3†	7FF8h, 7FF9h	4
Timer 1 Overflow Timer 1 Compare 1 Timer 1 Compare 2 Timer 1 External Edge Timer 1 Input Capture Watchdog Overflow	T1 OVRFL INT FLAG T1C1 INT FLAG T1C2 INT FLAG T1EDGE INT FLAG T1IC INT FLAG WD OVRFL INT FLAG	T1INT‡	7FF4h, 7FF5h	6
SCI RX Data Register Full SCI RX Break Detect	RXRDY FLAG BRKDT FLAG	RXINT†	7FF2h, 7FF3h	7
SCI TX Data Register Empty	TXRDY FLAG	TXINT	7FF0h, 7FF1h	8
Timer 2 Overflow Timer 2 Compare 1 Timer 2 Compare 2 Timer 2 External Edge Timer 2 Input Capture 1 Timer 2 Input Capture 2	T2 OVRFL INT FLAG T2C1 INT FLAG T2C2 INT FLAG T2EDGE INT FLAG T2IC1 INT FLAG T2IC2 INT FLAG	T2INT	7FEEh, 7FEFh	9
A-D Conversion Complete	AD INT FLAG	ADINT	7FECh, 7FEDh	10

† Releases microcontroller from STANDBY and HALT low power modes.
‡ Releases microcontroller from STANDBY low power mode.
§ Relative priority within an interrupt level.

privileged operation and EEPROM write protection override

The TMS370Cx4x family is designed with significant flexibility to enable the designer to software-configure the system and peripherals to meet the requirements of a broad variety of applications. The non-privileged mode of operation ensures the integrity of the system configuration once defined for an end application. Following a hardware reset, the TMS370Cx4x operates in the privileged mode, where all peripheral file registers have unrestricted read/write access and the application program will configure the system during the initialization sequence following reset. As the last step of system initialization, the PRIVILEGE DISABLE bit (SCCR2.0) will be set to 1, entering the non-privileged mode and disabling write operations to specific configuration control bits within the peripheral file. The following system configuration bits are write-protected during the non-privileged mode and must be configured by software prior to exiting the privileged mode:

REGISTER†		CONTROL BIT
NAME	LOCATION	
SCCR0	P010.5	PF AUTO WAIT
SCCR0	P010.6	OSC POWER
SCCR1	P011.2	MEMORY DISABLE
SCCR1	P011.4	AUTOWAIT DISABLE
SCCR2	P012.0	PRIVILEGE DISABLE
SCCR2	P012.1	INT1 NMI
SCCR2	P012.2	OSC FLT DISABLE
SCCR2	P012.5	OSC FLT RST ENA
SCCR2	P012.6	PWRDWN/IDLE
SCCR2	P012.7	HALT/STANDBY
SCIPRI	P05F.5	SCI RX PRIORITY
SCIPRI	P05F.6	SCI TX PRIORITY
T1PRI	P04F.6	T1 PRIORITY
T2PRI	P06F.6	T2 PRIORITY
ADPRI	P07F.6	AD PRIORITY

† The privileged bits are shown in a **bold typeface** in the Peripheral File Frames of the following sections.

The Write Protection Override (WPO) mode provides an external hardware method of overriding the write protection registers (WPR) of data EEPROM on the TMS370Cx4x. WPO mode is entered by applying a 12-V input to the MC pin after the $\overline{\text{RESET}}$ pin input goes high. The high voltage on the MC pin during the WPO mode is not the programming voltage for the data EEPROM or program EPROM. All EEPROM programming voltages are generated on-chip. The WPO mode provides hardware system level capability to modify the personality or calibration information in the data EEPROM while the device remains in the application, but only while requiring a 12-volt external input on the MC pin (normally not available in the end application except in a service or diagnostic environment).

low-power operating modes

The STANDBY and HALT low power modes significantly reduce power consumption by reducing or stopping the activity of the various on-chip peripherals when processing is not required. Each of the low power modes is entered by executing the IDLE instruction when the PWRDWN/IDLE bit in SCCR2 has been set to1. The HALT/STANDBY bit in SCCR2 controls which low-power mode is entered.

In the STANDBY mode (HALT/STANDBY = 0), all CPU activity and most peripheral module activity is stopped; however, the oscillator, internal clocks, Timer 1, and the receive start bit detection circuit of the serial communications interface remain active. System processing is suspended until a qualified interrupt (hardware $\overline{\text{RESET}}$, external interrupt on INT1, INT2, INT3, Timer 1 interrupt, or a low level on the receive pin of the serial communications interface) is detected.

In the HALT mode (HALT/STANDBY=1), the TMS370x4x is placed in its lowest power consumption mode. The oscillator and internal clocks are stopped, causing all internal activity to be halted. System activity is suspended until a qualified interrupt (hardware RESET external interrupt on INT1, INT2, or INT3, or low level on the receive pin of the serial communications interface) is detected.

POWERDOWN CONTROL BITS		MODE SELECTED
PWRDWN/IDLE (SCCR2.6)	HALT/STANDBY (SCCR2.7)	
1	0	Standby
1	1	Halt

The following information is preserved throughout both the STANDBY and HALT modes: RAM (register file), CPU registers (stack pointer, program counter, and status register), I/O pin direction and output data, and status registers of all on-chip peripheral functions. Since all CPU instruction processing is stopped during the STANDBY and HALT modes, the clocking of the watchdog timer is inhibited.

The following table, Peripheral File Frame 1, contains system configuration and control functions and registers for controlling EEPROM programming. The privileged bits are shown in a **bold typeface**.

Peripheral File Frame 1: System Configuration and Control Registers†

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1010h	P010	COLD START	**OSC POWER**	**PF AUTO WAIT**	OSC FLT FLAG	MC PIN WPO	MC PIN DATA	—	µP/µC MODE	SCCR0
1011h	P011	—	—	—	**AUTOWAIT DISABLE**	—	**MEMORY DISABLE**	—	—	SCCR1
1012h	P012	**HALT/ STANDBY**	**PWRDWN/ IDLE**	**OSC FLT RST ENA**	**BUS STEST**	**CPU STEST**	**OSC FLT DISABLE**	**INT1 NMI**	**PRIVILEGE DISABLE**	SCCR2
1013h to 1016h	P013 to P016	RESERVED								
1017h	P017	INT1 FLAG	INT1 PIN DATA	—	—	—	INT1 POLARITY	INT1 PRIORITY	INT1 ENABLE	INT1
1018h	P018	INT2 FLAG	INT2 PIN DATA	—	INT2 DATA DIR	INT2 DATA OUT	INT2 POLARITY	INT2 PRIORITY	INT2 ENABLE	INT2
1019h	P019	INT3 FLAG	INT3 PIN DATA	—	INT3 DATA DIR	INT3 DATA OUT	INT3 POLARITY	INT3 PRIORITY	INT3 ENABLE	INT3
101Ah	P01A	BUSY	—	—	—	—	AP	W1W0	EXE	DEECTL
101Bh	P01B	RESERVED								
101Ch	P01C	BUSY	VPPS	—	—	—	—	W0	EXE	EPCTL
101Dh 101Eh 101Fh	P01D P01E P01F	RESERVED								

† Privileged bits are shown in **bold typeface.**

peripheral file frame 2

Peripheral File Frame 2 contains the digital I/O pin configuration and control registers. The following tables detail the specific addresses, registers, and control bits within the Peripheral File Frame.

Peripheral File Frame 2: System Configuration and Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	
1020h	P020	Reserved								A PORT1
1021h	P021	Port A Control Register 2 (must be 0)								A PORT2
1022h	P022	Port A Data								A DATA
1023h	P023	Port A Direction								A DIR
1024h	P024	Reserved								B PORT1
1025h	P025	X	X	X	X	X	Port B Control Register 2			B PORT2
1026h	P026	X	X	X	X	X	Port B Data			B DATA
1027h	P027	X	X	X	X	X	Port B Direction			B DIR
1028h to 102Bh	P028 to P02B	Reserved								
102Ch	P02C	Port D Control Register 1 (must be 0)					X	X	X	D PORT1
102Dh	P02D	Port D Control Register 2 (must be 0)†					X	X	X	D PORT2
102Eh	P02E	Port D Data					X	X	X	D DATA
102Fh	P02F	Port D Direction					X	X	X	D DIR

† To configure pin D3 as CLKOUT, set Port D Control Register 2 equal to 08h.

Port Configuration Register Set-up

PORT	PIN	abcd 00q1	abcd 00x0
A	0 — 7	Out q	Data In
B	0 — 2	Out q	Data In
C	3 — 7	Out q	Data In
a = Port x Control Register 1 b = Port x Control Register 2 c = Data d = Direction			

NOTES: 2. Each bit controls the corresponding pin; for example, bit 6 controls port pin 6. Each pin is individually configurable.
3. Only register combination 00xx is defined for TMS370Cx4x.

programmable timers

The two programmable timer modules of the TMS370Cx4x provide the designer with the enhanced timer resources required to perform realtime system control. The Timer 1 module contains the general-purpose timer T1 and the watchdog timer (WD). The three independent 16-bit timers, T1, T2, and WD, allow program selection of input clock sources (realtime, external event, or pulse accumulate) with multiple 16-bit registers (input capture and compare) for special timer function control. These timers provide the capabilities for:

System Requirements	Timer Resource
Realtime system control	Interval Timers with Interrupts
Input Pulse-Width measurement	Pulse-Accumulate or Input-Capture Functions
External Event Synchronization	Event Counter Function
Timer Output Control	Compare Function
Pulse-Width Modulated Output Control	PWM Output Function
System Integrity	Watchdog Function

timer 1 module

The timer 1 module consists of three major blocks:

1. Prescaler/Clock Source, which determines the independent clock sources for the general purpose timer and the watchdog timer.
2. 16-bit General Purpose Timer, T1, which provides the event count, input capture, and compare functions.
3. 16-bit Watchdog Timer, which may be software programmed as an event counter/pulse accumulator if the watchdog function is not desired.

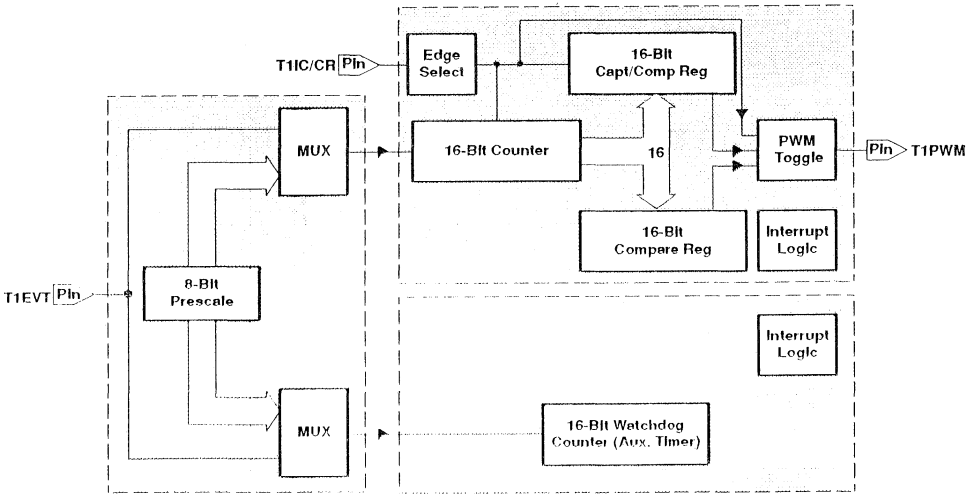


Figure 7. Timer 1 Module Block Diagram

timer 1 module prescaler/clock source

The clock source inputs for the general purpose timer and the watchdog timer are independently configured by the T1 and WD INPUT SELECT control bits of the T1CTL1 control register. The WD INPUT SELECT control bits cannot be changed after entering the watchdog mode (WD RST ENA = 1). Eight possible clock sources are programmable for each counter.

T1 INPUT			CLOCK SOURCE	WD INPUT		
SELECT 2	SELECT 1	SELECT 0		SELECT 2	SELECT 1	SELECT 0
0	0	0	System Clock	0	0	0
0	0	1	Pulse Accumulate	0	0	1
0	1	0	Event Input	0	1	0
0	1	1	No Clock Input	0	1	1
1	0	0	System Clock/4	1	0	0
1	0	1	System Clock/16	1	0	1
1	1	0	System Clock/64	1	1	0
1	1	1	System Clock/256	1	1	1

For realtime control applications, both the general-purpose timer and the watchdog timer are independently programmable from 16 to 24 bits in length. The 24-bit prescaler/timer generates overflow rates ranging from

13.1 ms with 200 ns timer resolution to 3.35 seconds with 51.2 µs timer resolution (external clock = 20 MHz).

In the **event counter mode**, an external high-to-low transition on the T1EVT pin is used to provide the clock for the internal timers. As shown in Figure 8, the T1EVT input provides the timer clock and is not routed through the prescaler. The T1EVT external clock frequency may not exceed the system clock frequency divided by 2. The general-purpose timer and the watchdog timer are programmable as 16-bit event counters.

In the **pulse accumulate mode**, an external input on the T1EVT pin is used to gate the internal system clock to the internal timers. While T1EVT input is logic one (high), the timers will be clocked at the system clock rate and will accumulate system clock pulses until the T1EVT pin returns to a logic 0. Both the general purpose timer and the watchdog timer are programmable as 16-bit pulse accumulators.

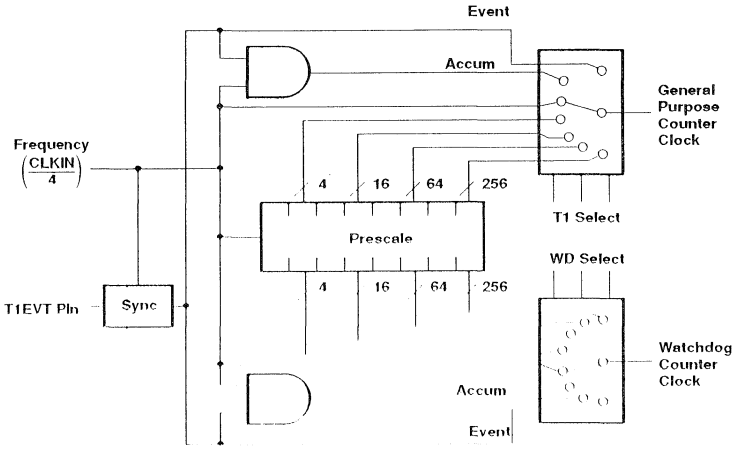


Figure 8. Timer 1 Counter Prescaler

timer 1 general purpose timer

The 16-bit general purpose timer (T1) is composed of a 16-bit resettable counter, a 16-bit compare register and associated compare logic, and a 16-bit register that functions as a capture register in one mode and a compare register in the other mode. The T1 MODE bit selects whether T1 operates in the capture/compare mode or the dual compare mode.

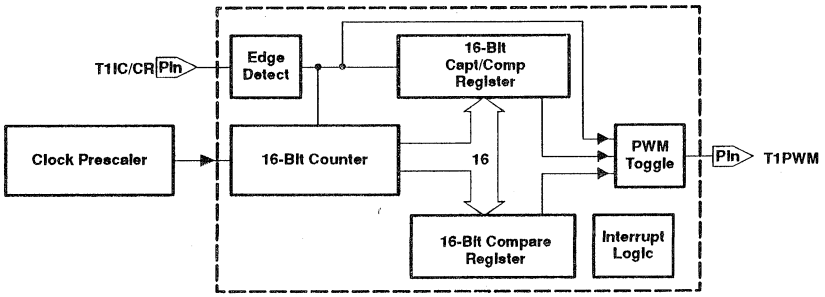


Figure 9. Timer 1 — General Purpose Timer

The counter is a free-running, 16-bit up-counter, clocked by the output of the prescaler/clock source. During initialization the counter is loaded with 0000h and begins its up-count. If the counter is not reset before reaching FFFFh, the counter will roll over to 0000h and continue counting. Upon counter roll-over, the T1 OVRFL INT FLAG is set to 1, and a timer interrupt is generated if the T1 OVRFL INT ENA bit is set to 1.

The counter may be reset to 0000h during counting by either; 1) writing a 1 to the T1 SW RESET bit, 2) a compare equal condition from the dedicated T1 compare function, or 3) an external pulse on the T1IC/CR pin (dual compare mode). The designer may select via software (T1EDGE POLARITY bit) which external transition, low-to-high or high-to-low, on the T1IC/CR pin will cause the counter to be reset.

Special circuitry prevents the 16-bit registers, including the Counter, Compare, or Capture registers, from changing in the middle of a 16-bit read or write operation. When reading a 16-bit register, read the LSB first, then read the MSB. When writing to a 16-bit register, write the MSB first, then write the LSB. The register value will not change between reading or writing the bytes when done in this order.

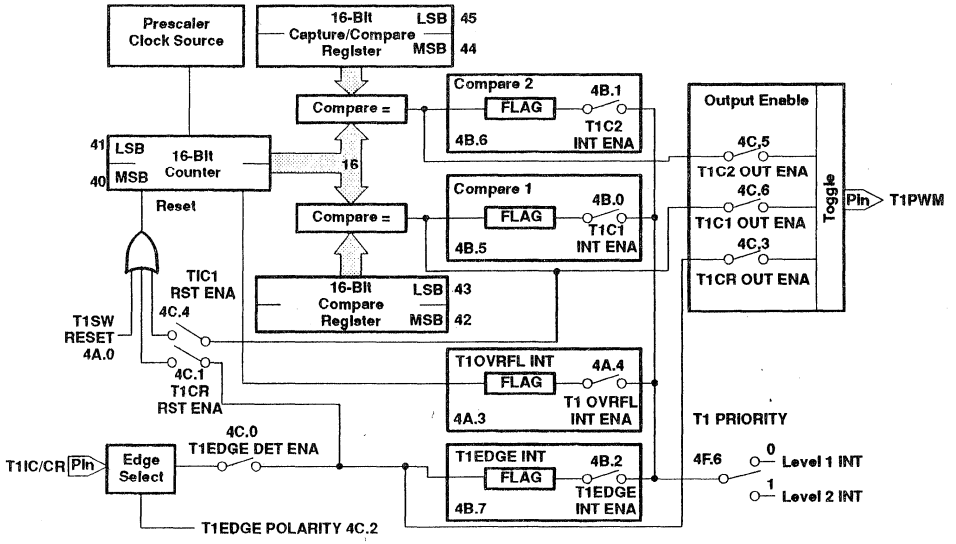
The timer 1 module has three I/O pins used for the functions shown in the following table. Any of these three pins not used in a timer application may be individually configured as general purpose digital I/O pins by the timer 1 module port control registers (T1PC1 and T1PC2).

Timer 1 Module I/O Pin Functions

PIN	DUAL COMPARE MODE	CAPTURE/COMPARE MODE
T1IC/CR	Counter reset input	Input capture input
T1PWM	PWM output	Compare output
T1EVT	External event input or pulse accumulate input	External event input or pulse accumulate input

The **Dual Compare mode** (T1 MODE = 0) provides two compare registers, an external resettable counter, and a timer output pin. These allow the timer to act as an interval timer, a PWM output, simple output toggle, or many other timer functions. The dual compare mode as shown in Figure 10 continuously compares the contents of the two compare registers to the current value of the 16-bit counter. If a timer compare register equals the counter, the circuit sets the associated interrupt flag to 1 and toggles the T1PWM output pin if enabled, and/or generates a Timer 1 interrupt. An output compare equal condition from the dedicated compare register can also initiate a counter reset. A programmable interval timer function, selected by using the compare equal condition to generate a system interrupt and the counter reset function, generates a periodic interrupt.

Either compare function may be used to toggle the T1PWM output pin when a timer compare equal occurs, while the other compare function may be used for another system timing function. Using both compare functions to control the T1PWM pin allows direct PWM generation with minimal CPU software overhead. In typical PWM applications, the compare register is written with the periodic interval and is configured to allow counter reset on compare equal, and the capture/compare register is written with the pulse width to be generated within that interval. The program pulse width may be changed by the application program during the timer operation to alter the PWM output. For high-speed control applications, a minimum pulse width of 200 ns and a period as low as 400 ns can be maintained when using a clock of 20 MHz.

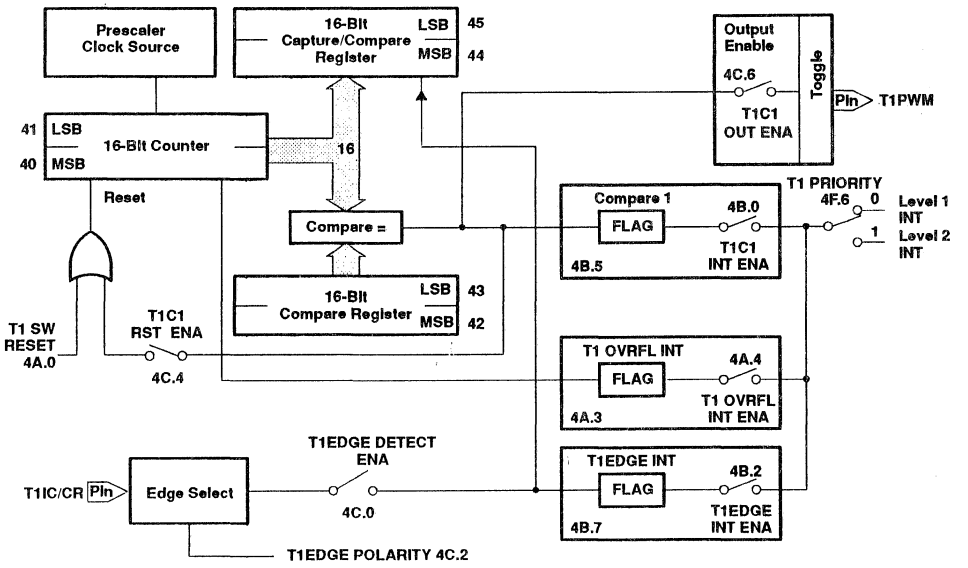


NOTE 4: The numbers on the diagram, such as 4C.0, identify the register and the bit in the peripheral frame. For example, the actual address of 4C.0 is 104Ch, bit 0, in the T1CTL4 register.

Figure 10. Timer 1 — Dual Compare Mode

In addition, a PWM output that is initiated by a transition on an external pin is provided by the timer hardware to support time-critical control applications. Typically, in these applications an external input (T1IC/CR) is used to reset the counter, generate a timer interrupt, and toggle the T1PWM pin to start the PWM output. The compare function will then toggle the output after the programmed pulse width has elapsed. The input edge detect function is enabled under program control by the T1CR DET ENA bit, and upon the next occurrence of the selected edge transition, the T1EDGE INT FLAG bit is set to 1, a timer interrupt is generated (if T1EDGE INT ENA = 1), and the T1PWM output pin is toggled (if T1CR OUT ENA = 1). Selection of the active input transition is under control of T1EDGE POLARITY. In the dual compare mode, the edge detect function must be re-enabled after each valid edge detect.

In the **Capture/Compare mode** (T1 MODE = 1), T1 is configured to provide one input capture register for external timing and pulse width measurement, and one compare register for use as a programmable interval timer. The compare register in this mode functions the same as in the dual compare mode described above, including the ability to toggle the PWM pin. The capture/compare register functions in this mode as a 16-bit input capture register, as shown in Figure 11. On the occurrence of a valid input on the T1IC/CR pin, the current counter value is loaded into the 16-bit input capture register, the T1EDGE INT FLAG is set to 1, and a timer interrupt is generated (if TIEDGE INT ENA = 1). The input detect function is enabled by the T1EDGE DET ENA bit, with T1EDGE POLARITY selecting the active input transition. In the capture/compare mode, the edge detect function, once enabled, remains enabled following a valid edge detect.



NOTE 4: The numbers on the diagram, such as 4C.0, identify the register and the bit in the peripheral frame. For example, the actual address of 4C.0 is 104Ch, bit 0, in the T1CTL4 register.

Figure 11. Timer 1 — Capture/Compare Mode

timer 1 module, watchdog timer

The watchdog timer, contained in the timer 1 module, is a free-running 16-bit resettable up-counter clocked by the output of the Prescaler/Clock Source. The timer is software configured as either a watchdog timer to protect against system software failures and errors, or as a general purpose timer if the watchdog function is not desired. The 16-bit up-counter is programmable (via the WD OVRFL TAP SEL bit) to set the initial count at either 0000h or 8000h. The current value of the watchdog timer may be read at any time during its operation.

In the **watchdog mode** [WD OVRFL RST ENA = 1(high)], the timer will generate a system reset if the timer is re-initialized by an incorrect value or if the counter overflows. The required re-initialization frequency is determined by the system clock frequency, the prescaler/clock source selected, and whether the WD OVRFL TAP SEL bit is set for a 15- or a 16-bit counter rollover. With a clock = 20 MHz, the watchdog timer overflow rates range form 6.55 ms to 3.35 seconds. These values are selected prior to entering the watchdog mode because once the software enables the watchdog reset function (WD OVRFL RST ENA set to 1), subsequent writes to these control bits are ignored. Writes to these watchdog control bits can occur only following a powerup reset, which enhances watchdog timer system integrity.

The watchdog timer is re-initialized by writing a predefined value to the watchdog reset key (WDRST) located in the peripheral file. The proper reset key alternates between 55h and AAh, beginning with 55h following the enable of the watchdog reset function. Writes of the correct value must occur prior to the timer overflow period. A write of any value other than the correct predefined value to the watchdog reset key will be interpreted as a lost program and a system reset will be initiated. A watchdog timer overflow or incorrect reset key will set the WD OVRFL INT FLAG bit to 1 and may be interrogated by the program following system reset to determine the source of the reset.

In the **non-watchdog mode** (WD OVRFL RST ENA = 0), the watchdog timer may be used as an event counter, pulse accumulator, or as an interval timer. In this mode, the system reset function is disabled. The watchdog counter is re-initialized by writing any value to the watchdog reset key (WDRST). When used as an interval timer, the timer overflow interval is determined by the system clock frequency, the prescaler/clock source value selected, and the value of the WD OVRFL TAP SEL bit. If the WD counter is not reset before overflowing, the counter will roll over to either 0000h or 8000h, as determined by the WD OVRFL TAP SEL bit, and continue counting. Upon counter overflow, the WD OVRFL INT FLAG is set to 1 and a timer interrupt is generated if the WD OVRFL INT ENA bit set to 1. Alternately, an external input on the T1EVT pin may be used with the watchdog timer to provide an additional 16-bit event counter or pulse accumulator.

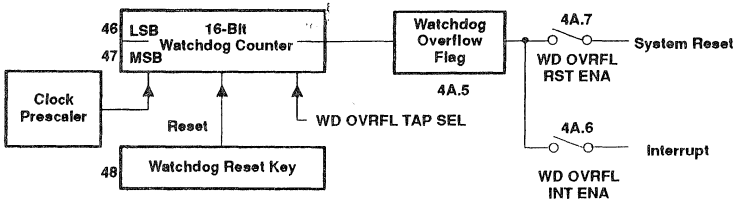


Figure 12. Watchdog/General Purpose Timer

Peripheral File Frame 4: Timer 1 Module Control Registers†

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1040h	P040	Bit 15			Counter MSB				Bit 8	
1041h	P041	Bit 7			Counter LSB				Bit 0	T1CNTR
1042h	P042	Bit 15			Compare Register MSB				Bit 8	
1043h	P043	Bit 7			Compare Register LSB				Bit 0	T1C
1044h	P044	Bit 15			Capture/Compare Register MSB				Bit 8	
1045h	P045	Bit 7			Capture/Compare Register LSB				Bit 0	T1CC
1046h	P046	Bit 15			Watchdog Counter MSB				Bit 8	
1047h	P047	Bit 7			Watchdog Counter LSB				Bit 0	WDCNTR
1048h	P048	Watchdog Reset Key								WDRST
1049h	P049	WD OVRFL TAP SEL‡	WD INPUT SELECT2‡	WD INPUT SELECT1‡	WD INPUT SELECT0‡	—	T1INPUT SELECT2	T1INPUT SELECT1	T1INPUT SELECT0	T1CTL1
104Ah	P04A	WD OVRFL RST ENA‡	WD OVRFL INT ENA	WD OVRFL INT FLAG	T1 OVRFL INT ENA	T1 OVRFL INT FLAG	—	—	T1 SW RESET	T1CTL2
Mode: Dual Compare										
104Bh	P04B	T1EDGE INT FLAG	T1C2 INT FLAG	T1C1 INT FLAG	—	—	T1EDGE INT ENA	T1C2 INT ENA	T1C1 INT ENA	T1CTL3
104Ch	P04C	T1 MODE = 0	T1C1 OUT ENA	T1C2 OUT ENA	T1C1 RST ENA	T1CR OUT ENA	T1EDGE POLARITY	T1CR RST ENA	T1EDGE DET ENA	T1CTL4
Mode: Capture/Compare										
104Bh	P04B	T1EDGE INT FLAG	—	T1C1 INT FLAG	—	—	T1EDGE INT ENA	—	T1C1 INT ENA	T1CTL3
104Ch	P04C	T1 MODE = 1	T1C1 OUT ENA	—	T1C1 RST ENA	—	T1EDGE POLARITY	—	T1EDGE DET ENA	T1CTL4
104Dh	P04D	—	—	—	—	T1EVT DATA IN	T1EVT DATA OUT	T1EVT FUNCTION	T1EVT DATA DIR	T1PC1
104Eh	P04E	T1PWM DATA IN	T1PWM DATA OUT	T1PWM FUNCTION	T1PWM DATA DIR	T1IC/CR DATA IN	T1IC/CR DATA OUT	T1IC/CR FUNCTION	T1IC/CR DATA DIR	T1PC2
104Fh	P04F	**T1 STEST**	**T1 PRIORITY**	—	---	—	—	—	—	T1PRI

† Privileged bits are shown in **bold typeface.**
‡ Once the WD OVRFL RST ENA bit is set, these bits cannot be changed until after a full power-down cycle has been completed.

The formulas in Figure 13 show the calculations for the resulting time, given values in the compare registers T1C and T1CC.

$$\text{time} = \left(\frac{4}{\text{CLKIN}}\right) (\text{prescale}) (\text{compare} + 1)$$

or

$$\text{time} = t_c (\text{prescale}) (\text{compare} + 1)$$

Figure 13. Timer 1 Compare Register Formulas

timer 2 module

Timer 2 consists of a clock source block and a 16-bit general purpose timer that provides the event count, input capture, and compare functions.

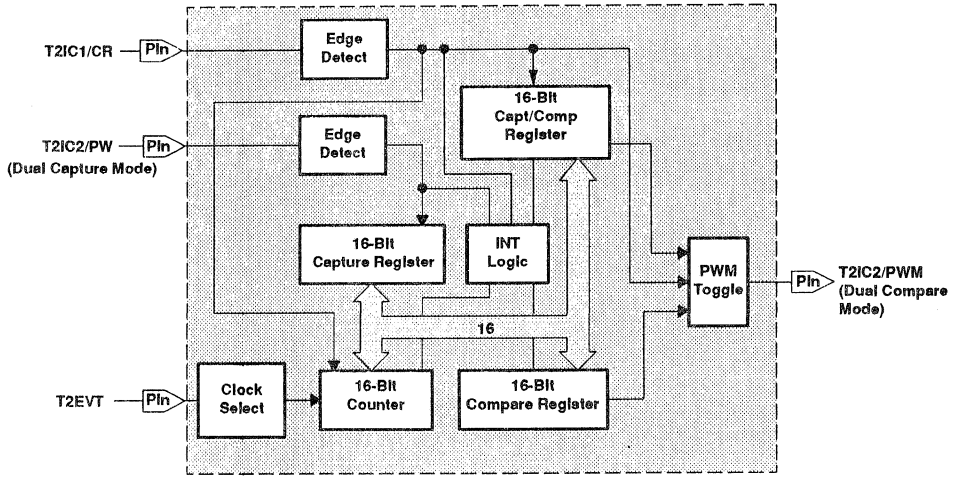


Figure 14. Timer 2 Module Block Diagram

timer 2 clock source

The clock source input for the general purpose timer is configured by the T2 INPUT SELECT control bits of the T2CTL1 control register. The four programmable clock sources for the general purpose counter are system clock, pulse accumulate, event input, or no clock input (counter stopped). When using the system clock input, the 16-bit timer generates an overflow rate of 13.1 ms with 200 ns resolution (clock = 20 MHz).

In the **event counter mode**, the general purpose timer is programmable as a 16-bit event counter. An external low-to-high transition on the T2EVT pin is used to provide the clock for the internal timer. The T2EVT external clock frequency may not exceed the system clock frequency divided by 2.

In the **pulse accumulate mode**, the general purpose timer is programmable as a 16-bit pulse accumulator. An external input on the T2EVT pin is used to gate the internal system clock to the internal timers. While T2EVT input is logic 1, the timers will be clocked at the system clock rate and will count system clock pulses until the T2EVT pin returns to logic zero.

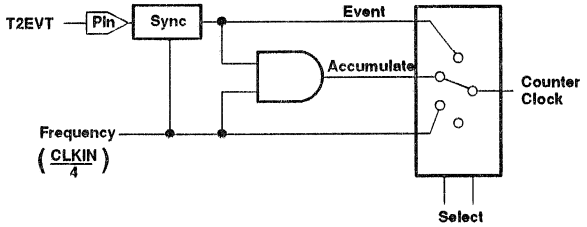


Figure 15. Timer 2 Clock Select

timer 2 general purpose timer

The 16-bit general purpose timer, T2 , is composed of a 16-bit resettable counter, a 16-bit compare register with associated compare logic, a 16-bit capture register, and a 16-bit register that functions as a capture register in one mode and a compare register in the other mode. The T2 MODE bit selects whether T2 operates in the dual compare mode or the Dual Capture mode.

The counter is a free-running 16-bit up-counter, clocked by the system clock, external event, or system clock while external event active (pulse accumulate). During initialization, the counter is loaded with 0000h and begins its up-count. If the counter is not reset before reaching FFFFh, the counter will roll over to 0000h and continue counting. Upon counter roll-over, the T2 OVRFLINT FLAG is set to 1, and a timer interrupt is generated if the T2 OVRFL INT ENA bit is set to 1.

The counter may be reset to 0000h during counting by either; 1) writing a 1 (high) to the T2 SW RESET bit, 2) a compare equal condition from the dedicated T2 compare function, or 3) an external pulse on the T2IC1/CR pin (Dual Compare mode). The designer may select via software (T2CR POLARITY bit) which external transition, low-to-high or high-to-low, on the T2IC1/CR pin will cause the counter to be reset.

Special circuitry prevents the 16-bit registers, including the Counter, Compare, or Capture registers, from changing in the middle of a 16-bit read or write operation. When reading a 16-bit register, read the LSB first, then read the MSB. When writing to a 16-bit register, write the MSB first, then write the LSB. The register value will not change between reading or writing the bytes when done in this order.

Timer 2 has three I/O pins used for functions as shown in the table below. Any of these three pins not used in a timer application may be individually configured as general purpose digital I/O pins by the Timer 2 port control registers (T2PC1 and T2PC2).

Timer 2 I/0 Pin Functions

PIN	DUAL COMPARE MODE	CAPTURE/COMPARE MODE
T2IC1/CR	Counter reset input.	Input Capture 1 input.
T2IC2/PWM	PWM output.	Input Capture 2 input.
T2EVT	External event input or pulse accumulate input.	External event input or pulse accumulate input.

The **Dual Compare mode** (T2 MODE = 0) provides two compare registers, an external resettable counter, and a timer output pin. These allow the timer to act as a interval timer, a PWM output, simple output toggle, or many other timer functions. In this mode, the capture/compare register functions as a 16-bit read/write compare register, as shown in Figure 16. The operation of T2 is identical to T1 while operating in the dual compare mode.

In the **Dual Capture mode** (T2 MODE = 1), T2 is configured to provide one compare register for use as a programmable interval timer, and two input capture registers for external input timing and pulse width measurement. In this mode the capture/compare register functions as a 16-bit input capture register, as shown in Figure 17. Each capture input pin (T2IC1/CR and T2IC2/PWM) has an input edge detect function enabled by the associated DET ENA control bit, with the associated POLARITY bit selecting the active input transition. On the occurrence of a valid input on the T2IC1/CR or T2IC2/PWM pin, the current counter value is loaded into the 16-bit capture/compare and 16-bit input capture register, respectively. In addition, the respective input capture INT FLAG is set to 1 and a timer interrupt is generated if the respective INT ENA is set to 1.

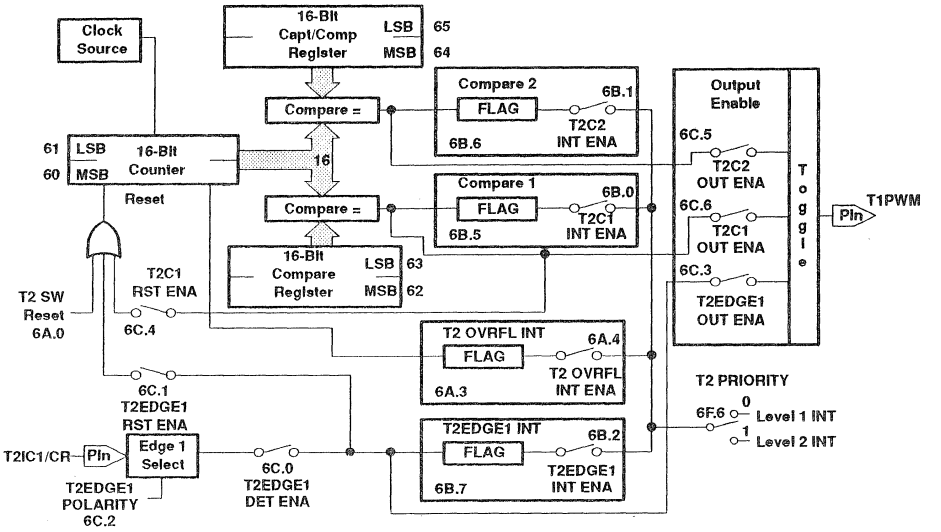


Figure 16. Timer 2 - Dual Compare Mode

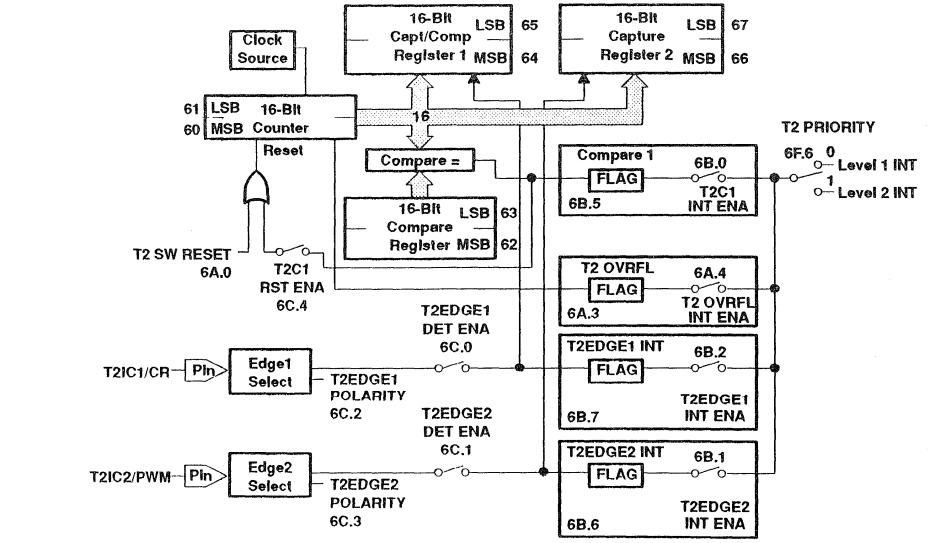


Figure 17. Timer 2 - Dual Capture Mode

Peripheral File Frame 6: Timer 2 Module Control Registers†

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1060h	P060	Bit 15			T2 Counter MSB				Bit 8	T2CNTR
1061h	P061	Bit 7			T2 Counter LSB				Bit 0	
1062h	P062	Bit 15			T2 Compare Register MSB				Bit 8	T2C
1063h	P063	Bit 7			T2 Compare Register LSB				Bit 0	
1064h	P064	Bit 15			T2 Capture/Compare Register MSB				Bit 8	T2CC
1065h	P065	Bit 7			T2 Capture/Compare Register LSB				Bit 0	
1066h	P066	Bit 15			T2 Capture Register 2 MSB				Bit 8	T2IC
1067h	P067	Bit 7			T2 Capture Register 2 LSB				Bit 0	
1068h 1069h	P068 P069	Reserved								
106Ah	P06A	—	—	—	T2 OVRFL INT ENA	T2 OVRFL INT FLAG	T2 INPUT SELECT1	T2 INPUT SELECT0	T2 SW RESET	T2CTL1
Mode: Dual Compare										
106Bh	P06B	T2EDGE1 INT FLAG	T2C2 INT FLAG	T2C1 INT FLAG	—	—	T2EDGE1 INT ENA	T2C2 INT ENA	T2C1 INT ENA	T2CTL2
106Ch	P06C	T2 MODE = 0	T2C1 OUT ENA	T2C1 OUT ENA	T2C2 OUT ENA	T2C1 RST ENA	T2EDGE1 POLARITY	T2EDGE1 RST ENA	T2EDGE1 DET ENA	T2CTL3
Mode: Dual Capture										
106Bh	P06B	T2EDGE1 INT FLAG	T2EDGE2 INT FLAG	T2C1 INT FLAG	—	—	T2EDGE1 INT ENA	T2EDGE2 INT ENA	T2C1 INT ENA	T2CTL2
106Ch	P06C	T2 MODE = 1	—	—	T2C1 RST ENA	T2EDGE2 POLARITY	T2EDGE1 POLARITY	T2EDGE2 DET ENA	T2EDGE1 DET ENA	T2CTL3
106Dh	P06D	—	—	—	—	T2EVT DATA IN	T2EVT DATA OUT	T2EVT FUNCTION	T2EVT DATA DIR	T2PC1
106Eh	P06E	T2IC2/PWM DATA IN	T2IC2/PWM DATA OUT	T2IC2/PWM FUNCTION	T2IC2/PWM DATA DIR	T2IC1/CR DATA IN	T2IC1/CR DATA OUT	T2IC1/CR FUNCTION	T2IC1/CR DATA DIR	T2PC2
106Fh	P06F	**T2 STEST**	**T2 PRIORITY**	—	—	—	—	—	—	T2PRI

† Privileged bits are shown in **bold typeface.**

The formulas in Figure 18 show the calculations for the resulting time, given values in the compare registers T2C and T2CC.

$$\text{time} = \left(\frac{4}{\text{CLKIN}}\right) (\text{prescale}) (\text{compare} + 1)$$

or

$$\text{time} = t_c (\text{prescale}) (\text{compare} + 1)$$

Figure 18. Timer 2 Compare Register Formulas

serial communications interface (SCI)

The Serial Communications Interface (SCI) is a full-duplex serial I/O port that supports standard NRZ serial communications in a programmed data format (start bit, 1 to 8 data bits, parity even/odd/off, one or two stop bits) at a variety of programmable baud rates. High-speed isosynchronous communications, as well as standard asynchronous communications, are supported for interfacing to peripheral devices. The isosynchronous communications mode combines features of the asynchronous mode with a synchronizing clock signal. The isosynchronous mode has the same format as the asynchronous mode using start, stop, parity, and data bits, but it it uses one serial clock cycle per bit to achieve a much higher transmission speed. Multiprocessor communications using idle line wake-up and address bit wake-up protocols are also supported by the SCI transmit and receive hardware.

As shown in Figure 19, the SCI receiver and transmitter are double buffered to reduce the possibility of overwriting data prior to the previous data being read or transmitted from the SCI.

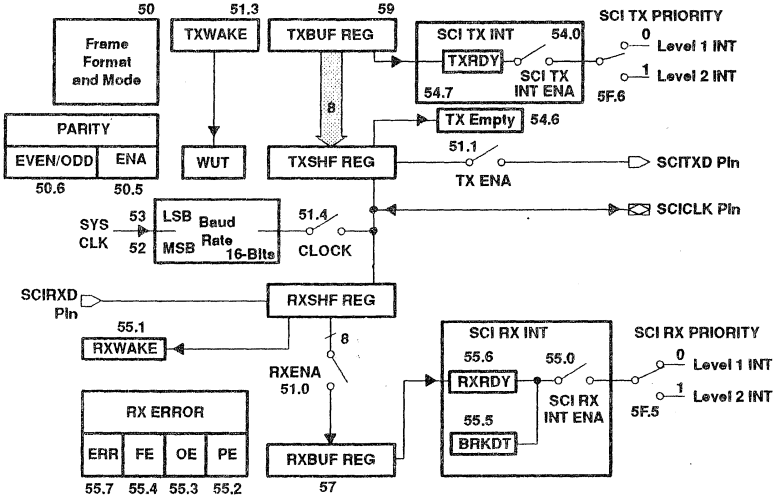


Figure 19. SCI Block Diagram

The SCI provides independent interrupt requests and vectors for the receiver and transmitter. Interrupts requested by the SCI receiver and SCI transmitter can be software programmed onto different priority levels by the SCI RX PRIORITY and SCI TX PRIORITY control bits. When SCI interrupt requests are made on the same level, the receiver always has higher priority than the transmitter to reduce the possibility of receiver overrun. An SCI TXINT interrupt is asserted whenever TXBUF is transferred to TXSHF. An SCI RXINT interrupt is asserted whenever the SCI receives a complete frame (RXSHF transfers to RXBUF) or when a break detect condition occurs (SCIRXD is low for 10 bit periods following a stop bit).

If the TMS370Cx4x has been placed in HALT or STANDBY low-power mode with the SCI RX INT ENA bit = 1 (high), the detection of the start bit (one-to-zero transition) by the SCI receiver initiates receipt of the SCI input, exits the low power mode, activates the microcontroller (CPU, clocks, on-chip peripherals), and initiates execution of the SCI RXINT interrupt service routine. To ensure valid data receipt of the first frame, the baud rate must be slow enough for the SCI to sample for a valid start bit after exiting from the power down mode, or the first data byte must be ignored.

The SCI transmitter and receiver are functionally independent to support full-duplex communications; however, they use the same data format, baud rate, communications mode, and multiprocessor communications protocol. The SCICCR control register selects the transmit and receive data format. Figure 20 shows the SCI data format of one frame of information, which consists of an idle line (logic 1), one start bit (logic 0), one to eight data bits, an address bit (if in address bit wake-up mode), a parity bit (if enabled), and one or two stop bits (logic 1). The character length of one to eight data bits is selected by the SCI CHAR2, SCI CHAR1, and SCI CHAR0 control bits. Parity on/off is selected by PARITY ENABLE with the EVEN/ODD PARITY bit selecting the type. Parity generation and verification is performed in the SCI hardware, requiring no CPU calculation overhead. One or two stop bits for transmission are selected by the STOP BITS control bit. The receiver checks for one stop bit on incoming data.

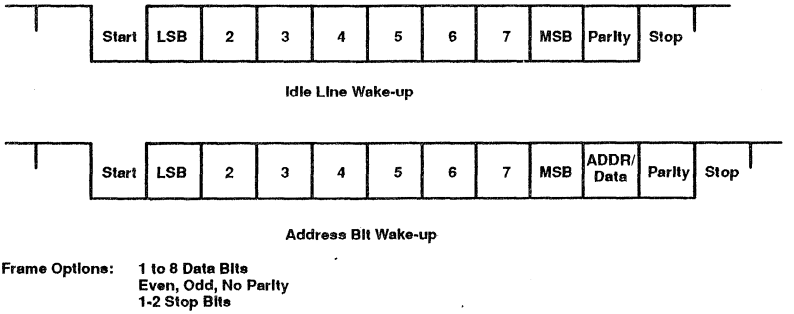


Figure 20. Frame Formats

The SCI communications mode is selected by the ASYNC/ISOSYNC control bit. The transmit and receive data format as described above are identical in both communication modes. In the **asynchronous mode** (ASYNC/ISOSYNC = 1), the external communications interface consists of the SCITXD and SCIRXD pins with an optional SCICLK input for driving the internal SCICLK. The transmit baud rate is 1/16 that of the SCICLK frequency. The receiver internally samples the data input at 16 times the bit rate. The receiver uses majority vote sampling on the seventh, eighth, and ninth SCICLK periods to determine the value of the start bit, data bits, parity, and first stop bit. Asynchronous data rates are supported up to 156K baud (SYSCLK/2^{21} to SYSCLK/32) at 20 MHz.

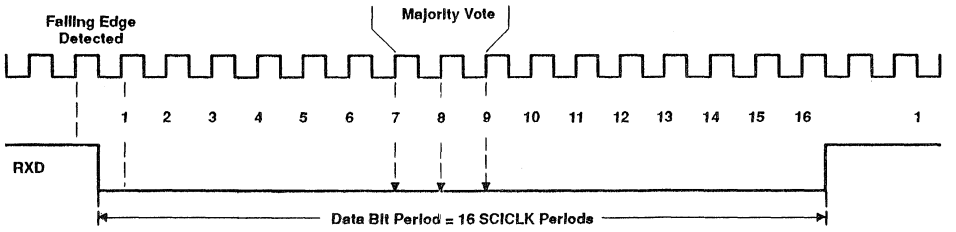


Figure 21. Asynchronous Mode

The **isosynchronous mode** (ASYNC/ISOSYNC = 0) has the same format as the asynchronous mode, consisting of a start bit, one to eight data bits, an even/odd/no parity bit, and one or two stop bits, but uses an additional synchronizing clock to support high speed serial communications. The external system interface consists of the SCITXD and SCIRXD pins and a continuous synchronizing clock on on the SCICLK pin. Isosynchronous transmit and receive data is clocked at a rate equal to the SCICLK rate, and receiver values are read on a single sample basis. Isosynchronous data rates with synchronizing SCICLK are supported up to 2.5M baud (SYSCLK/2^{17} to SYSCLK/2) at 20 MHz.

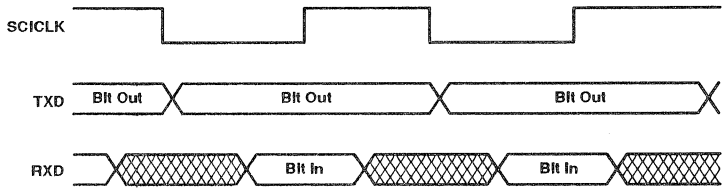


Figure 22. Isosynchronous Mode

The CLOCK bit in SCICTL determines whether the SCI clocking signal comes in from an external source through the SCICLK pin or goes out through SCICLK after generation in the integral baud rate timer. The isosynchronous mode baud rate equals the SCICLK rate; the asynchronous mode baud rate 1/16 the SCICLK rate. The maximum frequency of an external clock source can be no greater than 1/10 the system clock frequency. The frequency of the SCICLK when generated by the internal baud rate timer given by the formula.

$$\text{SCICLK} = \frac{\text{CLKIN}}{8(\text{Baud Rate Reg} + 1)}$$

The baud rate using the internal clock equals the SCICLK rate in the isosynchronous mode and equals 1/16 the SCICLK in the asynchronous mode. The 16-bit baud rate register allows the selection of many different standard baud rates.

$$\text{Asynchronous Baud Rate} = \frac{\text{CLKIN}}{128(\text{Baud Rate Reg} + 1)}$$

$$\text{Isosynchronous Baud Rate} = \frac{\text{CLKIN}}{8(\text{Baud Rate Reg} + 1)}$$

NOTE

When an external serial clock signal is used, the maximum SCICLK frequency is CLKIN/40.

In both asynchronous and isosynchronous modes, if the transmitter is enabled (TXENA = 1), SCI transmission is initiated following a CPU write to the TXBUF register. This sets TX EMPTY to 0; TXSHF is loaded from TXBUF, TXRDY flag is set to 1, and if SCI TX INT ENA is set to 1, SCI transmit interrupt (TXINT) will be asserted. Another write may then be performed to the TXBUF; if not, the transmitter idles (SCITXD outputs continuous high), and TX EMPTY is set to 1 (both TXBUF and TXSHF are empty) until the next write to TXBUF.

In both asynchronous and isosynchronous modes, when a frame is fully received, RXBUF is loaded from RXSHF, the error status bits are set accordingly, RXRDY flag is set to 1, and if SCI RX INT ENA is set to 1, an SCI receiver interrupt (RXINT) will be asserted. The SCI receiver performs extensive error checking during data bit reception and provides individual error flags for parity error (PE), overrun error (OE), framing error (FE), and break detect (BRKDT) for application program querying.

The SCI supports two multiprocessor communication formats to allow efficient transfer of information between many microcontrollers on the same serial data link. Information is typically transferred as a block of data from a source to a destination, with the destination address identified at the beginning of the block. The SCI has the ability to inhibit all SCI receiver flags and interrupts until a start of a block of data (a destination address) is identified. When a block start is identified, the SCI initiates the following sequence for both multiprocessor communication formats:

1. The serial port wakes up at the start of the block and receives the first frame (containing the destination address).
2. A software routine responds to the SCI receiver interrupt and checks the incoming byte against its address byte stored in memory.
3. If the block is addressed to the microcontroller, the SCI remains active and the CPU reads the rest of the block. If the address does not compare, the software routine puts the serial port to sleep and the SCI will inhibit all SCI receiver flags and interrupts until the next block start.

To provide system flexibility, the SCI, in both asynchronous and isosynchronous modes, recognizes the idle line wake-up and address bit wake-up multiprocessor protocols. The multiprocessor protocol is selected by the ADDRESS/IDLE WUP control bit in the SCICCR register. Both protocols use the SLEEP and TXWAKE bits to control the receive and transmit features of the wake-up mode, and the RXWAKE status bit to provide the receiver wake-up condition.

In **idle line wake-up**, blocks are separated by having a longer idle time (logic one) between the blocks than between frames within the blocks. As shown in Figure 23, an idle time of 10 or more bits after a frame indicates a start of a new block and wakes up all receivers. Under software control, all receivers that do not recognize the address in the first frame of the message ignore the rest of the message and await the next idle line. The SCI transmitter allows an idle time of exactly one frame to be transmitted to indicate the start of the next block to maintain serial data link efficiency by minimizing the idle time between block starts. Idle line wake-up protocol has no overhead within the message frames and is typically used when transferring large blocks of data.

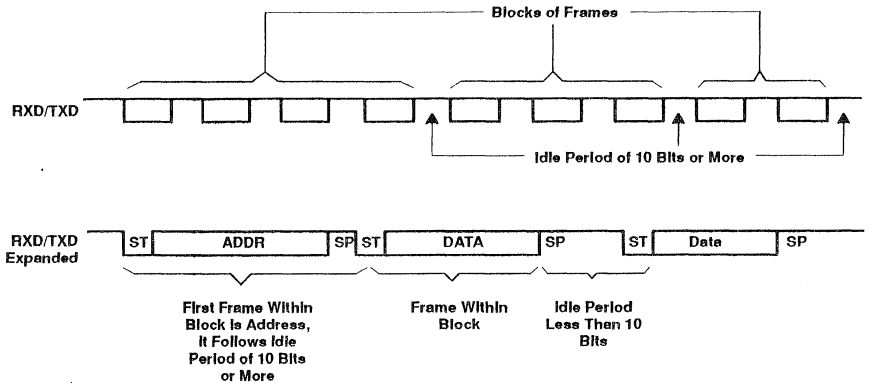


Figure 23. Idle Line Multiprocessor Mode

In **address bit wake-up**, each frame has an extra bit, the ADDR/DATA bit, positioned just before the parity bit (if used). As shown in Figure 24, block starts are distinguished by the ADDR/DATA bit set to 1 in the first frame of the block and all subsequent frames of the block have the ADDR/DATA bit set to 0. The start of the next block is identified by the next frame that has a 1 in ADDR/DATA. The idle line time is irrelevant in this protocol. All receivers wake up upon receiving a frame with ADDR/DATA set to 1. Under software control, all receivers that do not recognize their address in the first frame of the message ignore the rest of the message and await the next active ADDR/DATA bit. Address bit wake-up protocol eliminates interblock gaps and is efficient in transferring many small blocks of data.

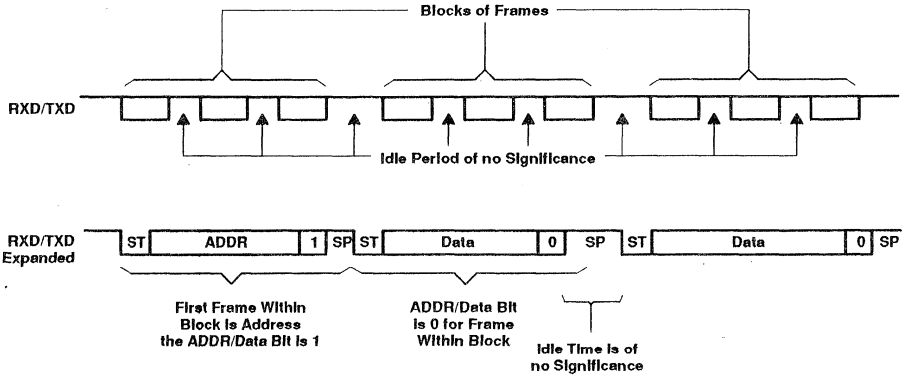


Figure 24. Address Bit Multiprocessor Mode

Peripheral File Frame 5: Serial Communication Interface (SCI) Control Registers†

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1050h	P050	STOP BITS	EVEN/ODD PARITY	PARITY ENABLE	ASYNC/ ISOSYNC	ADDRESS IDLE WUP	SCI CHAR2	SCI CHAR1	SCI CHAR0	SCICCR
1051h	P051	—	—	SCI SW RESET	CLOCK	TXWAKE	SLEEP	TXENA	RXENA	SCICTL
1052h	P052	Bit 15	Baud Rate Select Register MSB						Bit 8	BAUD MSB
1053h	P053	Bit 7	Baud Rate Select Register LSB						Bit 0	BAUD LSB
1054h	P054	TXRDY	TX EMPTY	—	—	—	—	—	SCI TX INT ENA	TXCTL
1055h	P055	RX ERROR	RXRDY	BRKDT	FE	OE	PE	RXWAKE	SCI RX INT ENA	RXCTL
1056h	P056	Reserved								
1057h	P057	Receive Data Buffer Register								RXBUF
1058h	P058	Reserved								
1059h	P059	Transmit Data Buffer Register								TXBUF
105Ah	P05A	Reserved								
105Bh	P05B									
105Ch	P05C									
105Dh	P05D	—	—	—	—	SCICLK DATA IN	SCICLK DATA OUT	SCICLK FUNCTION	SCICLK DATA DIR	SCIPC1
105Eh	P05E	SCI TXD DATA IN	SCI TXD DATA OUT	SCI TXD FUNC-TION	SCI TXD DATA DIR	SCI RXD DATA IN	SCI RXD DATA OUT	SCI RXD FUNCTION	SCI RXD DATA DIR	SCIPC2
105Fh	P05F	**SCI STEST**	**SCI TX PRIORITY**	**SCI RX PRIORITY**	**SCI ESPEN**	—	—	—	—	SCIPRI

† Privileged bits are shown in **bold typeface.**

analog-to-digital converter

The 8-bit analog-to-digital (A/D) converter provides the designer with eight multiplexed analog input channels for the 44-pin device and four multiplexed analog input channels for the 40-pin device. The A/D converter has internal sample and hold circuitry and uses a successive approximation conversion technique. The accuracy of the A/D conversion process is increased by providing separate analog positive supply and analog ground input pins (V_{CC3} and V_{SS3}). The V_{SS3} pin also provides the low reference voltage input for the conversion process.

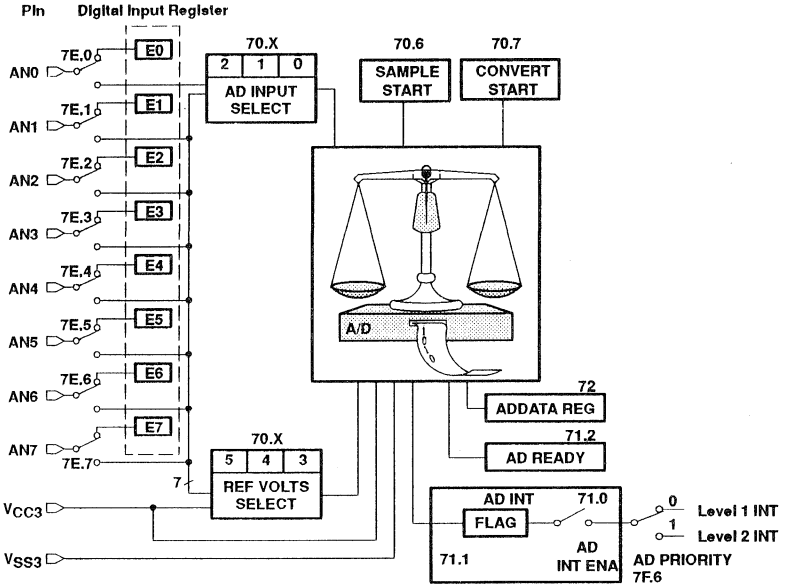


Figure 25. A/D Converter Block Diagram

The A/D converter high reference voltage input is software selectable as one of eight positive reference inputs, as shown in the table below. The A/D conversion process is ratiometric, using V_{SS3} and the software-selected high-reference voltage input as the limits for the selected analog input channel. An input voltage equal to or greater than the high reference input converts to FFh (full scale) with no overflow. An input voltage equal to or less than V_{SS3} converts to 00h. Ratiometric conversions allow analog inputs to be scaled against selected high reference inputs to achieve the greatest accuracy.

A/D INPUT			ANALOG
SEL2	SEL1	SEL0	INPUT CHANNEL
0	0	0	AN0†
0	0	1	AN1†
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4†
1	0	1	AN5†
1	1	0	AN6
1	1	1	AN7

REFERENCE VOLTAGE			HIGH REFERENCE
SEL2	SEL1	SEL0	INPUT
0	0	0	V_{CC3}
0	0	1	AN1‡
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4‡
1	0	1	AN5‡
1	1	0	AN6
1	1	1	AN7

† On the 40-pin devices AN0, AN1, AN4, and AN5 are not implemented.
‡ Cannot be used as a reference channels on 40-pin devices.

The four analog pins (AN0, AN1, AN4 and AN5) that are not implemented on the 40-pin parts are connected to GND internally. If these pins are read as a digital input they will read 0. If they are read as an analog channel they will read 00. Channels AN1, AN4 and AN5 may not be used as the reference channel since V_{ref} must be a minimum of 2.5 volts.

To read an A/D channel:

1. Write to the ADCTL peripheral file control register to:
   - Select the high reference voltage input (ADCTL.5-3).
   - Select the analog input channel for conversion (ADCTL.2-0).
   - Set the SAMPLE START bit to 1 (ADCTL.6).
2. Wait for the sample time to elapse.
3. Write to the ADCTL peripheral file control register to:
   - Set the CONVERT START bit to 1 and leave SAMPLE START bit set to 1.
4. Wait for either the interrupt flag to be set or the A/D interrupt to occur.
5. Read the conversion value from ADDATA when AD INT FLAG is set to 1 or the A/D interrupt occurs.
6. Clear the interrupt flag (ADSTAT.1).

To provide the designer with the flexibility to optimize the A/D conversion process with both high and low impedance sources, the sample time is independently defined by the application program. At the completion of the sample time, the conversion is initiated by setting the CONVERT START and SAMPLE START bits to 1. Eighteen clock cycles after the CONVERT START bit is set to 1, the CONVERT START and SAMPLE START bits will both be set to 0 by the A/D converter, indicating the conversion has started and the analog input signal can be removed. The AD READY bit is set to 0 by the A/D converter to indicate a conversion is in progress. The conversion is complete 164 system clock cycles after it is initiated by setting the CONVERT START bit to 1, and the result is located in the ADDATA result register. Upon completion of the conversion, the AD INT FLAG bit is set, and if the AD INT ENA bit is set to 1 an interrupt will be asserted.

The A/D converter has eight bits of resolution with absolution with absolute accuracy of plus or minus one LSB, with (High Reference Voltage – V_{SS3}) = 5 V.

To maximize I/O control capability, all analog input pins not used for an analog input or high reference voltage input may be individually configured as general purpose digital input pins. The control and input data values are contained in the ADENA and ADIN peripheral file control registers.

Peripheral File Frame 7: A-to-D Converter Control Registers†

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1070h‡	P070	CONVERT START	SAMPLE START	REF VOLT SELECT2	REF VOLT SELECT1	REF VOLT SELECT0	AD INPUT SELECT2	AD INPUT SELECT1	AD INPUT SELECT0	ADCTL
1071h	P071	—	—	—	—	—	AD READY	AD INT FLAG	AD INT ENA	ADSTAT
1072h	P072	A-to-D Conversion Data Register								ADDATA
1073h to 107Ch	P073 to P07C	Reserved								
107Dh	P07D	Port E Data Input Register								ADIN
107Eh	P07E	Port E Input Enable Register								ADENA

107Fh	P07F	**AD STEST**	**AD PRIORITY**	**AD ESPEN**	—	—	—	—	—	ADPRI

† Privileged bits are shown in **bold typeface.**

Instruction set

The TMS370Cx4x family instruction set consists of 64 instructions that control input, output, data manipulations, data comparisons, and program flow. The instruction set is supported with 14 addressing modes to provide the flexibility to optimize programs to the user's applications. For example, the MOV instruction has 27 operand combinations supported by its addressing modes.

ADDRESSING MODE	EXAMPLE		OPERATION
GENERAL:			
Implied	LDSP		(B) → (SP)
Register	MOV	R5,R4	(R5) → (R4)
Peripheral	MOV	P025,A	(1025) → A
Immediate	ADD	#123,R23	123 + (R3) → (R3)
PC Relative	JMP	offset	PCN + offset → (PC)
Stack Pointer Relative	MOV	2(SP),(A)	(2 + (SP)) → (A)
EXTENDED:			
Absolute Direct	MOV	A,1234	(A) → (1234)
Absolute Indexed	MOV	1234(B),A	(1234 + (B)) → (A)
Absolute Indirect	MOV	@R4,A	((R3:R4)) → (A)
Absolute Offset Indirect	MOV	12(R4),A	(12 + (R3:R4)) → (A)
Relative Direct	JMPL	1234	PCN + 1234 → (PC)
Relative Indexed	JMPL	1234(B)	PCN + 1234 + (B) → (PC)
Relative Indirect	JMPL	@R4	PCN + (R3:R4) → (PC)
Relative Offset Indirect	JMPL	12(R4)	PCN + 12 + (R3:R4) → (PC)

PCN = 16-bit address of next instruction.
(x) = Contents of memory at address x.
((x)) = Contents of memory location designated by contents at address x.

The CPU controls instruction execution by executing microinstructions from a dedicated control memory at a rate of one microinstruction per internal system clock cycle, t_c. The number of system clock cycles required to execute one assembly language instruction varies depending on the instruction complexity, operand addressing mode, and number of wait states. Instruction execution times are stated in terms of the number of internal system clock cycles per instruction. Instruction execution times vary from 5 to 63 internal system clock cycles, with most instructions requiring less than 10 cycles to complete.

Similarly, the number of bytes of program memory required to store an instruction will vary with instruction complexity and addressing mode. TMS370 instructions require from one to five bytes of program memory space, with most instructions occupying one or two bytes.

The table *TMS370 Instruction Set Summary*, beginning on page 39, shows the instruction set, the addressing modes, the program memory byte length, and the execution cycle count for each instruction. The addressing mode entries are in the format of BYTE LENGTH/CYCLE COUNT. The following symbols and abbreviations are used:

SYMBOL	DEFINITION	SYMBOL	DEFINITION
s	Source Operand	d/D	Destination Operand (8-bit/16-bit)
A	Register A or R0 in Register File	B	Register B or R1 in Register File
Rs	Source Register in Register File	Rd	Destination Register in Register File
Ps	Source Register in Peripheral File	Pd	Destination Register in Peripheral File
Rps	Source Register Pair (Rn, Rn-1)	Rpd	Destination Register Pair (Rn, Rn-1)
Rp	General Purpose Register Pair	label	16-bit Label
iop8	8-bit Immediate Operand	iop16	16-bit Immediate Operand
off8	8-bit Signed Offset (label – PCN)	off16	16-bit Signed Offset
PC	Program Counter	PCN	16-bit Address of Next Instruction
SP	Stack Pointer	ST	Status Register
#	Immediate Operand	@	Extended Addressing Operand (Direct, Indirect, Indexed)
C	Status Register Carry Bit	→	Is Assigned to
( )	Contents of		

TMS370 Instruction Set Summary

OPERATION		ADDRESSING MODES									OTHER	DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)		
ADC	B, —	1/8										Add with Carry
	Rs, —	2/7	2/7	3/9								(s) + (d) + (C) → (d)
	#lop8, —	2/6	2/6	3/8								
ADD	B, —	1/8										Add
	Rs, —	2/7	2/7	3/9								(s) + (d) → (d)
	#lop8, —	2/6	2/6	3/8								
AND	A, —				2/9							And
	B, —	1/8			2/9							(s) .AND. (d) → (d)
	Rs, —	2/7	2/7	3/9								
	#lop8, —	2/6	2/6	3/8	3/10							
BR						3/9	2/8	3/11	4/16			Branch; D → (PC)
BTJO†	A, —,off8				3/10							Bit Test and Jump If One
	B, —,off8	2/10			3/10							If (s) .AND. (d) ≠ 0 then
	Rs, —,off8	3/9	3/9	4/11								PCN + offset → (PC)
	#lop8, —,off8	3/8	3/8	4/10	4/11							
BTJZ†	A, —,off8				3/10							Bit Test and Jump If Zero
	B, —,off8	2/10			3/10							If (s).AND.(not d) ≠ 0 then
	Rs, —,off8	3/9	3/9	4/11								PCN + offset → (PC)
	#lop8, —,off8	3/8	3/8	4/10	4/11							
CALL	—					3/13	2/12	3/15	4/20			Call; Push PCN, D → (PC)
CALLR	—					3/15	2/14	3/17	4/22			Call Relative Push PCN, PCN + (d) → (PC)
CLR	—	1/8	1/8	2/6								Clear; 0 → (d)
CLRC											1/19	Clear Carry; 0 → (C)
CMP	—,A					3/11	2/10	3/13	4/18	2/8		Compare
	B, —	1/8										(d) – (s) computed and
	Rs, —	2/7	2/7	3/9								Status Register flags set
	#lop8, —	2/6	2/6	3/8								
CMPBIT	—			3/8	3/10							Complement Bit
CMPL	—	1/8	1/8	2/6								Twos complement; 0100h – (s) → (d)
DAC	B, —	1/10										Decimal Add with Carry
	Rs, —	2/9	2/9	3/11								(s) + (d) + (C) → (d) (BCD)
	#lop8, —	2/8	2/8	3/10								
DEC	—	1/8	1/8	2/6								Decrement; (d) – 1 → (d)
DINT											2/6	Disable Interrupt; 00 → (ST)
DIV	Rs, —	3/55/63‡										Integer Divide; 16 by 8-bit A:B/Rs → A(=quo), B(=rem)

DJNZ† __,off8	2/10	2/10	3/8								# cycles depends on operands Decrement and Jump If Not 0 (d) – 1 → (d); If (d) .NE. 0 then PCN + offset → (PC)

† Add 2 to cycle count if jump is taken.
‡ Actual number of cycles is 14 if the quotient is greater than 8 bits (overflow condition).

TMS370 Instruction Set Summary (continued)

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED					OTHER	
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)		
DSB	B, __	1/10										Decimal Subtract with Borrow
	RS, __	2/9	2/9	3/11								(d) – (s) – 1 + (C) → (d) (BCD)
	#iop8, __	2/8	2/8	3/1 0								
EINT											2/6	Enable Interrupts; 0Ch → (ST)
EINTH											2/6	EINT High Priority; 04h → (ST)
EINTL											2/6	EINT Low Priority; 08h → (ST)
IDLE											1/6	Idle Until Interrupt, Low Power entry
INC	__	1/8	1/8	2/6								Increment; (d) + 1 → (d)
INCW	#off8, __			3/11								Increment Word (Rp) + offset → (Rp)
INV	__	1/8	1/8	2/6								Invert; .NOT. (d) → (d)
JMP	__										2/7	Jump; PCN + offset8 → (PC)
JMPL	__					3/9	2/8	3/11	4/16			Jump; PCN + (D) → (PC)
Jcnd†												Jump Conditional
JN											2/5	Negative
JZ											2/5	Zero
JC											2/5	Carry
JP											2/5	Positive
JPZ											2/5	Positive or Zero
JNZ											2/5	Negative or Zero
JNC											2/5	No Carry
JV											2/5	Overflow, signed
JNV											2/5	No Overflow, signed
JGE											2/5	Greater Than or Equal, signed
JL											2/5	Less Than, signed
JG											2/5	Greater Than, Signed
JLE											2/5	Less Than or Equal, signed
JLO											2/5	Lower Value
JHS											2/5	Higher or Same
JBIT0†	__			4/1 0	4/11							Jump If Bit = 0
JBIT1†	__			4/1 0	4/11							Jump If Bit = 1
LDSP											1/7	Load Stack Pointer; (B) → (SP)
LDST	#iop8										2/6	Load ST Register; (s) → (SP)

MOV	A, __		1/9	2/7	2/8	3/10	2/9	3/12	4/17	2/7		Move; (s) → (d)
	__, A		1/8	2/7	2/8	3/10	2/9	3/12	4/17	2/7		
	B, __	1/8		2/7	2/8							
	Rs, __	2/7	2/7	3/9	3/1							
	Ps, __	2/8	2/8	3/1	0							
	#lop8, __	2/6	2/6	0								
				3/8	3/1							
					0							
MOVW	Rps, __			3/1								Move Word; 16-bit operands
	#lop16, __			2								(s) → (d)
	#lop16(B), __			4/1								
	#off8(Rp), __			3								
				4/1								
				5								
				5/2								
				0								

† Add 2 to cycle count if jump is taken.

TMS370 Instruction Set Summary (concluded)

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED					OTHER	
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)		
MPY	B, —	1/47										Multiply
	RS, —	2/46	2/4	3/4								(s) × (d) → (A:B)
	#lop8, —	2/45	6	8								A = MSB, B = LSB
			2/4	3/4								
			5	7								
NOP											1/7	NOP; (PC) +1 → (PC)
OR	A, —				2/9							OR
	B, —	1/8			2/9							(s) .OR. (d) → (d)
	Rs, —	2/7	2/7	3/9								
	#lop8, —	2/6	2/6	3/8	3/1							
					0							
POP	—	1/9	1/9	2/7							1/8	Pop Top of Stack
												((SP)) → (d); (SP) – 1 → (SP)
PUSH	—	1/9	1/9	2/7							1/8	Push onto Stack
												(SP) + 1 → (SP); (s) → ((SP))
RL	—	1/8	1/8	2/6								Rotate Left
RLC	—	1/8	1/8	2/6								Rotate Left Through Carry
RR	—	1/8	1/8	2/6								Rotate Right
RRC	—	1/8	1/8	2/6								Rotate Right Through Carry
RTI											1/12	Return from Interrupt
												Pop PC, Pop ST
RTS											1/9	Return from Subroutine, Pop PC
SBIT0	—			3/8	3/1							Set Bit to 0
					0							
SBIT1	—			3/8	3/1							Set BIT to 1
					0							
SETC											1/7	Set Carry; A0h → (ST)
SSB	B, —	1/8										Subtract with Borrow
	Rs, —	2/7	2/7	3/9								(d) – (s) – 1 + (C) → (d)
	#lop8, —	2/6	2/6	3/8								
STSP											1/8	Store Stack Pointer; (SP) → (B)
SUB	B, —	1/8										Subtract
	Rs, —	2/7	2/7	3/9								(d) – (s) → (d)
	#lop8, —	2/6	2/6	3/8								
SWAP	—	1/11	1/11	2/9								Swap Nibbles
												s(7- 4,3-0) → d(3-0,7- 4)
TRAPn											1/14	Trap to Subroutine; Push PCN;
												Vector n → (PC)
TST	—	1/9	1/1									Test; Set flags from register
			0									
XCHB	—	1/10	1/1	2/8								Exchange B; (B) ↔ (d)
			0									
XOR	A, —				2/9							Exclusive OR
	B, —	1/8			2/9							(s) .XOR. (d) → (d)
	Rs, —	2/7	2/7	3/9								
	#lop8, —	2/6	2/6	3/8	3/1							
					0							

TMS370 Family OPCODE/Instruction Map

SECOND NIBBLE	FIRST NIBBLE 0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	JMP ra 2/7							INCW #n,Rd 3/11	MOV Ps,A 2/8			CLRC TST A 1/9	MOV A,B 1/9	MOV A,Rd 2/7	TRAP 15 1/14	LDST n 2/6
1	JN ra 2/5		MOV A,Pd 2/8			MOV B,Pd 2/8		MOV Rs,Pd 3/10		MOV Ps,Rd 2/7				MOV B,Rd 2/7	TRAP 14 1/14	MOV n(SP),A 2/7
2	JZ ra 2/5	MOV Rs,A 2/7	MOV #n,A 2/6	MOV Rs,B 2/7	MOV Rs,Rd 3/9	MOV #n,B 2/6	MOV B,A 1/8	MOV #n,Rd 3/8			MOV Ps,Rd 3/10	DEC A 1/8	DEC B 1/8	DEC Rn 2/6	TRAP 13 1/14	MOV A,n(SP) 2/7
3	JC ra 2/5	AND Rs,A 2/7	AND #n,A 2/7	AND Rs,B 3/9	AND Rs,Rd 3/9	AND #n,B 2/6	AND B,A 1/8	AND #n,Rd 3/8	AND A,Pd 2/9	AND B,Pd 2/9	AND #n,Pd 3/10	INC A 1/8	INC B 1/8	INC Rn 2/6	TRAP 12 1/14	CMP n(SP),A 2/8
4	JP ra 2/5	OR Rs,A 2/7	OR #n,A 2/6	OR Rs,B 2/7	OR Rs,Rd 3/9	OR #n,B 2/6	OR B,A 1/8	OR #n,Rd 3/8	OR A,Pd 2/9	OR B,Pd 2/9	OR #n,Pd 3/10	INV A 1/8	INV B 1/8	INV Rn 2/6	TRAP 11 1/14	extend Inst,2 opcodes
5	JPZ ra 2/5	XOR Rs,A 2/7	XOR #n,A 2/6	XOR Rs,B 2/7	XOR Rs,Rd 3/9	XOR #n,B 2/6	XOR B,A 1/8	XOR #n,Rd 3/8	XOR A,Pd 2/9	XOR B,Pd 2/9	XOR #n,Pd 3/10	CLR A 1/8	CLR B 1/8	CLR Rn 2/6	TRAP 10 1/14	
6	JNZ ra 2/5	BTJO Rs,A 3/9	BTJO #n,A 3/8	BTJO Rs,B 3/9	BTJO Rs,Rd 4/11	BTJO #n,B 3/8	BTJO B,A 2/10	BTJO #n,Rd 4/10	BTJO A,Pd 3/11	BTJO B,Pd 3/10	BTJO #n,Pd 4/11	XCHB A 1/10	XCHB TESTB 1/10	XCHB Rn 2/8	TRAP 9 1/14	IDLE 1/6
7	JNC ra 2/5	BTJZ Rs,A 3/9	BTJZ #n,A 3/8	BTJZ Rs,B 3/9	BTJZ Rs,Rd 4/11	BTJZ #n,B 3/8	BTJZ B,A 2/10	BTJZ #n,Rd 4/10	BTJZ A,Pd 3/10	BTJZ B,Pd 3/10	BTJZ #n,Pd 4/11	SWAP A 1/11	SWAP B 1/11	SWAP Rn 2/9	TRAP 8 1/14	MOV #n,Pd 3/10
8	JV ra 2/5	ADD Rs,A 2/7	ADD #n,A 2/6	ADD Rs,B 2/7	ADD Rs,Rd 3/9	ADD #n,B 2/6	ADD B,A 1/8	ADD #n,Rd 3/8	MOVW #16,Rd 4/13	MOVW Rs,Rd 3/12	MOVW #16(B),Rd 4/15	PUSH A 1/9	PUSH B 1/9	PUSH Rs 2/7	TRAP 7 1/14	SETC 1/7
9	JL ra 2/5	ADC Rs,A 2/7	ADC #n,A 2/6	ADC Rs,B 2/7	ADC Rs,Rd 3/9	ADC #n,B 2/6	ADC B,A 1/8	ADC #n,Rd 3/8	JMPL lab 3/9	JMPL @Rd 2/8	JMPL lab(B) 3/11	POP A 1/9	POP B 1/9	POP Rd 2/7	TRAP 6 1/14	RTS 1/9
A	JLE ra 2/5	SUB Rs,A 2/6	SUB #n,A 2/6	SUB Rs,B 2/7	SUB Rs,Rd 3/9	SUB #n,B 2/6	SUB B,A 1/8	SUB B,A 1/8	MOV lab,A 3/10	MOV @Rs,A 2/9	MOV lab(B),A 3/12	DJNZ A,ra 2/10	DJNZ B,ra 2/10	DJNZ Rn,ra 3/8	TRAP 5 1/14	RTI 1/12
B	JHS ra 2/5	SBB Rs,A 2/7	SBB #n,A 2/6	SBB Rs,B 2/7	SUB Rs,Rd 3/9	SBB #n,B 2/6	SBB B,A 1/8	SBB #n,Rd 3/8	MOV A,lab 3/10	MOV A,@Rd 2/9	MOV A,lab(B) 3/12	COMPL A 1/8	COMPL B 1/8	COMPL Rn 2/6	TRAP 4 1/14	PUSH ST 1/8
C	JNV ra 2/5	MPY Rs, A 2/46	MPY #n,A 2/45	MPY Rs,B 2/46	MPY Rs,Rd 2/46	MPY #n,B 2/48	MPY B,A 1/47	MPY #n,Rd 3/47	BR lab 3/9	BR @Rd 2/8	BR lab(B) 3/11	RR A 1/8	RR B 1/8	RR Rn 2/6	TRAP 3 1/14	POP ST 1/8
D	JGE ra 2/5	CMP Rs,A 2/7	CMP #n,A 2/6	CMP Rs,B 2/7	CMP Rs,Rd 3/9	CMP #n,B 2/6	CMP B,A 1/8	CMP #n,Rd 3/8	CMP lab,A 3/11	CMP @Rs,A 2/10	CMP lab(B),A 3/13	RRC A 1/8	RRC B 1/8	RRC Rn 2/6	TRAP 2 1/14	LDSP 1/7
E	JG ra 2/5	DAC Rs,A 2/9	DAC #n,A 2/8	DAC Rs,B 2/9	DAC Rs,Rd 3/11	DAC #n,B 2/8	DAC B,A 1/10	DAC #n,Rd 3/10	CALL lab 3/13	CALL @Rd 2/12	CALL lab(B) 3/15	RL A 1/8	RL B 1/8	RL Rn 2/6	TRAP 1 1/14	STSP 1/8
F	JLO ra 2/5	DSB Rs,A 2/9	DSB #n,A 2/8	DSB Rs,B 2/9	DSB Rs,Rd 3/11	DSB #n,B 2/8	DSB B,A 1/10	DSB #n,Rd 3/10	CALLR lab 3/15	CALLR @Rd 2/14	CALLR lab(B) 3/17	RLC A 1/8	RLC B 1/8	RLC Rn 2/6	TRAP 0 1/14	NOP 1/7
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

FIRST NIBBLE

Second byte of two-byte instructions (F4xx):

	E	F
8	MOVW n(Rn) 4/15	DIV Rn,A 3/14-63
9	JMPL n(Rn) 4/16	
A	MOV n(Rn),A 4/17	
B	MOV A,n(Rn) 4/16	
C	BR n(Rn) 4/16	
D	CMP n(Rn) 4/18	
E	CALL n(Rn) 4/20	
F	CALLR n(Pn) 4/22	
	E	F

ra — relative address
Rn — Register
Rs — Register containing source byte
Rd — Register containing destination byte
Ps — Peripheral register containing source byte
Pd — Peripheral register containing destination byte
Pn — Peripheral register
#n — Immediate 8-bit number
#16 — Immediate 16-bit number
lab — 16-bit label
@Rn — 16-bit address of contents of register pair

development system support

The TMS370 family development support tools include an Assembler, a C-Compiler, a Linker, an In-Circuit emulator (XDS – eXtended Development Support), and a microcontroller programmer.

- Assembler/Linker (Part No. TMDS3740810-02 for PC, Part No. TMDS3740210-08 for VAX™/ VMS™ Part No. TMDS3740510-09 for Sun-3™ or Sun-4™)
  - — Extensive macro capability.
  - — High-speed operation.
  - — Format conversion utilities available for popular formats.
- ANSI C-Compiler (Part No. TMDS3740815-02 for PC, Part No. TMDS3740215-08 for VAX™/ VMS™, Part No. TMDS3740515-09 for Sun-3™ or Sun-4™)
  - — Generates assembly code of the TMS370 that can be easily inspected.
  - — The compilation, assembly, and linking steps can all be performed with a single command.
  - — Enables the user to directly reference the TMS370's port registers by using a naming convention.
  - — Provides flexibility in specifying the storage for data objects.
  - — C functions and assembly functions can be easily interfaced.
  - — Includes assembler and linker.
- XDX/11 (eXtended Development Support) In-Circuit Emulator
  - — Base (Part No. TMDS3761100 - For PC, requires cable)
    - – Cable for 44-pin PLCC, 40-pin DIP, or shrink DIP (Part No. TMDS3788844)
  - — PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
  - — Symbolic debugging.
  - — Execute single/multiple instructions, single/multiple statements, until/while condition, or at full speed until breakpoint.
  - — The user needs to provide a regulated 5-V power supply with a 3-A current capability.
- XDS/22 (eXtended Development Support) In-Circuit Emulator
  - — Base (Part No. TMDS3762200 - For PC, requires cable)
    - – Cable for 44-pin PLCC, 40-pin DIP, or shrink DIP (Part No. TMDS3788844)
  - — Contains all of the features of the XDS/11 described above but does not require an external power supply.
  - — Contains sophisticated breakpoint trace and timing hardware that provides up to 2047 qualified trace samples with symbolic disassembly.
  - — Allows breakpoints to be qualified by address and/or data on any type of memory acquisition. Up to four levels of events can be combined to cause a breakpoint.
  - — Provides timers for analyzing total and average time in routines.
  - — Contains an eight line logic probe for adding visibility of external signals to the breakpoint qualifier and to the trace display.

VMS and **VAX** are trademarks of Digital Equipment Corporation.
Sun-3 and **Sun-4** are trademarks of Sun Microsystems, Inc.

- Microcontroller Programmer
  - — Base (Part No. TMDS3760500 – For PC, requires programming head)
    - – Single unit head for 44-pin PLCC (Part No. TMDS3780510)
    - – Single unit head for 40-pin DIP or shrink DIP (Part No. TMDS3780511)
    - – Gang programmer head supports 16 40-pin DIP parts (Part No. TMDS3780525)
    - – Gang programmer head supports 16 40-pin shrink DIP parts (Part No. TMDS3780526)
    - – Gang programmer head supports 16 44-pin PLCC parts (Part No. TMDS3780524)
  - — PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
- Design Kit (Part No. TMDS3770110 – For PC)
  - — Includes TMS370 Application Board and TMS370 Assembler diskette and documentation.
  - — Supports quick evaluation of TMS370 functionality.
  - — Capability to upload and download code.
  - — Capability to execute programs and software routines, and to single-step executable instructions.
  - — Software breakpoints to halt program execution at selected addresses.
  - — Wire-wrap prototype area.
  - — Reverse assembler.
  - — Requires adapter for programming 'Cx4x devices.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V_{CC}, V_{CC3} (see Note 5) ..... −0.6 V to 7 V
Input voltage range, All pins except MC ..... −0.6 V to 7 V
MC ..... −0.6 V to 14 V
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{CC}$) ..... ±20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{CC}$) ..... ±20 mA
Continuous output current per buffer, I_O (V_O = 0 to V_{CC})‡ ..... ±10 mA
Maximum I_{CC} current ..... 170 mA
Maximum I_{SS} current ..... −170 mA
Continuous power dissipation ..... 1 W
Storage temperature range ..... −65°C to 150 °C

† Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
‡ Electrical characteristics are specified with all output buffers loaded with the specified I_O current. Exceeding the specified I_O current in any buffer may affect the levels on other buffers.

recommended operating conditions (see Note 5)

PARAMETER			MIN	NOM	MAX	UNIT
V_{CC}	Digital logic supply voltage (see Note 5)		4.5	5	5.5	V
V_{CC}	RAM data retention supply voltage (see Note 6)		3		5.5	V
V_{CC3}	Analog supply voltage (see Note 5)		4.5	5	5.5	V
V_{SS3}	Analog supply ground		−0.3	0	0.3	V
V_{IL}	Low-level input voltage	All pins except MC	V_{SS}		0.8	V
		MC	V_{SS}		0.3	
V_{IH}	High-level input voltage	All pins except MC, XTAL2/CLKIN, and $\overline{RESET}$	2		V_{CC}	V
		MC (non-WPO mode)	V_{CC} − 0.3		V_{CC}+0.3	
		XTAL2/CLKIN	0.8 V_{CC}		V_{CC}	
		$\overline{RESET}$	0.7 V_{CC}		V_{CC}	
V_{MC}	MC (mode control) voltage	EEPROM write protect override	11.7	12	13	V
		Microcomputer	V_{SS}		0.3	
		EPROM programming voltage (V_{PP})	12	12.5	13	
T_A	Operating free-air temperature	A version	−40		85	°C
		L version	0		70	°C

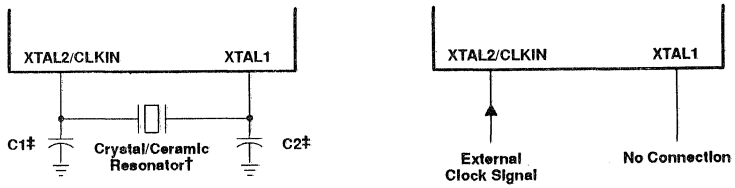
NOTES: 5. All voltage values are with respect to V_{SS}.
6. $\overline{RESET}$ is externally released while V_{CC} is within the recommended operating range of 4.5 V to 5.5 V and is externally activated when $V_{CC} < 4.5$ or $V_{CC} > 5.5$ V. RAM data retention is valid throughout the 2 MHz-20 MHz frequency range. An active $\overline{RESET}$ initializes (clears) RAM locations 0000h and 0001h.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER			TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OL}	Low-level digital output voltage		I_{OL} = 1.4 mA			0.4	V
V_{OH}	High-level output voltage		I_{OH} = −50 µA	0.9 V_{CC}			V
			I_{OH} = −2 mA	2.4			
I_I	Input current	MC	0.3 V ≤ V_I ≤ V_{CC} − 0.3 V			50	µA
			0 V ≤ V_I ≤ 0.3 V			10	
			V_{CC} − 0.3 ≤ V_I ≤ V_{CC} + 0.3 V			10	
			V_{CC} + 0.3 V ≤ V_I ≤ 13			650	µA
		I/O pins	0 V ≤ V_I ≤ V_{CC}			±10	µA
I_{OL}	Low-level output current		V_{OL} = 0.4 V	1.4			mA
I_{OH}	High-level output current		V_{OH} = 0.9 V_{CC}	−50			µA
			V_{OH} = 2.4 V	−2			mA
I_{CC}	Supply current (Operating mode) Osc Power bit = 0 (see Note 8)		CLKIN = 20 MHz (see Note 7 and 8)			45	mA
			CLKIN = 12 MHz (see Note 7)			30	
			CLKIN = 2 MHz (see Note 7 and 8)			11	
I_{CC}	Supply current (Standby mode) Osc Power bit = 0 (see Note 9)		CLKIN = 20 MHz (see Note 7 and 8)			17	mA
			CLKIN = 12 MHz (see Note 7 and 8)			11	
			CLKIN = 2 MHz (see Note 7 and 8)			3.5	
I_{CC}	Supply current (Standby mode) Osc Power bit = 1 (see Note 10)		CLKIN = 12 MHz (see Note 7 and 8)			8.6	mA
			CLKIN = 2 MHz (see Note 7 and 8)			3.0	
I_{CC}	Supply current (Half Mode)		CLKIN < 0.2 V (see Note 7 and 8)			30	µA

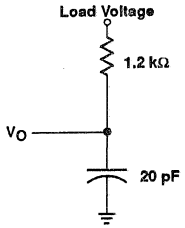
NOTES: 7. Microcontroller–Single chip mode, ports configured as inputs, or outputs with no load. All inputs ≤ 0.2 V or ≥ V_{CC} −0.2 V.
8. XTAL2/CLKIN is driven with an external square wave signal with 50% duty cycle and rise and fall times less than 10 ns. Current may be higher with a crystal oscillator. At 20 MHz this extra current = .01 mA × (total load capacitance + crystal capacitance in pF).
9. Maximum operating current for = 1.90 (Fx) + 7 mA.
10. Maximum standby current for = .75 (Fx) + 2 mA. (OSC power bit = 0)
11. Maximum standby current for = 0.56 (Fx) + 1.88 mA. (Osc power bit = 1)
(Valid only from 2 MHz to 12 MHz.)

RECOMMENDED CRYSTAL/CLOCK CONNECTIONS



† The crystal/ceramic resonator frequency is four times the reciprocal of the system clock period.
‡ The values of C1 and C2 should be the values recommended by the crystal/ceramic resonator manufacturer.

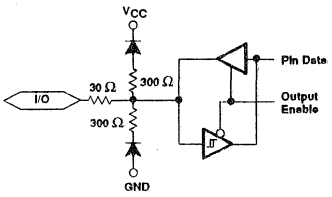
TYPICAL OUTPUT LOAD CIRCUIT§

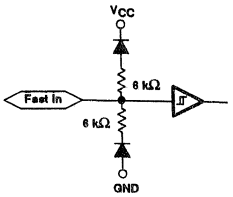


Case 1: $V_O = V_{OH} = 2.4$ V; Load Voltage = 0 V
Case 2: $V_O = V_{OL} = 0.4$ V; Load Voltage = 2.8 V for Ports A, B and D, and $\overline{\text{RESET}}$
Load Voltage = 2.1 for other Outputs

§ All measurements are made with the pin loading as shown unless otherwise noted. All measurements are made with XTAL2/CLKIN driven by an external square wave signal with a 50% duty cycle and rise and fall times less than 10 ns unless otherwise stated.

TYPICAL INPUT BUFFERING





PARAMETER MEASUREMENT INFORMATION

timing parameter symbology

Timing parameter symbols have been created in accordance with JEDEC Standard 100. In order to shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

A	Address	RXD	SCIRXD
AR	Array	S	Slave mode
B	Byte	SCC	SCICLK
CI	XTAL2/CLKIN	SIMO	SPISIMO
CO	CLKOUT	SOMI	SPISOMI
D	Data	TXD	SCITXD
PGM	Program	W	Write
R	Read		

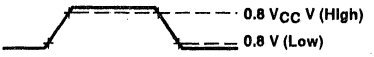
Lowercase subscripts and their meanings are:

c	cycle time (period)	r	rise time
d	delay time	su	setup time
f	fall time	v	valid time
h	hold time	w	pulse duration (width)

The following additional letters are used with these meanings:

H	High	V	Valid
L	Low	Z	High Impedance

All timings are measured between high and low measurement points as indicated in the figures below.



XTAL2/CLKIN Measurement Points

General Measurement Points

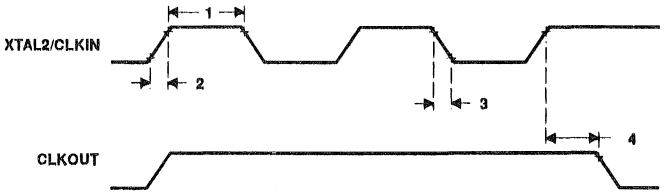
external clocking requirements†

NO.	PARAMETER		MIN	NOM	MAX	UNIT
1	$t_{w(CI)}$	XTAL2/CLKIN pulse duration (see Note 12)	20			ns
2	$t_{r(CI)}$	XTAL2/CLKIN rise time			30	ns
3	$t_{f(CL)}$	XTAL2/CLKIN fall time			30	ns
4	$t_{d(CIH-COL)}$	Delay time, XTAL2/CLKIN rise to CLKOUT fall			100	ns
	CLKIN	Crystal operating frequency	2		20	MHz

† For V_{IL} and V_{IH}, refer to "Recommended Operating Conditions".

NOTE 12: This pulse may be either a high pulse, as illustrated below, which extends from the earliest valid high to the final valid high in an XTAL2/CLKIN cycle, or a low pulse, which extends from the earliest valid low to the final valid low in an XTAL2/CLKIN cycle.

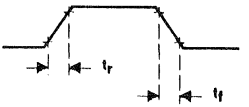
external clock timing



EXPANSION MODE OUTPUT

general purpose output signal switching time requirements

		MIN	NOM	MAX	UNIT
t_r	Rise time		30		ns
t_f	Fall time		30		ns

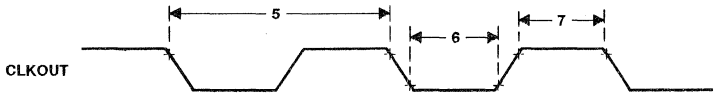


switching characteristics and timing requirements (see Note 13)

NO.	PARAMETER		MIN	MAX	UNIT
5	t_c	CLKOUT (system clock) cycle time	200	2000	ns
6	$t_{w(COL)}$	CLKOUT low pulse duration	$0.5t_c$-20	$0.5t_c$	ns
7	$t_{w(COH)}$	CLKOUT high pulse duration	$0.5t_c$	$0.5t_c$ + 20	ns

NOTE 13: t_c = system clock cycle time = $4/f_x$.

CLKOUT timing



recommended EEPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(PGM)B}$	Programming signal pulse duration to insure valid data is stored (byte mode)	10			ms
$t_{w(PGM)AR}$	Programming signal pulse duration to insure valid data is stored (any array mode)	20			ms

recommended EPROM operating conditions for programming

PARAMETER		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.75	5.5	6.0	V
V_{PP}	Supply voltage at MC pin	12	12.5	13	V
I_{PP}	Supply current at MC pin during programming (V_{PP} = 13 V)		35	50	mA
CLKIN	Operating crystal frequency	2		20	MHz

recommended EPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(IEPGM)}$	Initial programming signal pulse (see Note 14)	.95	1	1.05	ms
tw(FEPGM)	Final programming signal pulse	2.85		78.75	ms

NOTE 14: Programming pulse is active when both EXE (EPCTL.0) and V_{PPS} (EPCTL.6) are set.

SERIAL COMMUNICATIONS INTERFACE (SCI) INTERNAL CLOCK ISOSYNCHRONOUS MODE I/O TIMING

SCI Isosynchronous mode timing characteristics for internal clock (see Note 13)

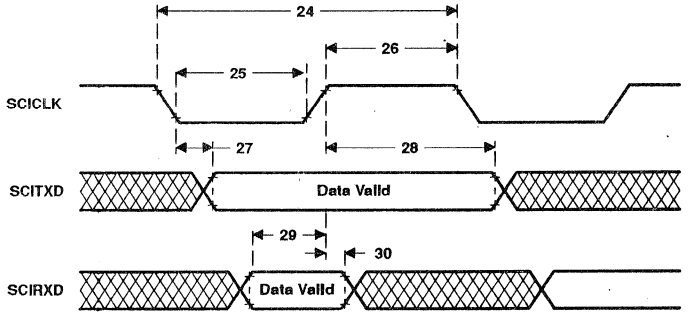
NO.	PARAMETER		MIN	MAX	UNIT
24	$t_{c(SCC)}$	SCICLK cycle time	$2t_c$	$131{,}072t_c$	ns
25	$t_{w(SCCL)}$	SCICLK low pulse duration	$t_c - 45$	$0.5t_{c(SCC)}+45$	ns
26	$t_{w(SCCH)}$	SCICLK high pulse duration	$t_c - 45$	$0.5t_{c(SCC)}+45$	ns
27	$t_{d(SCCL-TXDV)}$	Delay time, SCITXD valid after SCICLK low	−50	50	ns
28	$t_{v(SCCH-TXD)}$	SCITXD data valid after SCICLK high	$t_{w(SCCH)} - 50$		ns

SCI Isosynchronous mode timing requirements for internal clock (see Note13)

NO.			MIN	MAX	UNIT
29	$t_{su(RXD-SCCH)}$	SCIRXD setup time to SCICLK high	$0.25t_c + 145$		ns
30	$t_{v(SCCH-RXD)}$	SCIRXD data valid after SCICLK high	0		ns

NOTE 13: t_c = system clock cycle time = $4/f_X$.

SCI Isosynchronous mode timing diagram for internal clock



SERIAL COMMUNICATIONS INTERFACE (SCI) EXTERNAL CLOCK ISOSYNCHRONOUS MODE I/O TIMING

SCI Isosynchronous mode timing characteristics for external clock (see Note 13)

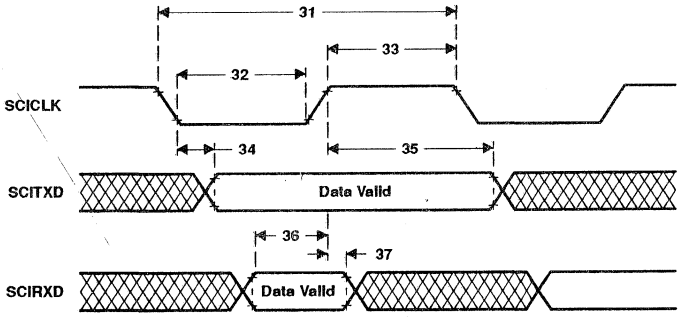
NO.	PARAMETER		MIN	MAX	UNIT
34	$t_{d(SCCL-TXDV)}$	Delay time, SCITXD valid after SCICLK low		$4.25t_c + 145$	ns
35	$t_{v(SCCH-TXD)}$	SXITXD data valid after SCICLK high	$t_{w(SCCH)}$		ns

SCI Isosynchronous mode timing requirements for external clock (see Note 13)

NO.			MIN	MAX	UNIT
31	$t_{c(SCC)}$	SCICLK cycle time	$10t_c$		ns
32	$t_{w(SCCL)}$	SCICLK low pulse duration	$4.25t_c + 120$		ns
33	$t_{w(SCCH)}$	SCICLK high pulse duration	$t_c + 120$		ns
36	$t_{su(RXD-SCCH)}$	SCIRXD setup time to SCICLK high	40		ns
37	$t_{v(SCCH-RXD)}$	SCIRXD data valid after SCICLK high	$2t_c$		ns

NOTE 13: t_c = system clock cycle time = $4/f_x$.

SCI Isosynchronous mode timing diagram for external clock



A/D converter

The A/D Converter has a separate power bus for its analog circuitry. These pins are referred to as V_{CC3} and V_{SS3}. The purpose is to enhance A/D performance by preventing digital switching noise on the logic circuitry which could be present on V_{SS} and V_{CC} from coupling into the A/D analog stage. All A/D specifications will be given with respect to V_{SS3} unless otherwise noted.

Resolution ..... 8 bits (256 values)
Monotonic ..... Yes
Output conversion code ..... 00h to FFh (00h for $V_I \leq V_{SS3}$; FFh for $V_I \geq V_{ref}$)
Conversion time (excluding sample time) ..... $164t_c$

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC3}	Analog supply voltage	4.5	5	5.5	V
		$V_{CC} - 0.3$		$V_{CC} + 0.3$	
V_{SS3}	Analog ground	$V_{SS} - 0.3$		$V_{SS} + 0.3$	V
V_{ref}	Non-V_{CC3} reference (see Note 15)	2.5	V_{CC3}	$V_{CC3} + 0.1$	V
	Analog input for conversion	V_{SS3}		V_{ref}	V

NOTE 15: V_{ref} must be stable, within ± 1/2 LSB of the required resolution, during the entire conversion time.

operating characteristics over full ranges of recommended operating conditions

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
	Absolute accuracy (see Note 16)	$V_{CC3} = 5.5$ V, $V_{ref} = 5.1$ V			±1.5	LSB
	Differential/integral linearity error (see Notes 16 and 17)	$V_{CC3} = 5.5$ V, $V_{ref} = 5.1$ V			±0.9	LSB
I_{CC3}	Analog supply current	Converting			2	mA
		Not Converting			5	µA
I_I	Input current, AN0-AN7	$0 \text{ V} \leq V_I \leq 5.5$ V			2	µA
	V_{ref} input charge current				1	mA
Z_{ref}	Source impedance V_{ref}	XTAL2/CLKIN ≤ 12 MHz			24	kΩ
		12 MHz < XTAL2/CLKIN ≤ 20 MHz			10	kΩ

NOTES: 16. Absolute resolution = 20 mV. At $V_{ref} = 5.1$ V, this is 1 LSB. As V_{ref} decreases, LSB size decreases and thus absolute accuracy and differential / integral linearity errors in terms of LSBs increases.
17. Excluding quantization error of 1/2 LSB.

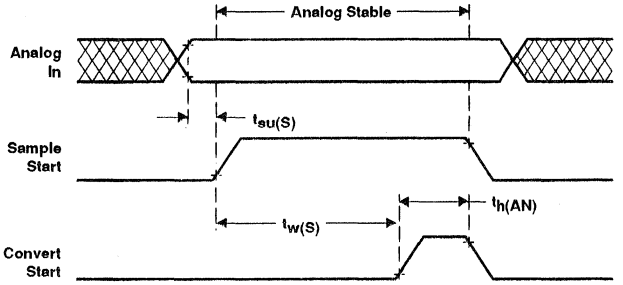
The A/D module allows complete freedom in design of the sources for the analog inputs. The period of the sample time is user-defined such that high impedance sources can be accommodated without penalty to low-impedance sources. The sample period begins when the SAMPLE START bit of the A/D Control Register (ADCTL) is set to 1. The end of the signal sample period occurs when the conversion bit (CONVERT START) of the ADCTL is set to 1. After a hold time, the converter will reset the SAMPLE START and CONVERT START bits, signaling that a conversion has started and the analog signal can be removed.

analog timing requirements

		MIN	NOM	MAX	UNIT
$t_{su(S)}$	Analog input setup to sample command	0			ns
$t_{h(AN)}$	Analog input hold from start of conversion	$18t_c$			ns
$t_{w(S)}$	Duration of sample time per kilohm of source impedance (see Note 18)	1			µs/kΩ

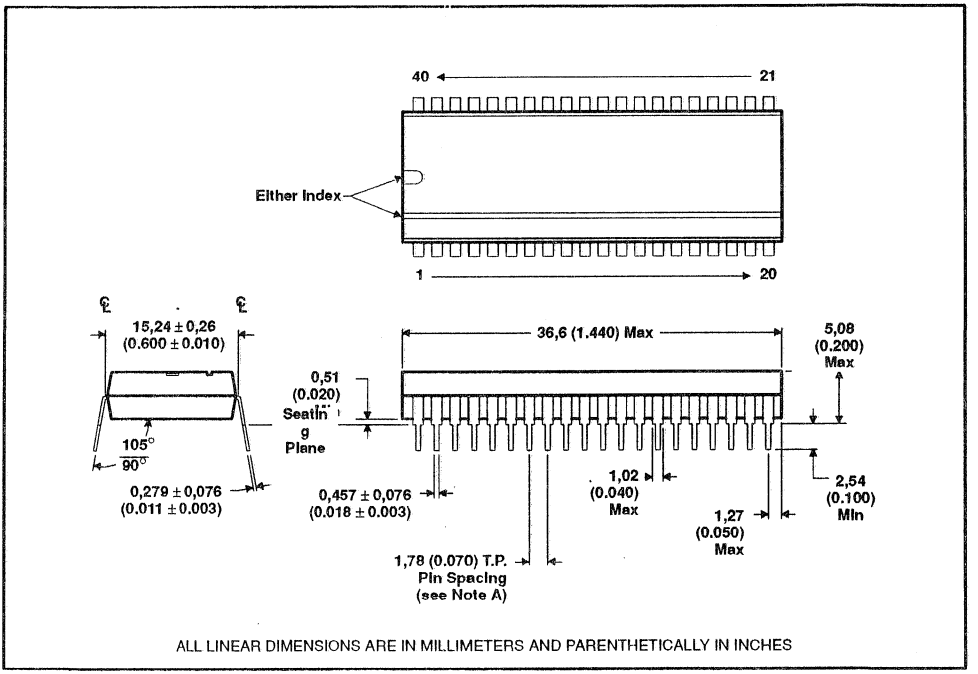
NOTE 18: The value given is valid for a signal with a source impedance greater than 1 kΩ. If the source impedance is less than 1 kΩ, use a minimum sampling time of 1 µs.

analog timing



MECHANICAL DATA

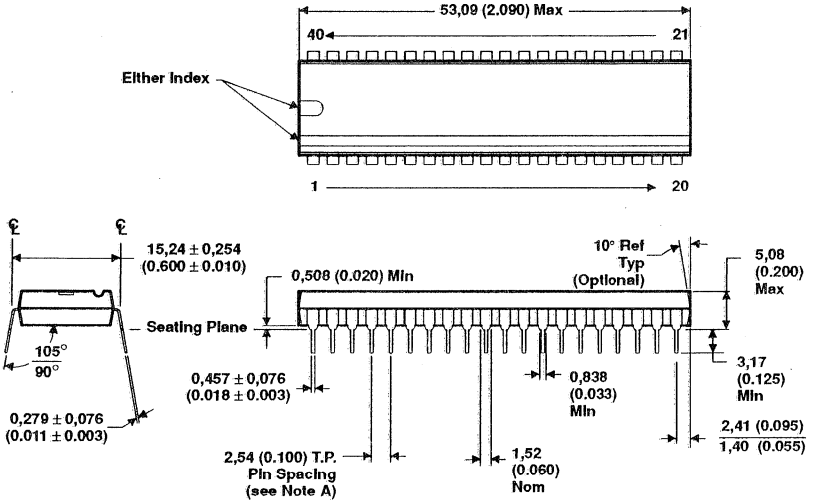
40-pin leaded chip carrier package (N2 suffix)



ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTE: A. Each pin centerline is located within 0,26 (0.010) of its true longitudinal position.

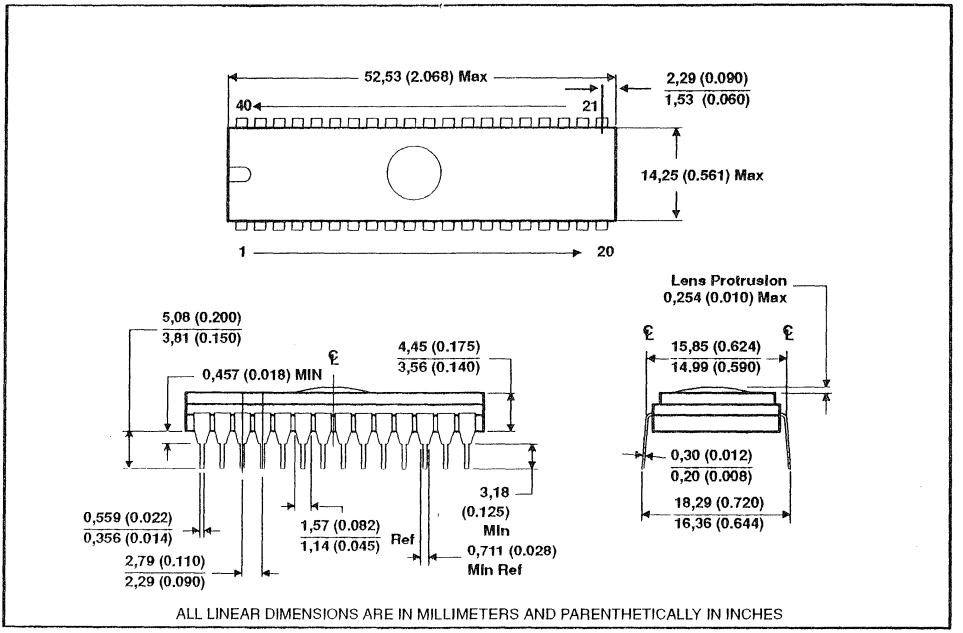
40-pin plastic leaded chip carrier package (N suffix)



ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTE: A. Each pin centerline is located within 0,26 (0.010) of its true longitudinal position.

40-pin ceramic dual-in-line package (J suffix)

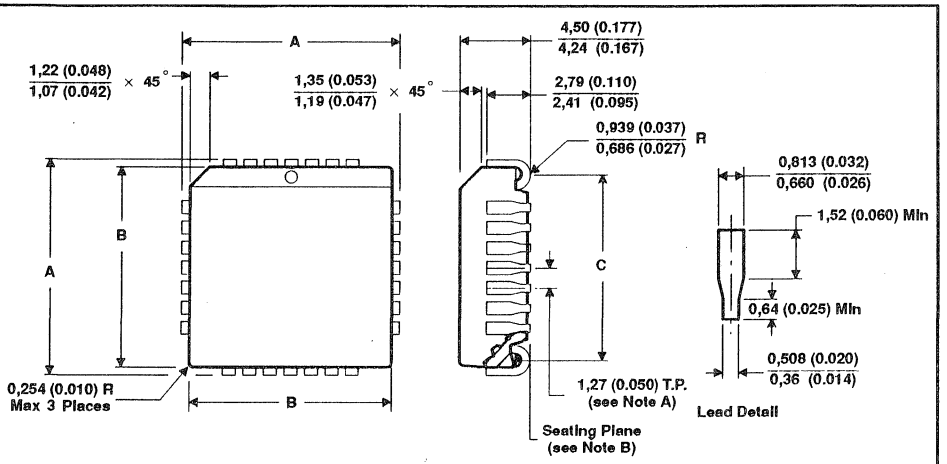


ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTE: A. Each pin centerline is located within 0,26 (0.010) of its true longitudinal position.

MECHANICAL DATA

44-pin plastic leaded chip carrier package (FN suffix)



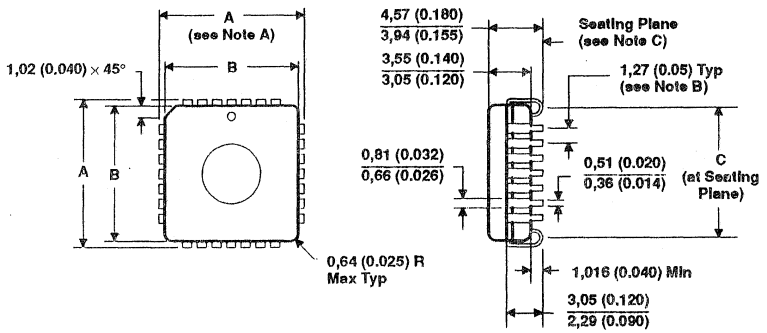
JEDEC OUTLINE	NO. OF TERMINALS	A		B		C	
		MIN	MAX	MIN	MAX	MIN	MAX
M0-047AB	28	12,32 (0.485)	12,57 (0.495)	10,92 (0.430)	11,56 (0.455)	10,41 (0.410)	10,92 (0.430)
M0-047AC	44	17,40 (0.685)	17,65 (0.695)	16,00 (0.630)	16,64 (0.655)	15,49 0.610)	16,00 (0.630)
M0-047AE	68	25,02 (0.985)	25,27 (0.995)	23,62 (0.930)	24,26 (0.955)	23,11 (0.910)	23,62 (0.930)

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTES: A. Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
B. The lead contact points are planar within 0,101(0.004).

MECHANICAL DATA

44-lead FZ cerquad chip carrier package



JEDEC OUTLINE	NO. OF TERMINALS	A		B		C	
		MIN	MAX	MIN	MAX	MIN	MAX
M0-087AA	28	12,32 (0.485)	12,57 (0.495)	10,92 (0.430)	11,56 (0.455)	10,41 (0.410)	10,92 (0.430)
M0-087AB	44	17,40 (0.685)	17,65 (0.695)	16,00 (0.630)	16,64 (0.655)	15,49 0.610)	16,00 (0.630)
M0-087AD	68	25,02 (0.985)	25,27 (0.995)	23,62 (0.930)	24,26 (0.955)	23,11 (0.910)	23,62 (0.930)

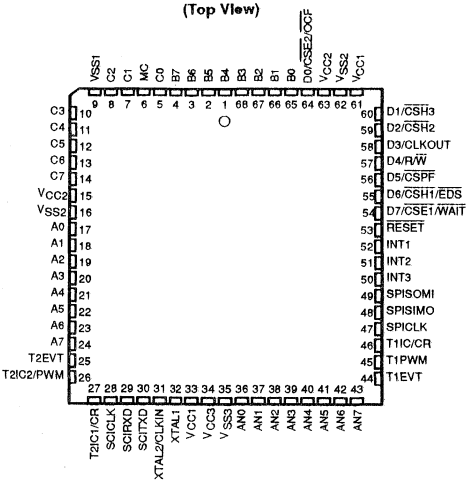
ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTES: A. Center line of center pin each side is within 0,10 (0.004) of package centerline as determined by dimension B.
B. Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
C. The lead contact points are planar with 0,15 (0.006).

SPNS010B — DECEMBER 1986 — REVISED JULY 1992

- CMOS/EEPROM/EPROM Technology on a Single Device
  - — Mask ROM Devices for High-Volume Production
  - — One Time Programmable (OTP) Devices for Low-Volume Production
  - — EPROM Devices for Prototyping Purposes
- Flexible Operation Features
  - — Power-Reduction STANDBY and HALT Modes
  - — Commercial and Industrial Temperature Ranges
  - — Input Clock Frequency 2 MHz to 20 MHz
  - — Voltage (V_{CC}): 5 V ± 10%
- System Memory Configurations
  - — On-Chip Program Memory Versions
    - – ROM: 4K, 8K, 16K and 32KBytes
    - – EPROM: 16K and 32KBytes
    - – ROMless
  - — Data EEPROM: 256 Bytes or 512 Bytes
  - — Static RAM: 256 Bytes, 512 Bytes or 1 KByte
  - — External Memory/Peripheral WAIT States
  - — Precoded External Chip Select Outputs in Microcomputer Mode
    - – Allows 112K-Bytes External Addressable Memory
    - – No Logic Needed for External Memory Addressing
- Eight-Channel 8-Bit A/D Converter
- Two 16-Bit General-Purpose Timers
  - — Software Configurable as Two 16-Bit Event Counters, or Two 16-Bit Pulse Accumulators, or Three 16-Bit Input Capture Functions, or Four Compare Registers, or Two Self-Contained PWM Functions
  - — Software Programmable Input Polarity
  - — One Timer Has an 8-Bit Prescaler, Providing a 24-Bit Realtime Timer
- On-Chip 24-Bit Watchdog Timer
- Serial Communications Interface (SCI)
  - — Asynchronous and Isosynchronous Modes
  - — Full Duplex, Double-Buffered Rx and Tx
  - — Two Multiprocessor Communications Formats
- Plastic and Ceramic 68-Pin Leaded Chip Carrier Packages

FN/FZ Package
(Top View)



- Serial Peripheral Interface (SPI)
  - — Variable-Length High-Speed Shift Register
  - — Synchronous Master/Slave Operation
- Flexible Interrupt Handling
  - — Two S/W Programmable Interrupt Levels
  - — Global and Individual Interrupt Masking
  - — Programmable Rising or Falling Edge Detect
- 55 CMOS/TTL Compatible I/O Pins
  - — All Peripheral Function Pins Software Configurable for Digital I/O
  - — 46 Bidirectional, 9 Input Pins
- TMS370 Series Compatibility
  - — Register-to-Register Architecture
  - — 256 General-Purpose Registers
  - — Fourteen Powerful Addressing Modes
- PC-Based Workstation Development Support Emphasizes Productivity, Featuring:
  - — C Compiler Support
  - — Realtime In-Circuit Emulation
  - — Symbolic Debug
  - — Extensive Breakpoint/Trace Capability
  - — Software Performance Analysis
  - — Multi-Window User Interface
  - — EEPROM/EPROM Programming

description

TMS370Cx5x family of single-chip 8-bit microcontrollers provides cost-effective realtime system control through VLSI integration of advanced peripheral function modules and various on-chip memory configurations. The TMS370Cx5x family presently consists of eleven devices, which can be grouped into four main sub-families: the TMS370Cx50, the TMS370Cx52, the TMS370Cx56, and the TMS370Cx58.

Unless otherwise noted, the following terms are used to refer to the individual devices described in this data sheet:

TMS370Cx50 refers to the TMS370C050, TMS370C150, TMS370C250, and TMS370C350 devices.

TMS370Cx52 refers to the TMS370C052 and TMS370C352 devices.

TMS370Cx56 refers to the TMS370C056, TMS370C156, TMS370C256, TMS370C356, and TMS370C756 devices.

TMS370Cx58 refers to the TMS370C058, TMS370C358, and TMS370C758 devices.

TMS370Cx5x refers to the TMS370Cx50, TMS370Cx52, TMS370Cx56, and TMS370Cx58 subfamilies.

The TMS370Cx5x family is implemented using high-performance silicon-gate CMOS technology. The low operating power, wide operating temperature range, and noise immunity of CMOS technology, coupled with the high performance and extensive on-chip peripheral functions, make the TMS370Cx5x devices attractive in system designs for automotive electronics, industrial motor control, computer peripheral control, and telecommunications.

The on-chip memory configurations of the TMS370Cx5x family include various amounts of static RAM, data EEPROM, and program memory (ROM or EPROM). TMS370Cx5x devices with mask ROM are presently available in sizes ranging form 4K-bytes to 32K-bytes. Devices with EPROMare available with 16K- or 32K-bytes. Devices are also available as ROMless microprocessors. The following table provides an overview of the various memory configurations and operating modes of the TMS370Cx5x devices.

DEVICE	PROGRAM MEMORY (BYTES)		DATA MEMORY (BYTES)		OPERATING MODES†		PACKAGE
	ROM	EPROM	RAM	EEPROM	µC	µP	
TMS370CX50: TMS370C050, TMS370C150, TMS370C250 AND TMS370C350							
TMS370C050	4K	–	256	256	√	√	FN - PLCC
TMS370C150	–	–	256	–	–	√	FN - PLCC
TMS370C250	–	–	256	256	–	√	FN - PLCC
TMS370C350	4K	–	256	–	√	√	FN - PLCC
TMS370Cx52: TMS370C052, AND TMS370C352							
TMS370C052	8K	–	256	256	√	√	FN - PLCC
TMS370C352	8K	–	256	–	√	√	FN - PLCC
TMS370Cx56: TMS370C056, TMS370C156, TMS370C256, TMS370C356 AND TMS370C756							
TMS370C056	16K	–	512	512	√	√	FN - PLCC
TMS370C156	–	–	512	–	–	√	FN - PLCC
TMS370C256	–	–	512	512	–	√	FN - PLCC
TMS370C356	16K	–	512	–	√	√	FN - PLCC
TMS370C756	–	16K	512	512	√	√	FN - PLCC / FZ - CLCC
TMS370Cx58 :TMS370C058,TMS370C358,TMS370C758							
TMS370C058	32K	–	1K	256	√	–	FN – PLCC
TMS370C358	32K	–	1K	–	√	–	FN – PLCC
TMS370C758	–	32K	1K	256	√	–	FN – PLCC / FZ - CLCC

† µC – Microcomputer mode; µP – Microprocessor Mode

All TMS370Cx5x devices contain a minimum of the following on-chip peripheral modules:

- 256 bytes RAM (usable as registers)
- 8-channel, 8-bit Analog-to-Digital converter (A/D)
- Serial Communications Interface (SCI)
- Serial Peripheral Interface (SPI)
- Two 24-bit general-purpose timers, one of which can be used as a Watchdog timer
- One 16-bit general-purpose timer

The TMS370Cx5x provides two power reduction modes (STANDBY and HALT) for applications where low power consumption is critical. Both modes stop all CPU activity (i.e., no instructions are executed). In the STANDBY mode the internal oscillator, the general purpose timer, and the SCI receiver start bit detection remain active. In the HALT mode, all device activity is stopped. The device retains all RAM data and peripheral configuration bits throughout both powerdown modes.

The TMS370Cx5x features advanced register-to-register architecture that allows direct arithmetic and logical operations without requiring an accumulator (e.g., ADD r24, r47; add the contents of register 24 to the contents of register 47 and store the result in register 47). The TMS370Cx5x family is fully instruction-set-compatible, allowing easy transition between members.

The TMS370Cx5x family offers an 8-channel Analog-to-Digital converter with 8-bit accuracy. The 33-µs conversion time at 20 MHz and the variable sample period, combined with selectable positive reference voltage sources, turn real-world analog signals into digital data.

The SPI and the two operational modes of the SCI give three methods of serial communications. The SCI allows standard RS-232-C communications between other common data transmission equipment, while the SPI gives high speed communications between simpler shift register type devices, such as display drivers, A/D converter, PLL, I/O expansion, or other microcontrollers in the system.

For large memory applications, the TMS370Cx5x family provides an external bus with non-multiplexed address and data. Precoded memory chip select outputs can be enabled, which allow minimum-chip-count system implementations. Wait-state support facilitates performance matching between the CPU and external memory and the peripherals. All pins associated with memory expansion interface are individually software configurable for general-purpose digital input/output pins when operating in the microcomputer mode.

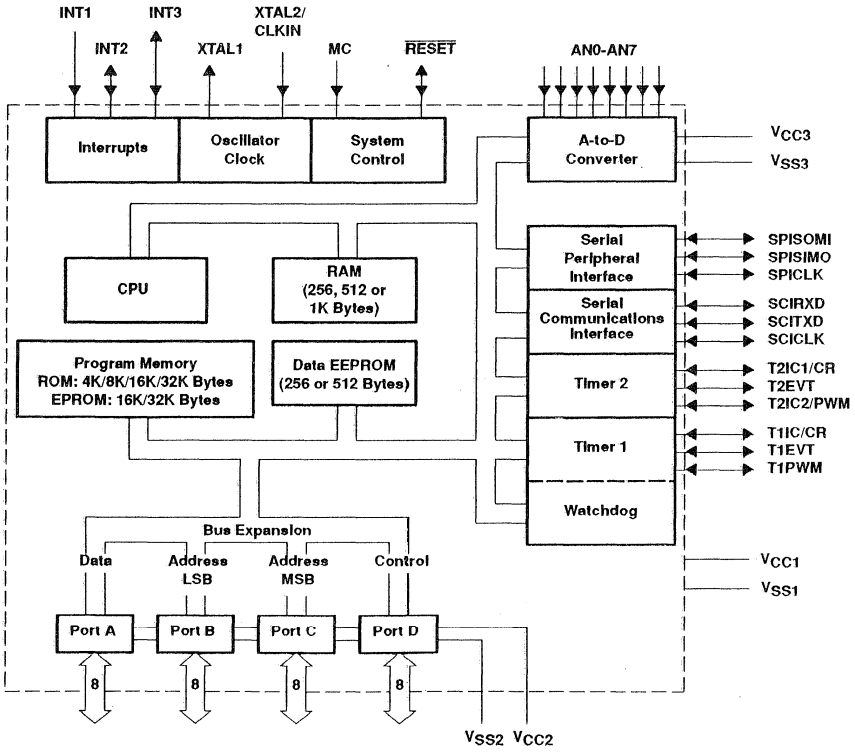
The TMS370C756FZ and TMS370C758FZ are available in a 68-pin, windowed ceramic package (FZ suffix) that allows memory reprogramming during the development design prototyping phase. This achieves quick updates to breadboards and prototype systems using socketed FFE devices while iterating initial designs.

The TMS370C756FN and TMS370C758FN are available in a 68-pin plastic package (FN suffix) and are one time programmable (OTP). This is an effective microcomputer to use for immediate production updates for other members of the TMS370Cx5x family or for low-volume production runs that cannot satisfy minimum volume or cycle time requirement for the lower cost mask ROM devices.

The TMS370Cx5x family provides a very economical, efficient solution to realtime control applications. The TMS370 family eXtended Development System (XDS) solves the challenge of efficiently developing the software and hardware required to design the TMS370Cx5x into an ever-increasing number of complex applications. The application source code can be written in assembly language or in C. The TMS370 family XDS communicates via standard RS-232-C interface with an existing personal computer to form a PC-DOS hosted workstation. This allows use of the PC's editors and software utilities already familiar to the designer. The TMS370 family XDS emphasizes ease-of-use through extensive use of menus and screen windowing so that the system designer can begin developing software with minimum training. Precise realtime in-circuit emulation and extensive symbolic debug and analysis tools ensure efficient software and hardware implementation as well as reducing time-to-market cycle.

The TMS370Cx5x family mask ROM and EPROM, together with the TMS370 family XDS for applications development, and comprehensive product documentation and customer support provide a complete solution to the needs of the system designer.

functional block diagram



TERMINAL FUNCTIONS

PIN			I/O	DESCRIPTION
NAME	ALTERNATE FUNCTION	NO.		
A0	DATA0 (LSB)	17	I/O	Single-Chip Mode: Port A is a general purpose bidirectional I/O port.
A1	DATA1	18	I/O	
A2	DATA2	19	I/O	Expansion Mode: Port A may be individually programmed as the external
A3	DATA3	20	I/O	bidirectional data bus (DATA0–DATA7).
A4	DATA4	21	I/O	
A5	DATA5	22	I/O	
A6	DATA6	23	I/O	
A7	DATA7 (MSB)	24	I/O	
B0	ADD0	65	I/O	Single-Chip Mode: Port B is a general purpose bidirectional I/O port.
B1	ADD1	66	I/O	
B2	ADD2	67	I/O	Expansion Mode: Port B may be individually programmed as the low order
B3	ADD3	68	I/O	address output bus (ADD0–ADD7).
B4	ADD4	1	I/O	
B5	ADD5	2	I/O	
B6	ADD6	3	I/O	
B7	ADD7	4	I/O	
C0	ADD8	5	I/O	Single-Chip Mode: Port C is a general purpose bidirectional I/O port.
C1	ADD9	7	I/O	
C2	ADD10	8	I/O	Expansion Mode: Port C may be individually programmed as the high order
C3	ADD11	10	I/O	address output bus (ADD8–ADD15).
C4	ADD12	11	I/O	
C5	ADD13	12	I/O	
C6	ADD14	13	I/O	
C7	ADD15	14	I/O	

TERMINAL FUNCTIONS (continued)

PIN NAME	PIN ALTERNATE FUNCTION		PIN NO.	I/O	DESCRIPTION
	FUNCTION				Single-Chip Mode: Port D is a general purpose bidirectional I/O port. Each of the Port D pins can be individually configured as either a general purpose I/O pin, a primary memory control signal (Function A), or a secondary memory control signal (Function B). All chip selects are independent and can be used for memory bank switching.
	A	B			
D0	$\overline{CSE2}$	$\overline{OCF}$	64	I/O	I/O Pin, *or* Function A: Chip Select Eighth output 2 goes low during memory accesses to 2000h–3FFFh, *or* Function B: Opcode fetch goes low during the opcode fetch memory cycle.
D1	$\overline{CSH3}$		60	I/O	I/O Pin, *or* Function A: Chip Select Half output 3 goes low during memory accesses to 8000h–FFFFh.
D2	$\overline{CSH2}$		59	I/O	I/O Pin, *or* Function A: Chip Select Half output 2 goes low during memory accesses to 8000h–FFFFh.
D3	CLKOUT	CLKOUT	58	I/O	I/O Pin, *or* Functions A & B: Internal clock signal is 1/4 XTAL2/CLKIN frequency.
D4	R/$\overline{W}$	R/$\overline{W}$	57	I/O	I/O Pin, *or* Function A & B: Read/Write output pin.
D5	$\overline{CSPF}$		56	I/O	I/O Pin, *or* Function A: Chip Select Peripheral output for peripheral file; goes low during memory accesses to 10C0h–10FFh.
D6	$\overline{CSH1}$	$\overline{EDS}$	55	I/O	I/O Pin, *or* Function A: Chip Select Half output 1 goes low during memory accesses to 8000h–FFFFh, *or* Function B: External Data Strobe output goes low during memory accesses from external memory and has the same timings as the five chip selects.
D7	$\overline{CSE1}$	$\overline{WAIT}$	54	I/O	I/O Pin, *or* Function A: Chip Select Eighth output 1 goes low during memory accesses to 2000h–3FFFh, *or* Function B: Wait input pin extends bus signals.
INT1	INTIN		52	I	External interrupt (non-maskable or maskable)/General purpose input pin.
INT2	INTIO1		51	I/O	External maskable interrupt input/General purpose bidirectional pin.
INT3	INTIO2		50	I/O	External maskable interrupt input/General purpose bidirectional pin.
T1IC/CR	T1IO1		46	I/O	Timer 1 input Capture/Counter Reset input pin/General purpose bidirectional pin.
T1PWM	T1IO2		45	I/O	Timer 1 PWM output pin/General Purpose bidirectional pin.
T1EVT	T1IO3		44	I/O	Timer 1 External Event input pin/General purpose bidirectional pin.
T2IC1/CR	T2IO1		27	I/O	Timer 2 input Capture 1/Counter Reset input pin/General purpose bidirectional pin.
T2IC2/PWM	T2IO2		26	I/O	Timer 2 input Capture 2 input pin/PWM output pin/General purpose bidirectional pin.
T2EVT	T2IO3		25	I/O	Timer 2 External Event input pin/General purpose bidirectional pin.

TERMINAL FUNCTIONS (continued)

PIN			I/O	DESCRIPTION
NAME	ALTERNATE FUNCTION	NO.		
SPISOMI	SPIIO1	49	I/O	SPI Slave Output pin, Master Input pin/General purpose bidirectional pin.
SPISIMO	SPIIO2	48	I/O	SPI Slave Input pin, Master Output pin/General purpose bidirectional pin.
SPICLK	SPIIO3	47	I/O	SPI bidirectional Serial Clock pin/General purpose bidirectional pin.
SCITXD	SCIIO1	30	I/O	SCI Transmit Data output pin/General purpose bidirectional pin.
SCIRXD	SCIIO2	29	I/O	SCI Receive Data input pin/General purpose bidirectional pin.
SCICLK	SCIIO3	28	I/O	SCI bidirectional Serial Clock pin/General purpose bidirectional pin.
AN0	E0	36	I	A/D analog input (AN0 – AN7) or positive reference pins (AN1 – AN7). Port E may be individually programmed as general purpose input pins if not used as A/D converter analog input or positive reference input..
AN1	E1	37	I	
AN2	E2	38	I	
AN3	E3	39	I	
AN4	E4	40	I	
AN5	E5	41	I	
AN6	E6	42	I	
AN7	E7	43	I	
V_{CC3}		34		A/D converter positive supply voltage and optional positive reference input pin.
V_{SS3}		35		A/D converter ground supply and low reference input pin.
$\overline{RESET}$		53	I/O	System reset bidirectional pin. As input it initializes microcontroller, as open drain output it indicates an internal failure was detected by the Watchdog or Oscillator Fault circuit.
MC		6	I	Microprocessor/Microcomputer mode control input pin, also enables EEPROM Write Protection Override (WPO) mode. For devices with EPROM, this pin is used for V_{PP} external supply for EPROM programming.
XTAL2/CLKIN		31	I	Internal oscillator crystal input/External clock source input.
XTAL1		32	O	Internal oscillator output for crystal.
V_{CC1}		33,61		Positive supply voltage for digital logic.
V_{CC2}		15,63		Positive supply voltage for digital I/O pins.
V_{SS1}		9		Ground reference for digital logic.
V_{SS2}		16,62		Ground reference for digital I/O pins.

NOTE 1: Each pin associated with the Interrupt, Timer 1, Timer 2, SPI, and SCI functional blocks may be individually programmed as a general purpose bidirectional pin if it is not used for its primary block function.

operating modes

The TMS370Cx5x has four operating modes, two basic modes with each mode having two memory configurations. The basic operating modes are the microcomputer and microprocessor modes, which are selected by the voltage level applied to the dedicated MC pin two cycles before the $\overline{\text{RESET}}$ pin goes inactive. The two memory configurations are then selected through software programming of the internal system configuration registers. The four operating modes are the microcomputer single chip, expanded microcomputer, microprocessor, and microprocessor with internal program memory. The TMS370Cx58 supports only the microcomputer single chip mode.

In the **microcomputer single chip mode**, the TMS370Cx5x functions as a self-contained microcomputer with all memory and peripherals on chip, increasing the number of pins for direct I/O control applications.

The **expanded microcomputer mode** supports bus expansion to external memory or peripherals, while all on-chip memory (RAM, ROM, EPROM, and data EEPROM) remains active. Digital I/O ports (Ports A,B,C, and D), under control of their associated port control registers, can be configured via software to become the external 16-bit address bus, 8-bit data bus, and control interface. In applications where the entire address, data, or control bus is not required, each of these pins can be individually programmed as a general-purpose input/output or as its associated memory expansion alternate function, thereby maximizing the pins available for general-purpose input/output. The address bus and data bus are not multiplexed, eliminating the requirement for an external address/data latch and lowering the system cost.

Further reductions in external interface decode logic can be realized by using the precoded chip-select outputs that provide direct memory/peripheral chip-select or chip-enable functions. When memory accesses are performed to any location between 2000h and 3FFFh, pins $\overline{\text{CSE1}}$ and $\overline{\text{CSE2}}$ will become active if enabled by their port control registers. Similarly, memory accesses to any location between 8000h and FFFFh will activate $\overline{\text{CSH1}}$, $\overline{\text{CSH2}}$, and $\overline{\text{CSH3}}$ if enabled by their respective port control registers. As a result, up to 96K bytes of external memory can be mapped into the 32K-byte logical address space of 8000h to FFFFh by using $\overline{\text{CSH1}}$, $\overline{\text{CSH2}}$, and $\overline{\text{CSH3}}$ as memory bank selects under software control.

In the **microprocessor mode**, Ports A,B,C, and D are the address, data, and control buses for interface to external memory and peripherals. In this mode these ports are not programmable. The on-chip RAM and data EEPROM remain active, while the on-chip ROM, or EPROM is disabled. The program area and the reset, interrupt, and trap vectors are located in off-chip memory locations.

The **microprocessor mode with internal program memory** is configured just as the microprocessor mode with respect to the external bus interface. However, the application program in external memory enables the internal program ROM or EPROM to also to be active in the system. This is accomplished by writing a zero to the MEMORY DISABLED control bit (SCCR1.2) of the SCCR1 control register.

Operating Mode Summary

OPERATING MODE	MC PIN	$\overline{\text{RESET}}$	OTHER
Microcomputer single chip	Low	↑	None
Microcomputer with expanded memory	Low	↑	Set digital I/O registers to function A or B†
Microprocessor	High	↑	None
Microprocessor with internal memory	High	↑	Enable internal memory (Clear SCCR1.2)

† Function A: Port D = chip select signals $\overline{\text{CSE1}}$, $\overline{\text{CSE2}}$, $\overline{\text{CSH1}}$, $\overline{\text{CSH2}}$, $\overline{\text{CSH3}}$ and $\overline{\text{CSPF}}$
Function B: Port D = Expansion memory control signals $\overline{\text{OCF}}$, $\overline{\text{EDS}}$ and $\overline{\text{WAIT}}$

memory/peripheral wait operation

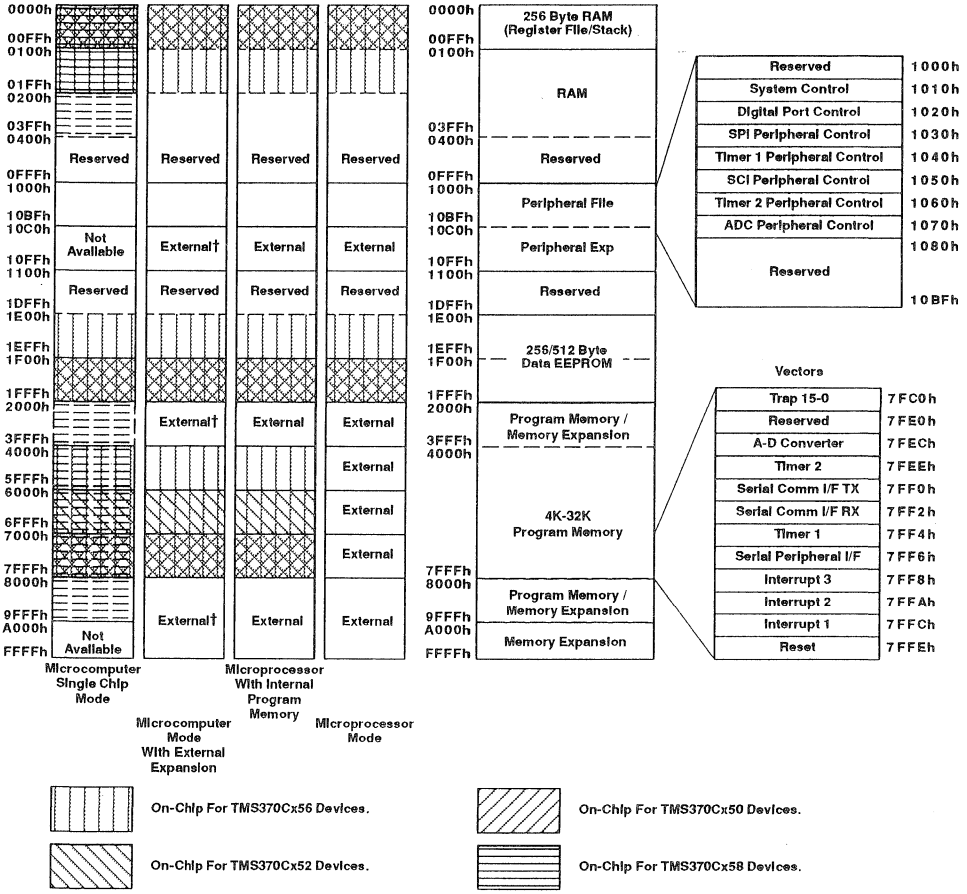
The TMS370Cx5x enhances interface flexibility by providing $\overline{\text{WAIT}}$ state support, thereby decoupling the cycle time of the CPU from the read/write access of the external memory or peripherals. External devices can extend their read/write accesses indefinitely by asserting the $\overline{\text{WAIT}}$ input pin, and the CPU will continue to wait as long as this signal remains active.

Programmable automatic wait state generation is also provided by the TMS370Cx5x on-chip bus controller. The TMS370Cx5x is configured, following hardware reset, to automatically add one wait state to all external bus transactions, and memory and peripheral accesses, thus making every external access a minimum of three system clock cycles. The designer can disable the automatic wait state generation if the AUTOWAIT DISABLE bit in SCCR1 is set to 1. Also, all accesses to the upper four frames of the peripheral file can be independently extended to four system clock cycles if the PF AUTO WAIT bit in SCCR1 is set to 1. Programmable wait states can be used in conjunction with the external $\overline{\text{WAIT}}$ pin. In applications where the external device read/write access can interface with the TMS370Cx5x CPU using one wait state, the automatic wait state generation can eliminate external $\overline{\text{WAIT}}$ interface logic, lowering system cost.

memory map

The TMS370 family architecture is based on the Von Neumann architecture, where the program memory and data memory share a common address space. All peripheral input/output is memory mapped into this same common address space. In the expansion mode, external memory or peripherals are also memory mapped into this common address. As shown in Figure 1, the TMS370 provides a 16-bit address range to access internal or external RAM, ROM, EPROM, EEPROM, input/output pins, and peripheral functions.

The peripheral file contains all input/output port control, on-and off-chip peripheral status and control, EPROM memory programming, and system-wide control functions. The peripheral file consists of 256 contiguous addresses located from 1000h to 10FFh. This page of 256 contiguous addresses is logically divided into 16 Peripheral File Frames of 16 bytes each. Each on-chip peripheral is assigned to a separate frame through which peripheral control and data information is passed. The TMS370Cx5x has its on-chip peripherals and system control assigned to Peripheral File Frames 1 through 7, addresses 1010h through 107Fh.



† Precoded Chip Select Outputs Available on External Expansion Bus.

NOTE: The term ***Reserved*** in this figure means the address space is reserved for future expansion, while the term ***Not Available*** means the address space is unavailable in the particular mode illustrated by that block.

Figure 1. TMS370Cx5x Memory Map

on-chip memories

RAM/register file

The TMS370Cx5x family has up to 1K bytes of on-chip static RAM, addressed as 1024 consecutive bytes mapped from location 0000h to 03FFh. The first 256 bytes (0000h-00FFh) serve as both the CPU register file and general purpose memory, the next 768 bytes (0100h-03FFh) serve as general purpose RAM. The first 256 bytes of RAM are treated as registers by the instruction set and are referenced as R0 through R255. The first two registers, R0 and R1, are also called the A and B registers, respectively. The stack is located in the on-chip RAM, and operates as a last-in first-out read/write memory. It is used to store the return address on subroutine calls and the status register during interrupts. Accessing this memory as registers is performed in one system clock cycle (t_c), while general purpose memory access is performed in two system clock cycles.

Instructions may be executed from RAM. This versatility enables the internal RAM to be used for functions such as microcontroller self-test, diagnostics, or system test of the end application. The user may load external programs or data into the RAM by incorporating a simple bootstrap loader in the program memory, or by operating the microcontroller in the microprocessor mode.

data EEPROM

The TMS370Cx5x family has up to 512 bytes of on-chip Electrically Erasable Programmable ROM (EEPROM), addressed as 512 consecutive bytes mapped from locations 1E00h to 1FFFh (1F00h to 1FFFh for devices with 256 bytes). The data EEPROM provides nonvolatile programmable storage for items such as calibration constants and configuration information for personalization of a generic program ROM/EPROM algorithm for use in specific end applications. The data EEPROM supports bit, byte, and block write/erase modes. Instructions may be executed from the data EEPROM, providing additional program space and the ability to patch algorithms by placing a branch table for volatile routines in the data EEPROM.

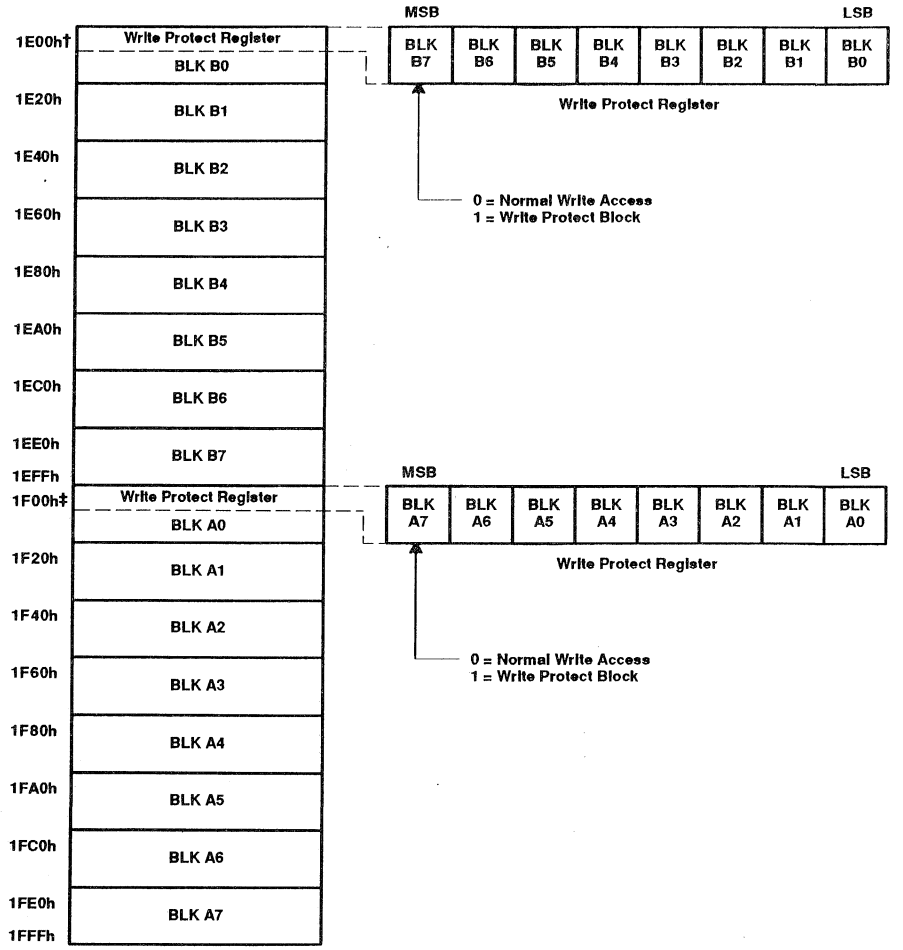
The data EEPROM uses the 5-V V_{CC} supply voltage and provides the programming voltage via an internal dedicated generator, eliminating the need for an external high-voltage programming source. The dedicated voltage generator optimizes the programming voltage characteristics, increasing the reliability as well as extending the write/erase endurance of the array.

Programming control and status monitoring are performed through the data EEPROM control register (DEECTL) in the peripheral file. An EEPROM write/erase operation is performed in the following sequence:

1. Perform normal memory write to the target EEPROM location.
2. Write to DEECTL control register to select WRITE1/WRITE0 and set the EXECUTE (EXE) bit to 1.
3. Wait for program time to elapse [$t_{w(PGM)B}$ or $t_{w(PGM)AR}$].
4. Write to DEECTL control register to set the EXECUTE (EXE) bit to 0.

The WRITE1/WRITE0 control bit selects whether the zeros or the ones in the data byte are to be programmed into the selected EEPROM location. For example, a WRITE1 operation will program ones into all bit positions within the EEPROM byte that have ones in the data byte, while bits that are zero in the data byte will not affect the EEPROM contents. The WRITE1 operation effectively performs a logical OR of the information previously stored on the EEPROM byte with the data byte. The WRITE0 operation effectively performs a logical AND between these two bytes. Single bit programming within an EEPROM byte is performed by writing only the zeros or ones of the data byte. The EEPROM programming algorithm may use this bit-programming capability to optimize the useful life of the EEPROM.

When a data value cannot be achieved by writing only zeros or only ones into the EEPROM byte, a WRITE1 followed by a WRITE0 will program any data value into the EEPROM byte, regardless of the previous data stored at that location.



† For devices with 512 bytes data EEPROM, the first first block of 32 bytes begins at loaction 1E00h and the last block ends at location 1FFFh.

‡ For devices with 256 bytes data EEPROM, the first first block of 32 bytes begins at loaction 1F00h and the last block ends at location 1FFFh.

Figure 2. Write Protect Register for TMS370Cx5x Devices With a Data EEPROM Array

Data EEPROM read accesses are performed as normal memory read operations in two system clock cycles. A memory read cycle to any EEPROM location while EXE = 1 returns the value currently being written to the EEPROM. Following an EEPROM write operation, the EEPROM voltages must stabilize prior to performing an EEPROM read operation. The BUSY FLAG indicates the status of the EEPROM voltage. When set, the EEPROM is not ready for a read operation. The BUSY flag is reset to 0 by the EEPROM control logic when 128 system clock cycles have elapsed following the EXE bit being set to 0. If an EEPROM read operation is performed while BUSY = 1, automatic WAIT states will be generated until BUSY = 0, and then the read operation will be performed.

Bytes within the data EEPROM can be protected from inadvertent overwriting of critical information. As shown in Figure 2, the 8-bit Write Protect Register (WPR), located at 1F00h within the data EEPROM, provides write protection for the TMS370Cx50 and TMS370Cx52 devices containing a 256-byte data EEPROM array, segmenting the array into eight blocks of 32 bytes each. The TMS370Cx56 devices containing a 512-byte data EEPROM array possess an additional Write Protection Register (see Figure 2) located at 1E00h within the data EEPROM, which will also segment the additional 256-byte array into eight blocks of 32 bytes each. Each of these 32-byte blocks may be individually write- and erase-protected by setting the corresponding bit to 1 in the appropriate WPR. Since the WPRs reside in the array in BLK A0 and BLK B0, the WPRs may also be write-protected, thereby increasing the system reliability by preventing bytes from being reprogrammed. Bytes left unprotected may be written to by the normal EEPROM programming sequence. The Write Protection Override (WPO) mode enables data to be written to any location in the data EEPROM, regardless of the WPR contents. The WPO mode is typically used in a service environment to update the protected EEPROM contents.

All unprotected bytes within the data EEPROM array may be programmed during a single EEPROM programming cycle by setting the ARRAY PROG bit DEECTL to 1 at the start of the programming cycle for TMS370Cx5x devices with a 256-byte data EEPROM array. The TMS370Cx5x devices with a 512-byte data EEPROM array must be in the WPO mode to enable array programming.

program ROM

The program ROM consists of 4K to 16K bytes of mask programmable read-only memory. The program ROM is used for permanent storage of data or instructions, with read operations performed in two system clock cycles. Memory addresses 7FECh through 7FFFh are reserved for interrupt and reset vectors. Memory locations 7FE0h through 7FEBh are reserved for factory use. Trap vectors, used with TRAP0 through TRAP15 instructions are located between addresses 7FC0h and 7FDFh. Programming of the mask ROM is performed at the time of device fabrication.

program EPROM (TMS370C756 and TMS370C758)

The program EPROM of the TMS370C756 and the TMS370C758 are 16K and 32K electrically programmable read-only memory, addressed as 16K consecutive bytes mapped from location 4000h to 7FFFh (TMS370C756) or 32K consecutive bytes mapped from location 2000h to 9FFFh (TMS370C758). It provides application performance identical to the TMS370Cx5x mask ROM devices with up to 32K bytes. Program instructions are read from the program EPROM in two system clock cycles, providing the prototyping capability of the mask program ROM.

An external supply is needed at the MC pin to provide the necessary programming voltage (V_{PP}). Programming is controlled through a register (EPCTL) in the peripheral file.

The TMS370C756FN and TMS370C758FN come in a plastic package and cannot be erased. They are one-time-programmable (OTP). The TMS370C756FZ and TMS370C758FZ come in a ceramic package with a quartz window. Before programming, the device's EPROM is erased by exposing the device through the transparent window to high-intensity ultraviolet light (wavelength 2537 Å). The recommended minimum exposure dose (UV intensity × exposure time) is 15W•s/cm^2. A typical 12-mW/cm^2, filterless UV lamp will erase the device in 21 minutes. The lamp should be located about 2.5 cm above the chip during erasure. After erasure, the entire array is in logic 1 state. A programmed logic 0 can be erased to a logic 1 only by exposure to ultraviolet

light. It should be noted that normal ambient light contains the correct wavelength for erasure. Therefore, when using the TMS370C756FZ or the TMS370C758FZ, the window should be covered with an opaque label. All devices are erased to a logic high upon delivery from the factory.

CAUTION

Exposing the EPROM module to ultraviolet light may also cause erasure in any EEPROM module. Any useful data stored in the EEPROM must be reprogrammed after exposure to UV light.

Programming lows into the EPROM is controlled by the EPCTL register via the EXE bit and the VPPS bit. The EXE bit initiates EPROM programming when set and disables programming when cleared. The VPPS bit connects the programming voltage V_{PP} at the MC pin to the EPROM module. VPPS (EPCTL.6) and EXE (EPCTL.0) should be set separately, and the VPPS bit should be set at least two microseconds before the EXE bit is set. After programming, the application programming should wait for four microseconds before any read attempt is made. The programming operation (see Figure 3) is performed in the following recommended sequence:

1. Supply the programming voltage to the MC pin.
2. Write to the EPCTL register to set the VPPS bit to 1.
3. Perform normal memory write register to the target EPROM location.
4. Write to the EPCTL register to set the EXE bit register to 1. (Wait at least two microseconds after step 2.)
5. Wait for program time to elapse (one millisecond).
6. Write to the EPCTL register to clear the EXE bit (leave VPPS bit set to 1).
7. Read the byte being programmed; if correct data is not read, repeat steps 4 through 6 up to a maximum X of 25.
8. Write to the EPCTL register to set the EXE bit to 1 for final programming.
9. Wait for program time to elapse (3× milliseconds duration).
10. Write to the EPCTL register to clear the EXE and VPPS bits.

An external power supply at V_{PP}, I_{PP} (30 mA), is required for programming operation. Programming voltage V_{PP} is supplied via the MC pin. This also automatically puts the microcontroller in the Write Protection Override (WPO) mode. Programming voltage may be applied via the MC pin anytime after $\overline{\text{RESET}}$ and remain at V_{PP} after programming (after the EXE bit is cleared). Applying programming voltage while $\overline{\text{RESET}}$ is active will put the microcontroller in a reserved mode, where programming operation is inhibited.

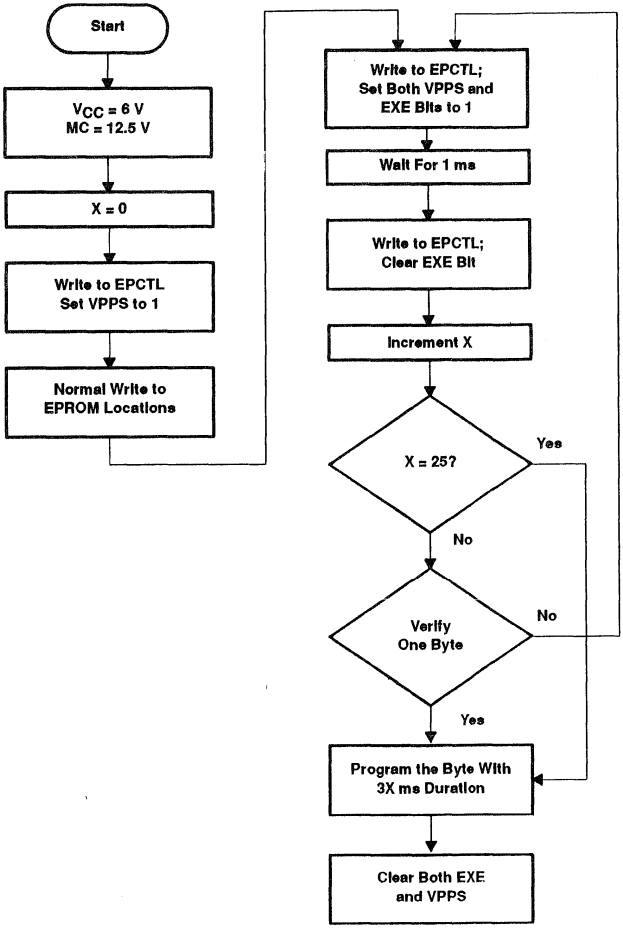


Figure 3. EPROM Programming Operation

central processing unit

The central processing unit (CPU) of the TMS370 family is an enhanced version of the TMS7000 Family CPU. The enhancements include additional user instructions such as integer divide, conditional jump instructions based on the overflow status bit, and addressing modes such as stack-pointer-relative addressing for subroutine parameter passing. The efficient register-to-register architecture of the TMS7000 family has been carried over to the TMS370 family to avoid the conventional accumulator bottleneck. The complete TMS370 family instruction set is summarized in the *TMS370 Instruction Set Summary*, page 47.

In addition to the interpretation and execution of the user program, the CPU performs the functions of bus protocol generation and interrupt priority arbitration. While the CPU is implemented independent of the memory, input/output, and peripheral modules, it performs the central system control function through communications with these on-chip modules and external memory and peripherals.

The TMS370 family CPU registers accessible to the programmer are shown in Figure 4. The register file consists of 256 general purpose registers, R0 through R255, implemented in on-chip RAM, and is used by the CPU for general purpose 8- and 16-bit source and destination operands, index registers, and indirect addressing. The first two registers, R0 and R1, are also called registers A and B and are used by the CPU for general purpose registers or for implied operands. The program counter (PC) contains the address of the next instruction to be executed. The stack pointer (SP) contains the address of the last or top entry on the stack, which is located in the on-chip register file. The status register (ST) contains four bits that reflect the outcome of the instruction just executed, and two bits that control the masking of the interrupt priority chains.

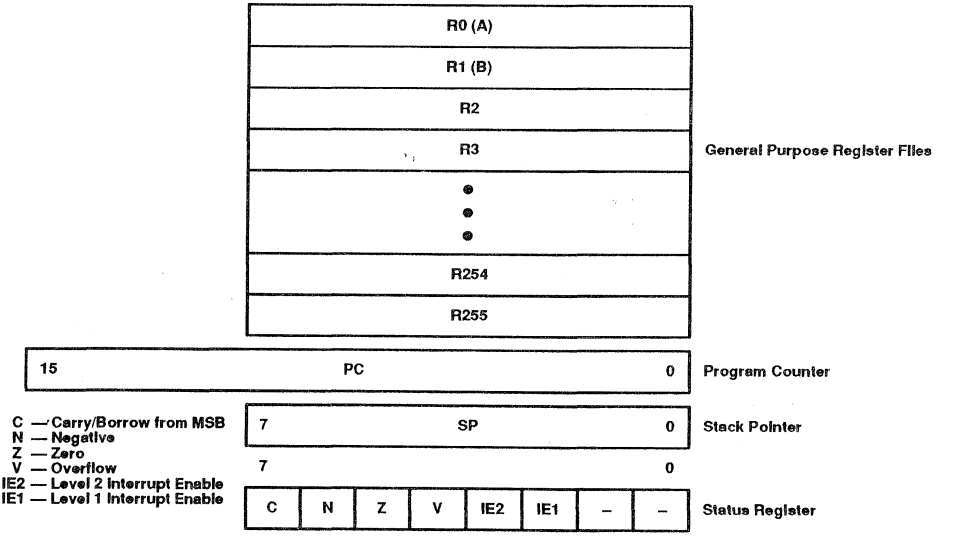


Figure 4. CPU Registers

TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77001

system resets

The TMS370Cx5x has three possible reset sources: a low input to the $\overline{\text{RESET}}$ pin, a programmable watchdog timer timeout, or a programmable oscillator fault failure. The $\overline{\text{RESET}}$ pin, an input/output pin, initiates TMS370Cx5x hardware initialization and ensures an orderly software startup. A low level input of at least 50 ns initiates the reset sequence. The microcontroller is held in reset until the $\overline{\text{RESET}}$ pin goes inactive (high). If the $\overline{\text{RESET}}$ input signal is low for less than eight system clock cycles, the TMS370Cx5x will hold the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The $\overline{\text{RESET}}$ pin must be activated by the application at power-up, which can be accomplished by an external input or an RC power-up reset circuit. Recall that the basic operating mode, microcomputer or microprocessor, is determined by the voltage level applied to the MC pin two cycles before the $\overline{\text{RESET}}$ pin goes inactive (high). The $\overline{\text{RESET}}$ pin can be asserted at any time during operation, resulting in an immediate initiation of the reset sequence.

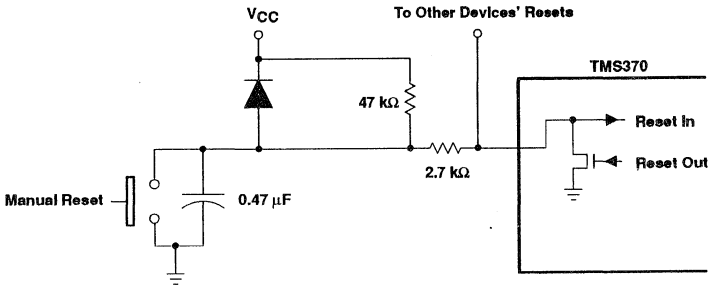


Figure 5. Typical Reset Circuit

The watchdog timer provides system software integrity by detecting a program that has become lost or is not executing as expected. A system reset is generated if the watchdog timer is not properly re-initialized by a specific software sequence, or if the re-initialization does not occur before the watchdog timer times out. The watchdog timer timeout initiates the TMS370Cx5x reset sequence and drives the external $\overline{\text{RESET}}$ pin low for eight system clock cycles to reset external system components. The watchdog reset function is enabled by setting WD OVRFL RST ENA bit of T1CTL2 to 1. Once the software enables the watchdog reset function, subsequent writes to the WD OVRFL RST ENA bit are ignored. Watchdog control bits can be initialized only following a powerup reset. The timer section discusses additional information on the watchdog timer and its configurations.

The oscillator fault circuit provides the means to monitor failures of the oscillator input signal (XTAL2/CLKIN). This function is enabled under software control by setting the OSC FLT RST ENA bit of SCCR2 to 1. If the oscillator input signal frequency remains above the 90% point of the minimum operating frequency (CLKIN), the oscillator input will not to be activated. However, if the oscillator input is lost or its frequency falls below 20 kHz and the oscillator fault reset is enabled, the TMS370Cx5x is reset and the external $\overline{\text{RESET}}$ pin is driven low.

Reset Sources

REGISTER	ADDRESS	PF	BIT #	CONTROL BIT	SOURCE OF RESET
SCCR0	1010h	P010	7	COLD START	Cold or Warm start reset
SCCR0	1010h	P010	4	OSC FLT FLAG	Oscillator out of range
T1CTL2	104Ah	P04A	5	WD OVRFL INT FLAG	Watchdog timer timeout

When an oscillator input failure occurs, the internal clocks are stopped and $\overline{\text{RESET}}$ is held active until the oscillator input frequency is greater than 100 kHz typical. If the OSC FLT RST ENA bit of SCCR2 is set to 0, the fault detection circuit independently sets the OSC FLT FLAG of SCCR0 without generating a system reset. The OSC FLT RST ENA bit is protected during non-privileged operation and therefore should be software configured during the initialization sequence following system reset. During a HALT mode, the oscillator fault circuitry will be disabled.

During a microcontroller reset. the majority of the peripheral file bits are set to 0, with the exception of the bits shown in the following table. During all reset, the COLD START, OSC FLT FLAG, and the WD OVRFL FLAG are appropriately set by the active reset and may be interrogated by the program to determine the source of the system reset. Registers A and B are set to zero during all resets. The other registers are not affected by a reset under power (warm reset).

Control Bit States Following Reset

REGISTER	CONTROL BIT	POWER-UP	WARM RESET	
			MICROCOMPUTER	MICROPROCESSOR
SCCR0	µP/µC MODE	0	0	1
SCCR0	MC PIN DATA	0	0	1
SCCR0	COLD START	1	†	†
SCCR0	OSC FLT FLAG	0	†	†
T1CTL2	WD OVRFL FLAG	0	†	†
TXCTL	TX EMPTY	1	1	1
TXCTL	TXRDY	1	1	1
ADSTAT	AD READY	1	1	1

† Status bit corresponding to active reset source is set to 1.

Interrupts

The TMS370 family software programmable interrupt structure supports flexible on-chip and external interrupt configurations to meet realtime interrupt-driven application requirements. The hardware interrupt structure incorporates two priority levels as shown in Figure 6. Interrupt level 1 has a higher priority than interrupt level 2. The two priority levels can be independently masked by the global interrupt mask bits (IE1 and IE2) of the Status Register.

Each system interrupt is independently configured on either the high or low priority chain by the application program during system initialization. Within each interrupt chain, the interrupt priority is fixed by the position of the system interrupt. However, since each system interrupt is selectively configured on either the high or low priority interrupt chain, the application program can elevate any system interrupt to the highest priority. Arbitration between the two priority levels is performed within the CPU. Arbitration within each of the priority chains is performed within the peripheral modules to support interrupt expansion to future modules. Pending interrupts are serviced upon completion of current instruction execution, depending on their interrupt mask and priority conditions.

The TMS370Cx5x has ten hardware system interrupts as shown in the following table (page 18). Each system interrupt has a dedicated interrupt vector located in program memory through which control is passed to the interrupt service routines. A system interrupt may have multiple interrupt sources. (e.g., SCI RXINT has two interrupt sources). All of the interrupt sources are individually maskable by local interrupt enable control bits in the associated peripheral file. Each interrupt source FLAG bit is individually readable for software polling or to determine which interrupt source generated the associated system interrupt.

Hardware System Interrupts

INTERRUPT SOURCE	INTERRUPT FLAG	SYSTEM INTERRUPT	VECTOR INTERRUPT	PRIORITY§
External $\overline{\text{RESET}}$ Watchdog Overflow Oscillator Fault Detect	COLD START WD OVRFL INT FLAG OSC FLT FLAG	$\overline{\text{RESET}}$†	7FFEh, 7FFFh	1
External INT1	INT1 FLAG	INT1†	7FFCh, 7FFDh	2
External INT2	INT2 FLAG	INT2†	7FFAh, 7FFBh	3
External INT3	INT3 FLAG	INT3†	7FF8h, 7FF9h	4
SPI RX/TX Complete	SPI INT FLAG	SPIINT	7FF6h, 7FF7h	5
Timer 1 Overflow Timer 1 Compare 1 Timer 1 Compare 2 Timer 1 External Edge Timer 1 Input Capture Watchdog Overflow	T1 OVRFL INT FLAG T1C1 INT FLAG T1C2 INT FLAG T1EDGE INT FLAG T1IC INT FLAG WD OVRFL INT FLAG	T1INT‡	7FF4h, 7FF5h	6
SCI RX Data Register Full SCI RX Break Detect	RXRDY FLAG BRKDT FLAG	RXINT†	7FF2h, 7FF3h	7
SCI TX Data Register Empty	TXRDY FLAG	TXINT	7FF0h, 7FF1h	8
Timer 2 Overflow Timer 2 Compare 1 Timer 2 Compare 2 Timer 2 External Edge Timer 2 Input Capture 1 Timer 2 Input Capture 2	T2 OVRFL INT FLAG T2C1 INT FLAG T2C2 INT FLAG T2EDGE INT FLAG T2IC1 INT FLAG T2IC2 INT FLAG	T2INT	7FEEh, 7FEFh	9
A-D Conversion Complete	AD INT FLAG	ADINT	7FECh, 7FEDh	10

† Releases microcontroller from STANDBY and HALT low power modes.
‡ Releases microcontroller from STANDBY low power mode.
§ Relative priority within an interrupt level.

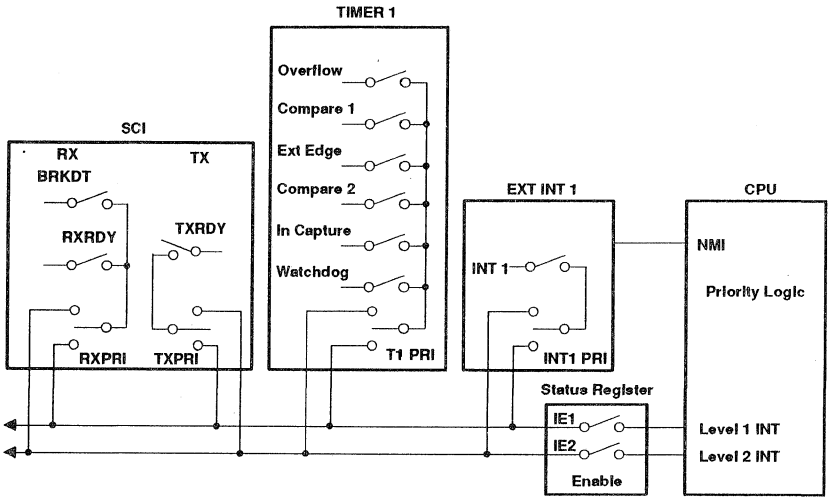


Figure 6. Interrupt Control

Six of the system interrupts are generated by on-chip peripheral functions, and three external interrupts are supported. Software configuration of the external interrupts is performed through the INT1, INT2, and INT3 control registers in peripheral file frame 1. Each external interrupt is individually software configurable for input polarity (rising or falling) for ease of system interface. External interrupt INT1 is software configurable as either a maskable or non-maskable interrupt. When INT1 is configured as non-maskable, it cannot be masked by the individual or global enable bits. Recall that the INT1 NMI bit is protected during nonprivileged operation and therefore should be configured during the initialization sequence following reset. To maximize pin flexibility, external interrupts INT2 and INT3 can be software configured as general purpose input/output pins if the interrupt function is not required (INT1 can be similarly configured as an input pin).

privileged operation and EEPROM write protection override

The TMS370Cx5x family is designed with significant flexibility to enable the designer to software-configure the system and peripherals to meet the requirements of a broad variety of applications. The non-privileged mode of operation ensures the integrity of the system configuration once defined for an end application. Following a hardware reset, the TMS370Cx5x operates in the privileged mode, where all peripheral file registers have unrestricted read/write access and the application program will configure the system during the initialization sequence following reset. As the last step of system initialization, the PRIVILEGE DISABLE bit (SCCR2.0) will be set to 1, entering the non-privileged mode and disabling write operations to specific configuration control bits within the peripheral file. The following system configuration bits are write-protected during the non-privileged mode and must be configured by software prior to exiting the privileged mode:

REGISTER†		CONTROL BIT
NAME	LOCATION	
SCCR0	P010.5	PF AUTO-WAIT
SCCR0	P010.6	OSC POWER
SCCR1	P011.2	MEMORY DISABLE
SCCR1	P011.4	AUTOWAIT DISABLE
SCCR2	P012.0	PRIVILEGE DISABLE
SCCR2	P012.6	PWRDWN/IDLE
SCRR2	P012.7	HALT/STANDBY
SCCR2	P012.1	INT1 NMI
SCRR2	P012.2	OSC FLT DISABLE
SCCR2	P012.5	OSC FLT RST ENA
SPIPRI	P03F.6	SPI PRIORITY
SPIPRI	P05F.6	SCI TX PRIORITY
SPIPRI	P05F.5	SCI RX PRIORITY
T1PRI	P04F.6	T1 PRIORITY
T2PRI	P06F.6	T2 PRIORITY
ADPRI	P07F.6	AD PRIORITY

† Identified by name and bit location within the register

The privileged bits are shown in a **bold typeface** in the following section Peripheral File Frame 1.

The Write Protection Override (WPO) mode provides an external hardware method of overriding the Write Protection Registers (WPR) of the data EEPROM on the TMS370Cx5x. The WPO mode is entered by applying a 12-V input to the MC pin after the $\overline{\text{RESET}}$ pin input goes high. The high voltage on the MC pin during the WPO mode is not the programming voltage for the data EEPROM or program EPROM. All EEPROM programming voltages are generated on-chip. The WPO mode provides hardware system level capability to modify the personality or calibration information in the data EEPROM while the device remains in the application, but only while requiring a 12-volt external input on the MC pin (normally not available in the end application except in a service or diagnostic environment).

peripheral file frame 1

Peripheral File Frame 1 contains system configuration and control functions and registers for controlling EEPROM programming. The privileged bits are shown in a **bold typeface** in the Peripheral File Frames.

Peripheral File Frame 1: System Configuration and Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1010h	P010	COLD START	**OSC POWER**	**PF AUTO WAIT**	OSC FLT FLAG	MC PIN WPO	MC PIN DATA	—	µP/µC MODE	SCCR0
1011h	P011	—	—	—	**AUTOWAIT DISABLE**	—	**MEMORY DISABLE**	—	—	SCCR1
1012h	P012	**HALT/ STANDBY**	**PWRDWN/ IDLE**	**OSC FLT RST ENA**	**BUS STEST**	**CPU STEST**	**OSC FLT DISABLE**	**INT1 NMI**	**PRIVILEGE DISABLE**	SCCR2
1013h to 1016h	P013 to P016	Reserved								
1017h	P017	INT1 FLAG	INT1 PIN DATA	—	—	—	INT1 POLARITY	INT1 PRIORITY	INT1 ENABLE	INT1
1018h	P018	INT2 FLAG	INT2 PIN DATA	—	INT2 DATA DIR	INT2 DATA OUT	INT2 POLARITY	INT2 PRIORITY	INT2 ENABLE	INT2
1019h	P019	INT3 FLAG	INT3 PIN DATA	—	INT3 DATA DIR	INT3 DATA OUT	INT3 POLARITY	INT3 PRIORITY	INT3 ENABLE	INT3
101Ah	P01A	BUSY	—	—	—	—	AP	W1W0	EXE	DEECTL
101Bh	P01B	Reserved								
101Ch	P01C	BUSY	VPPS	—	—	—	—	W0	EXE	EPCTL
101Dh 101Eh 101Fh	P01D P01E P01F	Reserved								

peripheral file frame 2

Peripheral File Frame 2 contains the digital I/O pin configuration and control registers. The following figure details the specific addresses, registers, and control bits within this Peripheral File Frame.

Peripheral File Frame 2: Digital Port Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1020h	P020	Reserved								APORT1
1021h	P021	Port A Control Register 2								APORT2
1022h	P022	Port A Data								ADATA
1023h	P023	Port A Direction								ADIR
1024h	P024	Reserved								BPORT1
1025h	P025	Port B Control Register 2								BPORT2
1026h	P026	Port B Data								BDATA
1027h	P027	Port B Direction								BDIR
1028h	P028	Reserved								CPORT1
1029h	P029	Port C Control Register 2								CPORT2
102Ah	P02A	Port C Data								CDATA
102Bh	P02B	Port C Direction								CDIR
102Ch	P02C	Port D Control Register 1								DPORT1
102Dh	P02D	Port D Control Register 2								DPORT2
102Eh	P02E	Port D Data								DDATA
102Fh	P02F	Port D Direction								DDIR

Port Configuration Registers Set-up

PORT	PIN	INPUT	OUTPUT	FUNCTION A	FUNCTION B (μP MODE)
		XPORT1 = 0† XPORT2 = 0 XDATA = y XDIR = 0	XPORT1 = 0† XPORT2 = 0 XDATA = q XDIR = 1	XPORT1 = 0† XPORT2 = 1 XDATA = x XDIR = x	XPORT1 = 1† XPORT2 = 1 XDATA = x XDIR = x
A	0 – 7	Data In y	Data Out q	Data Bus	Reserved
B	0 – 7	Data In y	Data Out q	Low ADDR	Reserved
C	0 – 7	Data In y	Data Out q	HI ADDR	Reserved
D	0	Data In y	Data Out q	$\overline{CSE2}$	$\overline{OCF}$
D	1	Data In y	Data Out q	$\overline{CSH3}$	
D	2	Data In y	Data Out q	$\overline{CSH2}$	
D	3	Data In y	Data Out q	CLKOUT	CLKOUT
D	4	Data In y	Data Out q	R/$\overline{W}$	R/$\overline{W}$
D	5	Data In y	Data Out q	$\overline{CSPF}$	
D	6	Data In y	Data Out q	$\overline{CSH1}$	$\overline{EDS}$
D	7	Data In y	Data Out q	$\overline{CSE1}$	$\overline{WAIT}$

XPORT1 = 1
XPORT2 = 0
XDATA = x
XDIR = x
} Not Defined

† DPORT ONLY

low-power operating modes

The STANDBY and HALT low-power modes significantly reduce power consumption by reducing or stopping the activity of the various on-chip peripherals when processing is not required. Each of the low-power modes is entered by executing the IDLE instruction when the PWRDWN/IDLE bit in SCCR2 has been set to 1. The HALT/STANDBY bit in SCCR2 controls which low-power mode is entered.

In the STANDBY mode (HALT/STANDBY=0), all CPU activity and most peripheral module activity is stopped; however, the oscillator, internal clocks, Timer 1 and the receiver start bit detection circuit of the serial communications interface remain active. System processing is suspended until a qualified interrupt (hardware $\overline{\text{RESET}}$, external interrupts, Timer 1 interrupt, or a low level on the receive pin of the serial communications interface) is detected.

In the HALT mode (HALT/STANDBY=1), the TMS370Cx5x is placed in its lowest power consumption mode. The oscillator and internal clocks are stopped, causing all internal activity to be halted. System activity is suspended until a qualified interrupt (hardware $\overline{\text{RESET}}$, external interrupt, or low level on the receive pin of the serial communications interface) is detected.

POWERDOWN CONTROL BITS		MODE SELECTED
PWRDWN/IDLE (SCCR2.6)	HALT/STANDBY (SCCR2.7)	
1	0	Standby
1	1	Halt

The following information is preserved throughout both the STANDBY and HALT modes: RAM (register file), CPU registers (stack pointer, program counter, and status register), I/O pin direction and output data, and status registers of all on-chip peripheral functions. All CPU instruction processing is stopped during the STANDBY and HALT modes, and clocking of the watchdog timer is inhibited.

programmable timers

The two programmable timer modules of the TMS370Cx5x provide the designer with the enhanced timer resources required to perform realtime system control. The Timer 1 module contains the general-purpose timer T1 and the Watchdog timer (WD). The three independent 16-bit timers, T1, T2, and WD, allow program selection of input clock sources (realtime, external event, or pulse accumulate) with multiple 16-bit registers (input capture and compare) for special timer function control. These timers provide the capabilities for:

System Requirements	Timer Resource
Realtime System Control	Interval Timers with Interrupts
Input Pulse-Width Measurement	Pulse-Accumulate or Input-Capture Functions
External Event Synchronization	Event Counter Function
Timer Output Control	Compare Function
Pulse-Width Modulated Output Control	PWM Output Function
System Integrity	Watchdog Function

timer 1 module

The timer 1 module consists of three major blocks:

1. Prescaler/Clock Source, which determines the independent clock sources for the general purpose timer and the watchdog timer.
2. 16-bit General Purpose Timer, T1, which provides the event count, input capture, and compare functions.
3. 16-bit Watchdog Timer, which may be software programmed as an event counter/pulse accumulator if the watchdog function is not desired.

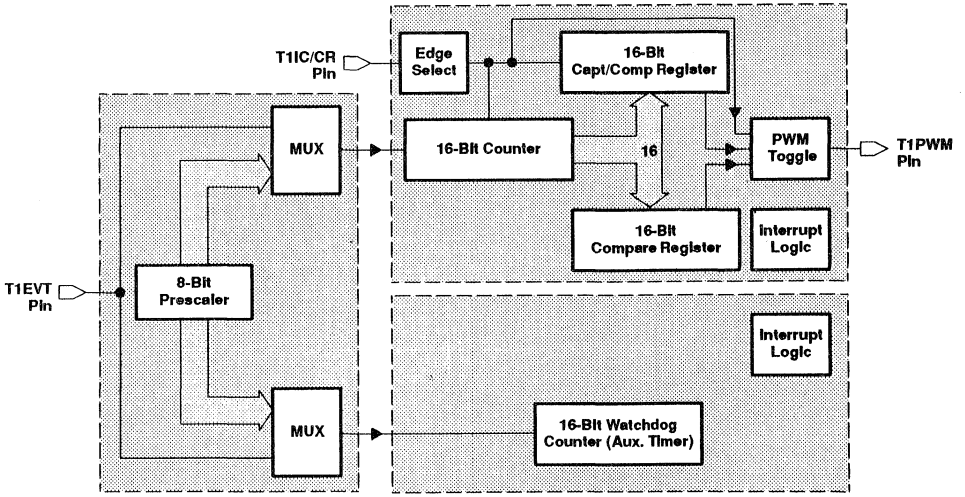


Figure 7. Timer 1 Module Block Diagram

timer 1 module prescaler/clock source

The clock source inputs for the general purpose timer and the watchdog timer are independently configured by the T1 and WD INPUT SELECT control bits of the T1CTL1 control register. The WD INPUT SELECT control bits cannot be changed after entering the watchdog mode (WD RST ENA = 1). Eight possible clock sources are programmable for each counter.

T1 INPUT			CLOCK SOURCE	WD INPUT		
SELECT 2	SELECT 1	SELECT 0		SELECT 2	SELECT 1	SELECT 0
0	0	0	System Clock	0	0	0
0	0	1	Pulse Accumulate	0	0	1
0	1	0	Event Input	0	1	0
0	1	1	No Clock Input	0	1	1
1	0	0	System Clock/4	1	0	0
1	0	1	System Clock/16	1	0	1
1	1	0	System Clock/64	1	1	0
1	1	1	System Clock/256	1	1	1

For realtime control applications, both the general-purpose timer and the watchdog timer are independently programmable from 16 to 24 bits in length. The 24-bit prescaler/timer generates overflow rates ranging from

13.1 ms with 200 ns timer resolution to 3.35 seconds with 51.2 µs timer resolution (external clock = 20 MHz).

In the **event counter mode**, an external high-to-low transition on the T1EVT pin is used to provide the clock for the internal timers. As shown in Figure 8, the T1EVT input provides the timer clock and is not routed through the prescaler. The T1EVT external clock frequency may not exceed the system clock frequency divided by 2. The general-purpose timer and the watchdog timer are programmable as 16-bit event counters.

In the **pulse accumulate mode**, an external input on the T1EVT pin is used to gate the internal system clock to the internal timers. While T1EVT input is logic 1, the timers will be clocked at the system clock rate and will accumulate system clock pulses until the T1EVT pin returns to logic 0. Both the general purpose timer and the watchdog timer are programmable as 16-bit pulse accumulators.

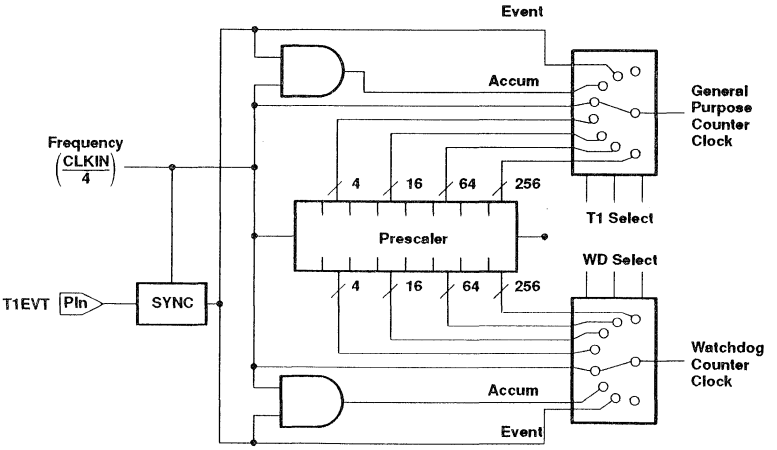


Figure 8. Timer 1 Counter Prescaler

timer 1 general purpose timer

The 16-bit general purpose timer, T1, is composed of a 16-bit resettable counter, a 16-bit compare register and associated compare logic, and a 16-bit register that functions as a capture register in one mode and a compare register in the other mode. The T1 MODE bit selects whether T1 operates in the Capture/Compare mode or the Dual Compare mode.

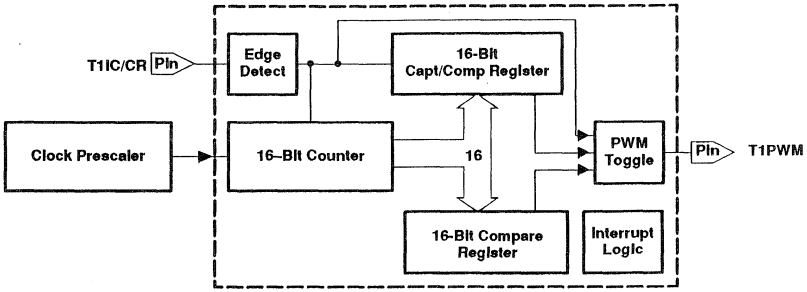


Figure 9. Timer 1 — General Purpose Timer

The counter is a free-running, 16-bit up-counter, clocked by the output of the Prescaler/Clock source. During initialization the counter is loaded with 0000h and begins its up-count. If the counter is not reset before reaching FFFFh, the counter will roll over to 0000h and continue counting. Upon counter roll-over, the T1 OVRFL INT FLAG is set to 1, and a timer interrupt is generated if the T1 OVRFL INT ENA bit is set to 1.

The counter may be reset to 0000h during counting by either, 1) writing a 1 to the T1 SW RESET bit, 2) a compare equal condition from the dedicated T1 compare function, or 3) an external pulse on the T1 IC/CR pin (Dual Compare mode). The designer may select via software (T1EDGE POLARITY bit) which external transition, low-to-high or high-to-low, on the T1IC/CR pin will cause the counter to be reset.

Special circuitry prevents the 16-bit registers, including the Counter, Compare, or Capture registers, from changing in the middle of a 16-bit read or write operation. When reading a 16-bit register, read the LSB first, then read the MSB. When writing to a 16-bit register, write the MSB first, then write the LSB. Take care to prevent accesses to another 16-bit register within this module during a 16-bit read or write. Such accesses might occur during the interrupt routines.

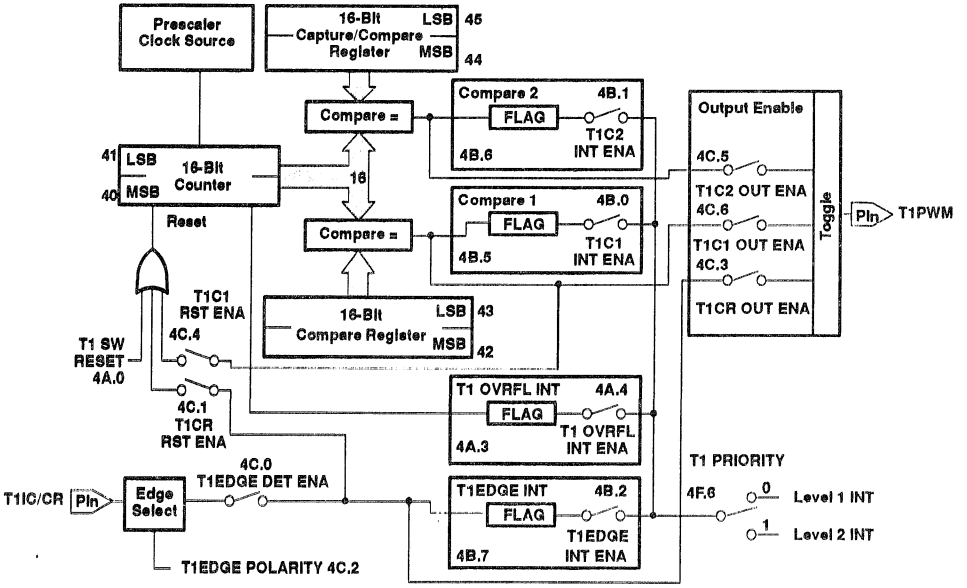
The timer 1 module has three I/O pins used for the functions as shown in the following table. Any of these three pins not used in a timer application may be individually configured as general purpose digital I/O pins by the timer 1 module port control registers (T1PC1 and T1PC2).

Timer 1 Module I/O Pin Functions

PIN	DUAL COMPARE MODE	CAPTURE/COMPARE MODE
T1IC/CR	Counter Reset input	Input Capture input
T1PWM	PWM output	Compare output
T1EVT	External Event input or Pulse Accumulate input	External Event input or Pulse Accumulate input

The **Dual Compare mode** (T1 MODE = 0) provides two compare registers, an external resettable counter, and a timer output pin. These allow the timer to act as an interval timer, a PWM output, simple output toggle, or many other timer functions. The Dual Compare mode as shown in Figure 10 continuously compares the contents of the two compare registers to the current value of the 16-bit counter. If a timer compare register equals the counter, the circuit sets the associated interrupt flag to 1 and toggles the T1PWM output pin if enabled, and/or generates a Timer 1 interrupt. An output compare equal condition from the dedicated compare register can also initiate a counter reset. A programmable interval timer function, selected by using the compare equal condition to generate a system interrupt and the counter reset function, generates a periodic interrupt.

Either compare function may be used to toggle the T1PWM output pin when a timer compare equal occurs, while the other compare function may be used for another system timing function. Using both compare functions to control the T1PWM pin allows direct PWM generation with minimal CPU software overhead. In typical PWM applications, the compare register is written with the periodic interval and is configured to allow counter reset on compare equal, and the Capture/Compare register is written with the pulse width to be generated within that interval. The program pulse width may be changed by the application program during the timer operation to alter the PWM output. For high-speed control applications, a minimum pulse width of 200 ns and a period as low as 400 ns can be maintained when using a clock of 20 MHz.

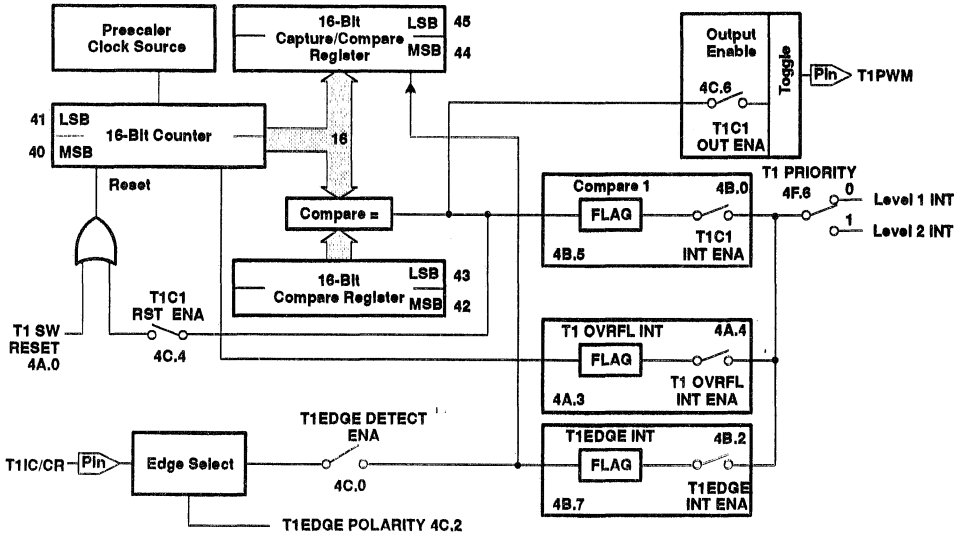


NOTE 2: The numbers on the diagram such as 4C.0, identify the register and the bit in the peripheral frame. For example, the actual address of 4C.0 is 104Ch, bit 0, in the T1CTL4 register.

Figure 10. Timer 1 — Dual Compare Mode

In addition, a PWM output that is initiated by a transition on an external pin is provided by the timer hardware to support time-critical control applications. Typically, in these applications an external input (T1IC/CR) is used to reset the counter, generate a timer interrupt, and toggle the T1 PWM pin to start the PWM output. The compare function will then toggle the output after the programmed pulse width has elapsed. The input edge detect function is enabled under program control by the T1CR DET ENA bit, and upon the next occurrence of the selected edge transition, the T1EDGE INT FLAG bit is set to 1, a timer interrupt is generated (if T1EDGE INT ENA = 1), and the T1PWM output pin is toggled (if T1CR OUT ENA = 1). Selection of the active input transition is under control of T1EDGE POLARITY. In the Dual Compare mode, the edge detect function must be re-enabled after each
valid edge detect.

In the **Capture/Compare mode** (T1 MODE = 1), T1 is configured to provide one input capture register for external timing and pulse width measurement, and one compare register for use as a programmable interval timer. The compare register in this mode functions the same as in the Dual Compare mode described above, including the ability to toggle the PWM pin. The capture/compare register functions in this mode as a 16-bit input capture register, as shown in Figure 11. On the occurrence of a valid input on the T1IC/CR pin, the current counter value is loaded into the 16-bit input capture register, the T1EDGE INT FLAG is set to 1, and a timer interrupt is generated (if T1EDGE INT ENA = 1). The input detect function is enabled by the T1EDGE DET ENA bit, with T1EDGE POLARITY selecting the active input transition. In the Capture/Compare mode, the edge detect function, once enabled, remains enabled following a valid edge detect.



NOTE 2: The numbers on the diagram, such as 4C.0, identify the register and the bit in the peripheral frame. For example, the actual address of 4C.0 is 104Ch, bit 0, in the T1CTL4 register.

Figure 11. Timer 1 — Capture/Compare Mode

timer 1 module watchdog timer

The watchdog timer, contained in the timer 1 module, is a free-running 16-bit resettable up-counter clocked by the output of the Prescaler/Clock Source. The timer is software configured as either a watchdog timer to protect against system software failures and errors, or as a general purpose timer if the watchdog function is not desired. The 16-bit up-counter is programmable (via the WD OVRFL TAP SEL bit) to set the initial count at either 0000h or 8000h. The current value of the watchdog timer may be read at any time during its operation.

In the **watchdog mode** (WD OVRFL RST ENA = 1), the timer will generate a system reset if the timer is re-initialized by an incorrect value or if the counter overflows. The required re-initialization frequency is determined by the system clock frequency, the prescaler/clock source selected, and whether the WD OVRFL TAP SEL bit is set for a 15 or 16 bit counter rollover. With a clock = 20 MHz, the watchdog timer overflow rates range from 6.55 ms to 3.35 seconds. These values are selected prior to entering the watchdog mode because once the software enables the watchdog reset function (WD OVRFL RST ENA set to 1), subsequent writes to these control bits are ignored. Writes to these watchdog control bits can occur only following a powerup reset, which enhances watchdog timer system software integrity.

The watchdog timer is re-initialized by writing a predefined value to the watchdog reset key (WDRST) located in the peripheral file. The proper reset key alternates between 55h and AAH, beginning with 55h following the enable of the watchdog reset function. Writes of the correct value must occur prior to the timer overflow period. A write of any value other than the correct predefined value to the watchdog reset key will be interpreted as a lost program and a system reset will be initiated. A watchdog timer overflow or incorrect reset key will set the WD OVRFL INT FLAG bit to 1 and may be interrogated by the program following system reset to determine the source of the reset.

In the **non-watchdog mode** (WD OVRFL RST ENA = 0), the watchdog timer may be used as an event counter, pulse accumulator, or as an interval timer. In this mode, the system reset function is disabled. The watchdog counter is re-initialized by writing any value to the watchdog reset key (WDRST). When used as an interval timer, the timer overflow interval is determined by the system clock frequency, the prescaler/clock source value selected, and the value of the WD OVRFL TAP SEL bit. If the WD counter is not reset before overflowing, the counter will roll over to either 0000h or 8000h, as determined by the WD OVRFL TAP SEL bit, and continue counting. Upon counter overflow, the WD OVRFL INT FLAG is set to 1 and a timer interrupt is generated if the WD OVRFL INT ENA bit is set to 1. Alternately, an external input on the T1 EVT pin may be used with the watchdog timer to provide an additional 16-bit event counter or pulse accumulator.

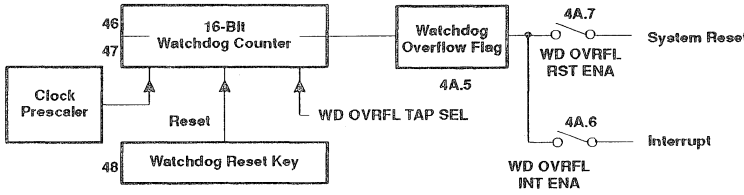


Figure 12. Watchdog/General Purpose Timer

Peripheral File Frame 4: Timer 1 Module Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1040h	P040	Bit 15			Counter MSB				Bit 8	T1CNTR
1041h	P041	Bit 7			Counter LSB				Bit 0	
1042h	P042	Bit 15			Compare Register MSB				Bit 8	T1C
1043h	P043	Bit 7			Compare Register LSB				Bit 0	
1044h	P044	Bit 15			Capture/Compare Register MSB				Bit 8	T1CC
1045h	P045	Bit 7			Capture/Compare Register LSB				Bit 0	
1046h	P046	Bit 15			Watchdog Counter MSB				Bit 8	WDCNTR
1047h	P047	Bit 7			Watchdog Counter LSB				Bit 0	
1048h	P048	Watchdog Reset Key								WDRST
1049h	P049	WD OVRFL TAP SEL†	WD INPUT SELECT2†	WD INPUT SELECT1†	WD INPUT SELECT0†	—	T1INPUT SELECT2	T1INPUT SELECT1	T1INPUT SELECT0	T1CTL1
104Ah	P04A	WD OVRFL RST ENA†	WD OVRFL INT ENA	WD OVRFL INT FLAG	TI OVRFL INT ENA	T1 OVRFL INT FLAG	—	—	T1 SW RESET	T1CTL2
Mode: Dual Compare										
104Bh	P04B	T1EDGE INT FLAG	T1C2 INT FLAG	T1C1 INT FLAG	—	—	T1EDGE INT ENA	T1C2 INT ENA	T1C1 INT ENA	T1CTL3
104Ch	P04C	T1 MODE = 0	T1C1 OUT ENA	T1C2 OUT ENA	T1C1 RST ENA	T1CR OUT ENA	T1EDGE POLARITY	T1CR RST ENA	T1EDGE DET ENA	T1CTL4
Mode: Capture/Compare										
104Bh	P04B	T1EDGE INT FLAG	—	T1C1 INT FLAG	—	—	T1EDGE INT ENA	—	T1C1 INT ENA	T1CTL3
104Ch	P04C	T1 MODE = 1	T1C1 OUT ENA	—	T1C1 RST ENA	—	T1EDGE POLARITY	—	T1EDGE DET ENA	T1CTL4
104Dh	P04D	—	—	—	—	T1EVT DATA IN	T1EVT DATA OUT	T1EVT FUNCTION	T1EVT DATA DIR	T1PC1
104Eh	P04E	T1PWM DATA IN	T1PWM DATA OUT	T1PWM FUNCTION	T1PWM DATA DIR	T1IC/CR DATA IN	T1IC/CR DATA OUT	T1IC/CR FUNCTION	T1IC/CR DATA DIR	T1PC2
104Fh	P04F	**T1 STEST**	**T1 PRIORITY**	—	—	—	—	—	—	T1PRI

† Once the WD OVRFL RST ENA bit is set, these bits cannot be changed until after a full power-down cycle has been completed.

The formulas in Figure 13 show the calculations for the resulting time, given values in the compare registers T1C and T1CC.

$$\text{time} = \left(\frac{4}{\text{CLKIN}}\right) (\text{prescale}) (\text{compare} + 1)$$

or

$$\text{time} = t_c (\text{prescale}) (\text{compare} + 1)$$

Figure 13. Timer 1 Compare Register Formulas

timer 2 module

Timer 2 consists of a clock source block and a 16-bit general purpose timer that provides the event count, input capture, and compare functions.

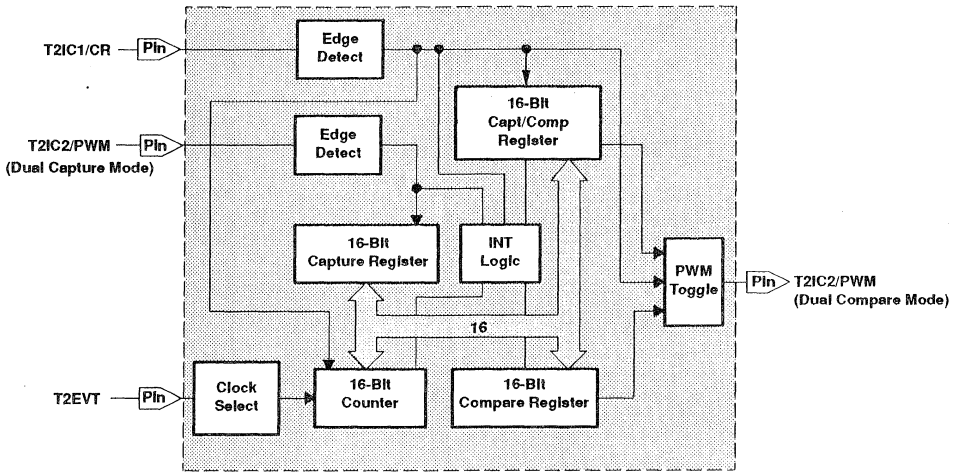


Figure 14. Timer 2 Module Block Diagram

timer 2 clock source

The clock source input for the general purpose timer is configured by the T2 INPUT SELECT control bits of the T2CTL1 control register. The four programmable clock sources for the general purpose counter are system clock, pulse accumulate, event input, or no clock input (counter stopped). When using the system clock input, the 16-bit timer generates an overflow rate of 13.1 ms with 200 ns resolution (clock = 20 MHz).

In the **event counter mode**, the general purpose timer is programmable as a 16-bit event counter. An external low-to-high transition on the T2EVT pin is used to provide the clock for the internal timer. The T2EVT external clock frequency may not exceed the system clock frequency divided by 2.

In the **pulse accumulate mode**, the general purpose timer is programmable as a 16-bit pulse accumulator. An external input on the T2EVT pin is used to gate the internal system clock to the internal timers. While T2EVT input is logic 1, the timers will be clocked at the system clock rate and will count system clock pulses until the T2EVT pin returns to logic zero.

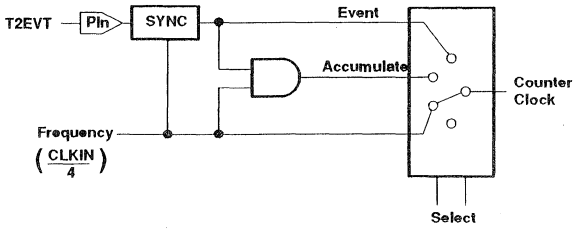


Figure 15. Timer 2 Clock Select

timer 2 general purpose timer

The 16-bit general purpose timer, T2 , is composed of a 16-bit resettable counter, 16-bit compare register with associated compare logic, a 16-bit capture register, and a 16-bit register that functions as a capture register in one mode and a compare register in the other mode. The T2 MODE bit selects whether T2 operates in the Dual Compare mode or the Dual Capture mode.

The counter is a free-running 16-bit up-counter, clocked by the system clock, external event, or system clock while external event active (pulse accumulate). During initialization, the counter is loaded with 0000h and begins its up-count. If the counter is not reset before reaching FFFFh, the counter will roll over to 0000h and continue counting. Upon counter roll-over, the T2 OVRFLINT FLAG is set to 1, and a timer interrupt is generated if the T2 OVRFL INT ENA bit is set to 1.

The counter may be reset to 0000h during counting by either; 1) writing a 1 to the T2 SW RESET bit, 2) a compare equal condition from the dedicated T2 compare function, or 3) an external pulse on the T2IC1/CR pin (Dual Compare mode). The designer may select via software (T2CR POLARITY bit) which external transition, low-to-high or high-to-low, on the T2IC1/CR pin will cause the counter to be reset.

Special circuitry prevents the 16-bit registers, including the Counter, Compare, or Capture registers, from changing in the middle of a 16-bit read or write operation. When reading a 16-bit register, read the LSB first and then read the MSB. When writing to a 16-bit register, write the MSB first and then write the LSB. The register value will not change between reading or writing the bytes when done in this order.

Timer 2 has three I/O pins used for functions as shown in the table below. Any of these three pins not used in a timer application may be individually configured as general purpose digital I/O pins by the Timer 2 port control registers (T2PC1 and T2PC2).

Timer 2 I/0 Pin Functions

PIN	DUAL COMPARE MODE	DUAL CAPTURE MODE
T2IC1/CR	Counter reset input.	Input Capture 1 input.
T2IC2/PWM	PWM output.	Input Capture 2 input.
T2EVT	External event input or pulse accumulate input.	External event input or pulse accumulate input.

The **Dual Compare mode** (T2 MODE = 0) provides two compare registers, an external resettable counter, and a timer output pin. These allow the timer to act as a interval timer, a PWM output, simple output toggle, or many other timer functions. In this mode, the capture/compare register functions as a 16-bit read/write compare register, as shown in Figure 16. The operation of T2 is identical to T1 while operating in the dual compare mode.

In the **Dual Capture mode** (T2 MODE = 1), T2 is configured to provide one compare register for use as a programmable interval timer, and two input capture registers for external input timing and pulse width measurement. In this mode the capture/compare register functions as a 16-bit input capture register, as shown in Figure 17. Each capture input pin (T2IC1/CR and T2IC2/PWM) has an input edge detect function enabled by the associated DET ENA control bit, with the associated POLARITY bit selecting the active input transition. On the occurrence of a valid input on the T2IC1/CR or T2IC2/PWM pin, the current counter value is loaded into the 16-bit capture/compare and 16-bit input capture register, respectively. In addition, the respective input capture INT FLAG is set to 1 and a timer interrupt is generated if the respective INT ENA is set to 1.

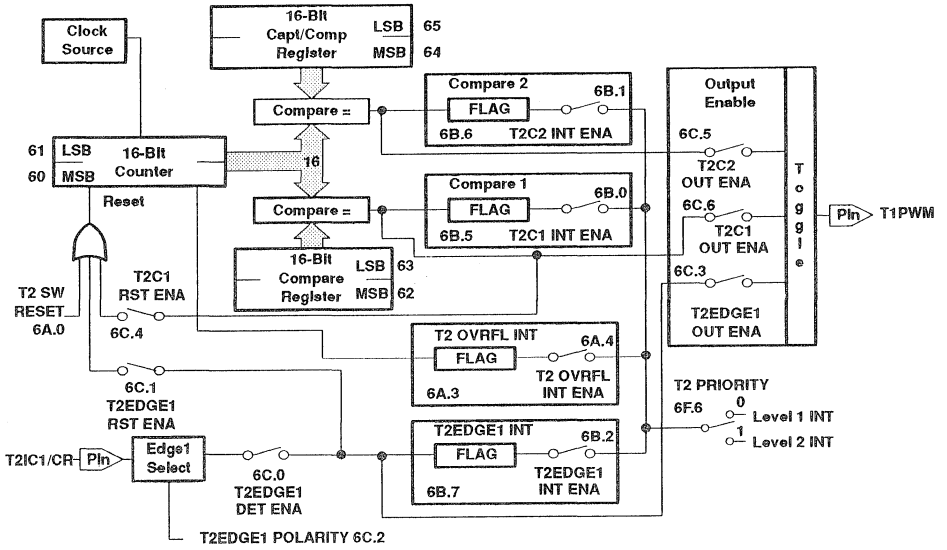


Figure 16. Timer 2— Dual Compare Mode

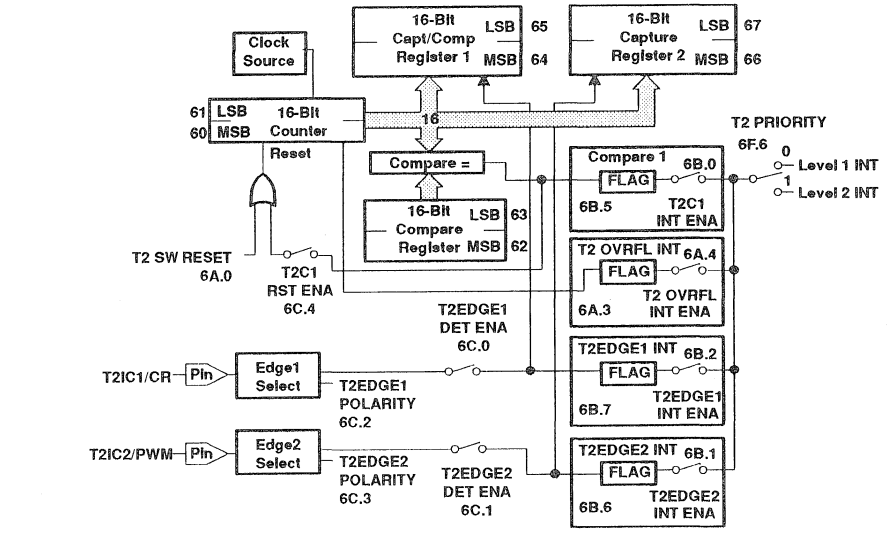


Figure 17. Timer 2— Dual Capture Mode

Peripheral File Frame 6: Timer 2 Module Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1060h	P060	Bit 15			T2 Counter MSB				Bit 8	T2CNTR
1061h	P061	Bit 7			T2 Counter LSB				Bit 0	
1062h	P062	Bit 15			T2 Compare 1 Register MSB				Bit 8	T2C
1063h	P063	Bit 7			T2 Compare 1 Register LSB				Bit 0	
1064h	P064	Bit 15			T2 Capture 1/Compare 2 Register MSB				Bit 8	T2CC
1065h	P065	Bit 7			T2 Capture 1/Compare 2 Register LSB				Bit 0	
1066h	P066	Bit 15			T2 Capture Register 2 MSB				Bit 8	T2IC
1067h	P067	Bit 7			T2 Capture Register 2 LSB				Bit 0	
1068h 1069h	P068 P069	Reserved								
106Ah	P06A	—	—	—	T2OVRFL INT ENA	T2OVRFL INT FLAG	T2INPUT SELECT1	T2INPUT SELECT0	T2 SW RESET	T2CTL1
Mode: Dual Compare										
106Bh	P06B	T2EDGE1 INT FLAG	T2C2 INT FLAG	T2C1 INT FLAG	—	—	T2EDGE1 INT ENA	T2C2 INT ENA	T2C1 INT ENA	T2CTL2
106Ch	P06C	T2 MODE = 0	T2C1 OUT ENA	T2C2 OUT ENA	T2C1 RST ENA	T2EDGE1 OUT ENA	T2EDGE1 POLARITY	T2EDGE1 RST ENA	T2EDGE1 DET ENA	T2CTL3
Mode: Dual Capture										
106Bh	P06B	T2EDGE1 INT FLAG	T2EDGE2 INT FLAG	T2C1 INT FLAG	—	—	T2EDGE1 INT ENA	T2EDGE2 INT ENA	T2C1 INT ENA	T2CTL2
106Ch	P06C	T2 MODE = 1	—	—	T2C1 RST ENA	T2EDGE2 POLARITY	T2EDGE1 POLARITY	T2EDGE2 DET ENA	T2EDGE1 DET ENA	T2CTL3
106Dh	P06D	—	—	—	—	T2EVT DATA IN	T2EVT DATA OUT	T2EVT FUNCTION	T2EVT DATA DIR	T2PC1
106Eh	P06E	T2IC2/PWM DATA IN	T2IC2/PWM DATA OUT	T2IC2/PWM FUNCTION	T2IC2/PWM DATA DIR	T2IC1/CR DATA IN	T2IC1/CR DATA OUT	T2IC1/CR FUNCTION	T2IC1/CR DATA DIR	T2PC2
106Fh	P06F	**T2 STEST**	**T2 PRIORITY**	—	—	—	—	—	—	T2PRI

The formulas in Figure 18 show the calculations for the resulting time, given values in the compare registers T2C and T2CC.

$$\text{time} = \left(\frac{4}{\text{CLKIN}}\right) (\text{prescale}) (\text{compare} + 1)$$

or

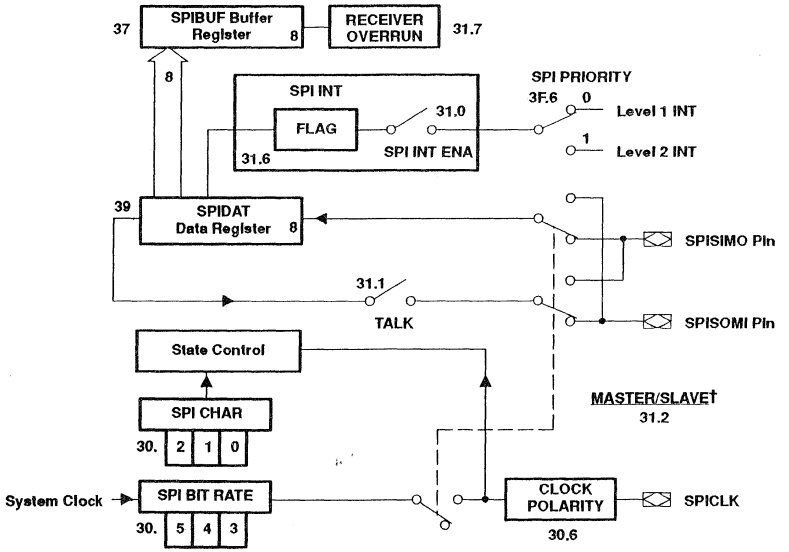
$$\text{time} = t_c (\text{prescale}) (\text{compare} + 1)$$

Figure 18. Timer 2 Compare Register Formulas

serial peripheral interface (SPI)

The Serial Peripheral Interface (SPI) is a high-speed synchronous serial I/O port that allows a serial bit stream of programmed length (one to eight data bits) to be shifted into or out of the device at a programmed bit transfer rate. The SPI is normally used for communications between the microcontroller and external peripherals or another microcontroller. Typical applications include external I/O or peripheral expansion using devices such as shift registers, display drivers, A/D converters, etc. Multiprocessor communications are also supported by the master/slave operation of the SPI.

Three I/O pins are associated with the SPI. These include the SPI slave-in master-out (SPISIMO), SPI slave-out master-in (SPISOMI), and SPI serial clock (SPICLK). These I/O pins can be configured for three-wire full-duplex transmit/receive or two-wire receive or transmit only. Any of these three pins not used in an SPI application may be individually configured as general purpose digital I/O pins controlled by SPIPC1 and SPIPC2.



† Diagram shown in SLAVE mode.

Figure 19. SPI Block Diagram†

A variety of multiprocessor configurations can be supported, ranging from single master with multiple slaves to multi-master systems. General purpose I/O pins can be used to implement the slave enables and multi-master hardware handshakes between microcontrollers in the network.

The MASTER/SLAVE bit of the SPICTL control register determines if the SPI operates in the master or slave mode. Master or slave data transmission can be disabled by writing a zero to the TALK bit of the SPICTL control register, forming a two-wire receive-only network (SPICLK and data in).

In the **master mode** (MASTER/SLAVE = 1), the SPI provides the serial clock on the SPICLK pin for the entire serial communications network. The SPICCR register (SPI BIT RATE2, RATE1, RATE0) determines the bit transfer rate for the network, both transmit and receive. For any specific system clock frequency, there are eight data transfer rates that can be selected by these control bits. The data transfer rate is defined by selecting a one-of-eight divide-by of the system clock frequency (divide-by-2, -4, -8, -16, -32, -64, -128, and -256).

$$\text{SPI Baud Rate} = \frac{\text{CLKIN}}{8 \times 2^b}$$

where b = bit rate in SPICCR bit 3, 4, 5 (range 0 – 7).

Data written to the SPIDAT register initiates data transmission on the SPISIMO pin, MSB of data transmitted first.Simultaneously, received data is shifted in through the SPISOMI pin into the SPIDAT register, and upon completion of transmitting the selected number of bits, the data is transferred to the SPIBUF (double buffered receiver) for reading by the CPU to permit new transactions to take place. Data is shifted into the SPI (the most significant bit first), there it is stored right-justified in the SPIBUF. To receive a character when operating as a master, data must be written to the SPIDAT to initiate the transaction. When the specified number of data bits have been shifted into or out of the SPIDAT register, the SPI INT FLAG bit is set and if the SPI INT ENA bit is set to one, an Interrupt is asserted.

In the **slave mode** (MASTER/SLAVE = 0), data shifts out on the SPISOMI pin and in on the SPISIMO pin. The SPICLK pin is used as the input for the serial shift clock, which is supplied from the external network master. The transfer rate is defined by the input clock on the SPICLK pin, which is supplied from the network master. The SPICLK input frequency should be no greater than system clock frequency divided by eight.

Data written to the SPIDAT register will be transmitted to the network when the SPICLK is received from the network master. To receive data, the SPI waits for the network master to send SPICLK and then shifts the data on the SPISIMO pin into the SPIDAT register. If data is to be transferred by the slave simultaneously, it must be written to the SPIDAT register prior to the beginning of SPICLK.

Compatibility with the broadest range of existing peripheral devices is provided by the SPI through its software programmable transmit/receive character length, bit transfer rate, and clock polarity. A character length from one to eight data bits is selected by writing to the SPICCR control register (SPI CHAR2, CHAR1, and CHAR0) to specifically match the peripheral's data length requirements, thereby not requiring the overhead of data bit padding during communications. Applications requiring more than eight bits of serial data use multiple back-to-back SPI operations.

External peripherals enable output data on either the rising or the falling edge of the serial clock, while latching incoming data on the opposite edge. The SPI supports data transfer using either of these approaches. The CLOCK POLARITY bit controls the steady-state or at-rest condition of the SPICLK signal. This bit affects both master and slave modes of operation. When CLOCK POLARITY is set to 1, the at-rest level of SPICLK is a logic 1 . Data is enabled at the output on the falling edge of SPICLK, and data is latched by the network master and slaves on the rising edge of SPICLK. When CLOCK POLARITY is set to 0, the at-rest level of SPICLK is a logic 0. Data is enabled for output on the rising edge of SPICLK, and data is latched by the network master on the falling edge of SPICLK.

Peripheral File Frame 3: Serial Peripheral Interface (SPI) Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1030h	P030	SPI SW RESET	CLOCK POLARITY	SPI BIT RATE2	SPI BIT RATE1	SPI BIT RATE0	SPI CHAR2	SPI CHAR1	SPI CHAR0	SPICCR
1031h	P031	RECEIVER OVERRUN	SPI INT FLAG	—	—	—	MASTER/ SLAVE	TALK	SPI INT ENA	SPICTL
1032h to 1036h	P032 to P036	Reserved								
1037h	P037	SPI Receive Data Buffer Register								SPIBUF
1038h	P038	Reserved								
1039h	P039	SPI Serial Data Register								SPIDAT
103Ah to 103Ch	P03A to P03C	Reserved								
103Dh	P03D	—	—	—	—	SPICLK DATA IN	SPICLK DATA OUT	SPICLK FUNC-TION	SPICLK DATA DIR	SPIPC1
103Eh	P03E	SPISIMO DATA IN	SPISIMO DATA OUT	SPISIMO FUNC-TION	SPISIMO DATA DIR	SPISOMI DATA IN	SPISOMI DATA OUT	SPISOMI FUNC-TION	SPISOMI DATA DIR	SPIPC2
103Fh	P03F	**SPI STEST**	**SPI PRIORITY**	**SPI ESPEN**	—	—	—	—	—	SPIPRI

serial communications interface (SCI)

The Serial communications Interface (SCI) is a full-duplex serial I/O port that supports standard NRZ serial communications in a programmed data format (start bit, 1 to 8 data bits, parity even/odd/off, one or two stop bits) at a variety of programmable baud rates. High-speed isosynchronous communications, as well as standard asynchronous communications, are supported for interfacing to peripheral devices. The isosynchronous communications mode combines features of the asynchronous mode with a synchronizing clock signal. The isosynchronous mode has the same format as the asynchronous mode using start, stop, parity, and data bits, but it it uses one serial clock cycle per bit to achieve a much higher transmission speed. Multiprocessor communications using idle line wake-up and address bit wake-up protocols are also supported by the SCI transmit and receive hardware.

As shown in Figure 20, the SCI receiver and transmitter are double buffered to reduce the possibility of overwriting data prior to the previous data being read or transmitted from the SCI.

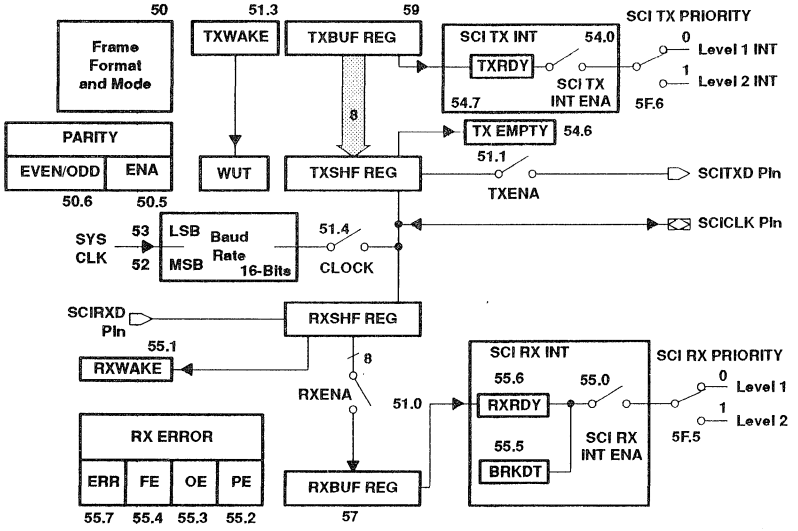


Figure 20. SCI Block Diagram

The SCI provides independent interrupt requests and vectors for the receiver and transmitter. Interrupts requested by the SCI receiver and SCI transmitter can be software programmed onto different priority levels by the SCI RX PRIORITY and SCI TX PRIORITY control bits. When SCI interrupt requests are made on the same level, the receiver always has higher priority than the transmitter to reduce the possibility of receiver overrun. An SCI TXINT interrupt is asserted whenever TXBUF is transferred to TXSHF. An SCI RXINT interrupt is asserted whenever the SCI receives a complete frame (RXSHF transfers to RXBUF) or when a break detect condition occurs (SCIRXD is low for 10 bit periods following a stop bit).

If the TMS370Cx5x has been placed in HALT or STANDBY low-power mode with the SCI RX INT ENA bit = 1, the detection of the start bit (one-to-zero transition) by the SCI receiver initiates receipt of the SCI input, exits the low power mode, activates the microcontroller (CPU, clocks, on-chip peripherals), and initiates execution of the SCI RXINT interrupt service routine. To ensure valid data receipt of the first frame, the baud rate must be slow enough for the SCI to sample for a valid start bit after exiting from the power down mode, or the first data byte must be ignored.

The SCI transmitter and receiver are functionally independent to support full-duplex communications; however, they use the same data format, baud rate, communications mode, and multiprocessor communications protocol. The SCICCR control register selects the transmit and receive data format. Figure 21 shows the SCI data format of one frame of information, which consists of an idle line (logic 1), one start bit (logic 0), one to eight data bits, an address bit (if in address bit wake-up mode), a parity bit (if enabled), and one or two stop bits (logic 1). The character length of one to eight data bits is selected by the SCI CHAR2, SCI CHAR1, and SCI CHAR0 control bits. Parity on/off is selected by PARITY ENABLE with the EVEN/ODD PARITY bit selecting the type. Parity generation and verification is performed in the SCI hardware, requiring no CPU calculation overhead. One or two stop bits for transmission are selected by the STOP BITS control bit. The receiver checks for one stop bit on incoming data.

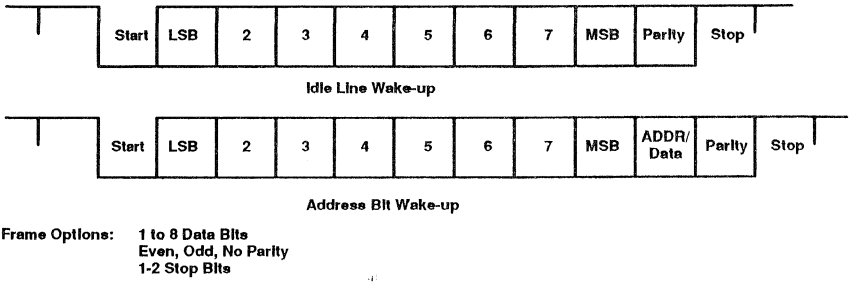


Figure 21. Frame Formats

The SCI communications mode is selected by the ASYNC/ISOSYNC control bit. The transmit and receive data format as described above are identical in both communication modes. In the **asynchronous mode** (ASYNC/ISOSYNC = 1), the external communications interface consists of the SCITXD and SCIRXD pins with an optional SCICLK input for driving the internal SCICLK. The transmit baud rate is 1/16 that of the SCICLK frequency. The receiver internally samples the data input at 16 times the bit rate. The receiver uses majority vote sampling on the seventh, eighth, and ninth SCICLK periods to determine the value of the start bit, data bits, parity, and first stop bit. Asynchronous data rates are supported up to 156K baud (SYSCLK/2^{21} to SYSCLK/32) at 20 MHz.

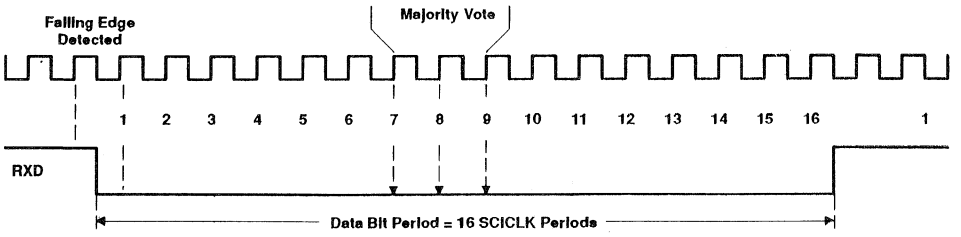


Figure 22. Asynchronous Mode

The **Isosynchronous mode** (ASYNC/ISOSYNC = 0) has the same format as the asynchronous mode, consisting of a start bit, one to eight data bits, an even/odd/no parity bit, and one or two stop bits, but uses an additional synchronizing clock to support high speed serial communications. The external system interface consists of the SCITXD and SCIRXD pins and a continuous synchronizing clock on on the SCICLK pin. Isosynchronous transmit and receive data is clocked at a rate equal to the SCICLK rate, and receiver values are read on a single sample basis. Isosynchronous data rates with synchronizing SCICLK are supported up to 2.5M baud (SYSCLK/2^{17} to SYSCLK/2) at 20 MHz.

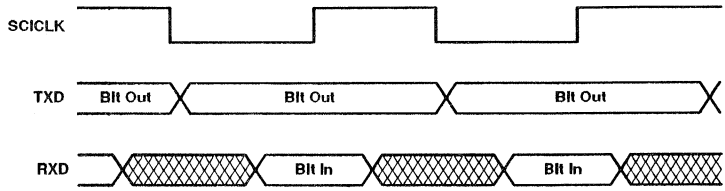


Figure 23. Isosynchronous Mode

The CLOCK bit in SCICTL determines whether the SCI clocking signal comes in from an external source through the SCICLK pin or goes out through SCICLK after generation in the integral baud rate timer. The isosynchronous mode baud rate equals the SCICLK rate; the asynchronous mode baud rate 1/16 the SCICLK rate. The maximum frequency of an external clock source can be no greater than 1/10 the system clock frequency. The frequency of the SCICLK when generated by the internal baud rate timer given by the formula.

$$\text{SCICLK} = \frac{\text{CLKIN}}{8(\text{Baud Rate Reg} + 1)}$$

The baud rate using the internal clock equals the SCICLK rate in the isosynchronous mode and equals 1/16 the SCICLK in the asynchronous mode. The 16-bit baud rate register allows the selection of many different standard baud rates.

$$\text{Asynchronous Baud Rate} = \frac{\text{CLKIN}}{128(\text{Baud Rate Reg} + 1)}$$

$$\text{Isosynchronous Baud Rate} = \frac{\text{CLKIN}}{8(\text{Baud Rate Reg} + 1)}$$

NOTE

When an external serial clock signal is used, the maximum SCICLK frequency is CLKIN/40.

In both asynchronous and isosynchronous modes, if the transmitter is enabled (TXENA = 1), SCI transmission is initiated following a CPU write to the TXBUF register. This sets TX EMPTY to 0; TXSHF is loaded from TXBUF, TXRDY flag is set to 1, and if SCI TX INT ENA is set to 1, SCI transmit interrupt (TXINT) will be asserted. Another write may then be performed to the TXBUF; if not, the transmitter idles (SCITXD outputs continuous high), and TX EMPTY is set to 1 (both TXBUF and TXSHF are empty) until the next write to TXBUF.

In both asynchronous and isosynchronous modes, when a frame is fully received, RXBUF is loaded from RXSHF, the error status bits are set accordingly, RXRDY flag is set to 1, and if SCI RX INT ENA is set to 1, an SCI receiver interrupt (RXINT) will be asserted. The SCI receiver performs extensive error checking during data bit reception and provides individual error flags for parity error (PE), overrun error (OE), framing error (FE), and break detect (BRKDT) for application program querying.

The SCI supports two multiprocessor communication formats to allow efficient transfer of information between many microcontrollers on the same serial data link. Information is typically transferred as a block of data from a source to a destination, with the destination address identified at the beginning of the block. The SCI has the ability to inhibit all SCI receiver flags and interrupts until a start of a block of data (a destination address) is identified. When a block start is identified, the SCI initiates the following sequence for both multiprocessor communication formats:

1. The serial port wakes up at the start of the block and receives the first frame (containing the destination address).
2. A software routine responds to the SCI receiver interrupt and checks the incoming byte against its address byte stored in memory.
3. If the block is addressed to the microcontroller, the SCI remains active and the CPU reads the rest of the block. If the address does not compare, the software routine puts the serial port to sleep and the SCI will inhibit all SCI receiver flags and interrupts until the next block start.

To provide system flexibility, the SCI, in both asynchronous and isosynchronous modes, recognizes the idle line wake-up and address bit wake-up multiprocessor protocols. The multiprocessor protocol is selected by the ADDRESS/IDLE WUP control bit in the SCICCR register. Both protocols use the SLEEP and TXWAKE bits to control the receive and transmit features of the wake-up mode, and the RXWAKE status bit to provide the receiver wake-up condition.

In **idle line wake-up**, blocks are separated by having a longer idle time (logic one) between the blocks than between frames within the blocks. As shown in Figure 24, an idle time of 10 or more bits after a frame indicates a start of a new block and wakes up all receivers. Under software control, all receivers that do not recognize the address in the first frame of the message ignore the rest of the message and await the next idle line. The SCI transmitter allows an idle time of exactly one frame to be transmitted to indicate the start of the next block to maintain serial data link efficiency by minimizing the idle time between block starts. Idle line wake-up protocol has no overhead within the message frames and is typically used when transferring large blocks of data.

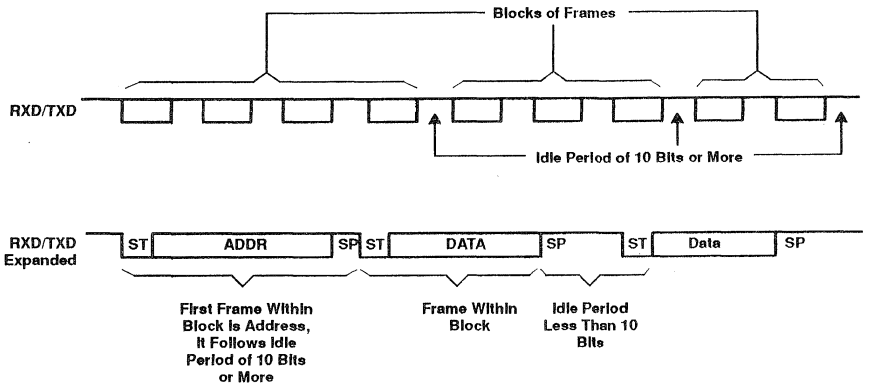


Figure 24. Idle Line Multiprocessor Mode

In **address bit wake-up**, each frame has an extra bit, the ADDR/DATA bit, positioned just before the parity bit (if used). As shown in Figure 25, block starts are distinguished by the ADDR/DATA bit set to 1 in the first frame of the block and all subsequent frames of the block have the ADDR/DATA bit set to 0. The start of the next block is identified by the next frame that has a 1 in ADDR/DATA. The idle line time is irrelevant in this protocol. All receivers wake up upon receiving a frame with ADDR/DATA set to 1. Under software control, all receivers that do not recognize their address in the first frame of the message ignore the rest of the message and await the next active ADDR/DATA bit. Address bit wake-up protocol eliminates interblock gaps and is efficient in transferring many small blocks of data.

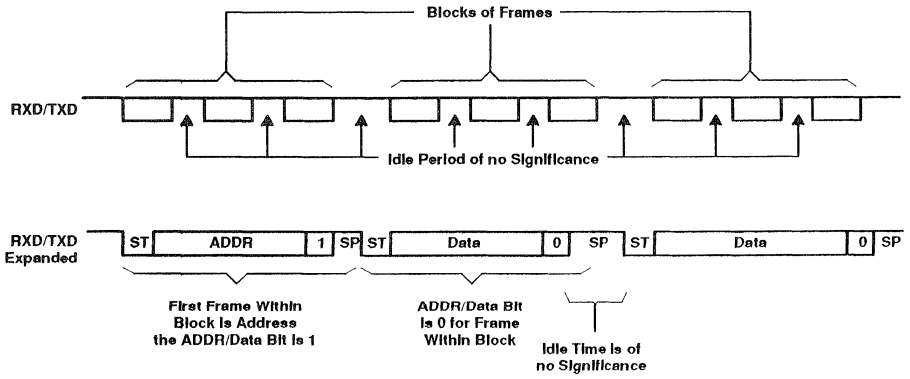


Figure 25. Address Bit Multiprocessor Mode

Peripheral File Frame 5: Serial Communication Interface (SCI) Control Registers

ADDR	PF	7	6	5	4	3	2	1	0	REG
1050h	P050	STOP BITS	EVEN/ODD PARITY	PARITY ENABLE	ASYNC/ ISOSYNC	ADDRESS/ IDLE WUP	SCI CHAR2	SCI CHAR1	SCI CHAR0	SCICCR
1051h	P051	—	—	SCI SW RESET	CLOCK	TXWAKE	SLEEP	TXENA	RXENA	SCICTL
1052h	P052	Bit 15	Baud Rate Select Register MSB						Bit 8	BAUD MSB
1053h	P053	Bit 7	Baud Rate Select Register LSB						Bit 0	BAUD LSB
1054h	P054	TXRDY	TX EMPTY	—	—	—	—	—	SCITX INT ENA	TXCTL
1055h	P055	RX ERROR	RXRDY	BRKDT	FE	OE	PE	RXWAKE	SCIRX INT ENA	RXCTL
1056h	P056	Reserved								
1057h	P057	Receive Data Buffer Register								RXBUF
1058h	P058	Reserved								
1059h	P059	Transmit Data Buffer Register								TXBUF
105Ah	P05A	Reserved								
105Bh	P05B									
105Ch	P05C									
105Dh	P05D	—	—	—	—	SCICLK DATA IN	SCICLK DATA OUT	SCICLK FUNCTION	SCICLK DATA DIR	SCIPC1
105Eh	P05E	SCITXD DATA IN	SCITXD DATA OUT	SCITXD FUNC-TION	SCITXD DATA DIR	SCIRXD DATA IN	SCIRXD DATA OUT	SCIRXD FUNCTION	SCIRXD DATA DIR	SCIPC2
105Fh	P05F	**SCI STEST**	**SCI TX PRIORITY**	**SCI RX PRIORITY**	**SCI ESPEN**	—	—	—	—	SCIPRI

analog-to-digital converter

The 8-bit analog-to-digital (A/D) converter provides the designer with eight multiplexed analog input channels. The A/D converter has internal sample and hold circuitry and uses a successive approximation conversion technique. The accuracy of the A/D conversion process is increased by providing separate analog positive supply and analog ground input pins (V_{CC3} and V_{SS3}). The V_{SS3} pin also provides the low reference voltage input for the conversion process.

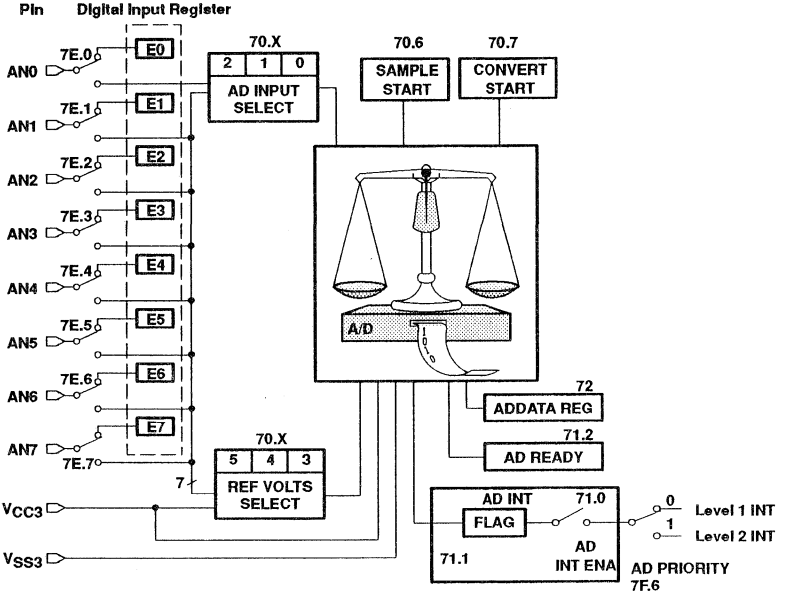


Figure 26. A/D Converter Block Diagram

The A/D converter high reference voltage input is software selectable as one of eight positive reference inputs, as shown in the table below. The A/D conversion process is ratiometric, using V_{SS3} and the software-selected high-reference voltage input as the limits for the selected analog input channel. An input voltage equal to or greater than the high reference input converts to FFh (full scale) with no overflow. An input voltage equal to or less than V_{SS3} converts to 00h. Ratiometric conversions allow analog inputs to be scaled against selected high reference inputs to achieve the greatest accuracy.

AD INPUT SELECT			ANALOG
SEL2	SEL1	SEL0	INPUT CHANNEL
0	0	0	AN0
0	0	1	AN1
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4
1	0	1	AN5
1	1	0	AN6
1	1	1	AN7

REF VOLTS SELECT			HIGH REFERENCE
SEL2	SEL1	SEL0	INPUT
0	0	0	V_{CC3}
0	0	1	AN1
0	1	0	AN2
0	1	1	AN3
1	0	0	AN4
1	0	1	AN5
1	1	0	AN6
1	1	1	AN7

To read an A/D channel:

1. Write to the ADCTL peripheral file control register to:
   - Select the high reference voltage input (ADCTL.5-3).
   - Select the analog input channel for conversion (ADCTL.2-0).
   - Set the SAMPLE START bit to 1 (ADCTL.6).
2. Wait for the sample time to elapse.
3. Write to the ADCTL peripheral file control register to:
   - Set the CONVERT START bit to 1 and leave SAMPLE START bit set to 1.
4. Wait for either the interrupt flag to be set or the A/D interrupt to occur.
5. Read the conversion value from ADDATA when AD INT FLAG is set to 1 or the A/D interrupt occurs.
6. Clear the interrupt flag (ADSTAT.1).

To provide the designer with the flexibility to optimize the A/D conversion process with both high and low impedance sources, the sample time is independently defined by the application program. At the completion of the sample time, the conversion is initiated by settling the CONVERT START and SAMPLE START bits to 1. Eighteen clock cycles after the CONVERT START bit is set to 1, the CONVERT START and SAMPLE START bits will both be set to 0 by the A/D converter, indicating the conversion has started and the analog input signal can be removed. The AD READY bit is set to 0 by the A/D converter to indicate a conversion is in progress. The conversion is complete 164 system clock cycles after it is initiated by setting the CONVERT START bit to 1, and the result is located in the ADDATA result register. Upon completion of the conversion, the AD INT FLAG bit is set, and if the AD INT ENA bit is set to 1 an interrupt will be asserted.

The A/D converter has eight bits of resolution with absolute accuracy of plus or minus one LSB, with (high Reference Voltage – V_{SS3}) = 5 V.

To maximize I/O control capability, all analog input pins not used for an analog input or high reference voltage input to be individually configured as general purpose digital input pins. The control and input data values are contained in the ADENA and ADIN peripheral file control registers.

Peripheral File Frame 7: A-to-D Converter Control Registers

ADDR	PF	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0	REG
1070h	P070	CONVERT START	SAMPLE START	REF VOLT SELECT2	REF VOLT SELECT1	REF VOLT SELECT0	AD INPUT SELECT2	AD INPUT SELECT1	AD INPUT SELECT0	ADCTL
1071h	P071	—	—	—	—	—	AD READY	AD INT FLAG	AD INT ENA	ADSTAT
1072h	P072	A-to-D Conversion Data Register								ADDATA
1073h to 107Ch	P073 to P07C	Reserved								
107Dh	P07D	Port E Data Input Register								ADIN
107Eh	P07E	Port E Input Enable Register								ADENA
107Fh	P07F	AD STEST	AD PRIORITY	AD ESPEN	—	—	—	—	—	ADPRI

Instruction set

The TMS370x5x family instruction set consists of 64 instructions that control input, output, data manipulations, data comparisons, and program flow. The instruction set is supported with fourteen addressing modes to provide the flexibility to optimize programs to the user's applications. For example, the MOV instruction has 27 operand combinations supported by its addressing modes.

ADDRESSING MODE	EXAMPLE		OPERATION
GENERAL:			
Implied	LDSP		(B) → (SP)
Register	MOV	R5,R4	(R5) → (R4)
Peripheral	MOV	P025,A	(1025h) → A
Immediate	ADD	#123,R3	123 + (R3) → (R3)
PC Relative	JMP	offset	PCN + offset → (PC)
Stack Pointer Relative	MOV	2(SP),(A)	(2 + (SP)) → (A)
EXTENDED:			
Absolute Direct	MOV	A,1234	(A) → (1234)
Absolute Indexed	MOV	1234(B),A	(1234 + (B)) → (A)
Absolute Indirect	MOV	@R4,A	((R3:R4)) → (A)
Absolute Offset Indirect	MOV	12(R4),A	(12 + (R3:R4)) → (A)
Relative Direct	JMPL	1234	PCN + 1234 → (PC)
Relative Indexed	JMPL	1234(B)	PCN + 1234 + (B) → (PC)
Relative Indirect	JMPL	@R4	PCN + (R3:R4) → (PC)
Relative Offset Indirect	JMPL	12(R4)	PCN + 12 + (R3:R4) → (PC)

PCN = 16-bit address of next instruction.
(x) = Contents of memory at address x.
((x)) = Contents of memory location designated by contents at address x.

The CPU controls instruction execution by executing microinstructions from a dedicated control memory at a rate of one microinstruction per internal system clock cycle, t_c. The number of system clock cycles required to execute one assembly language instruction varies depending on the instruction complexity, operand addressing mode, and number of wait states. Instruction execution times are stated in terms of the number of integral system clock cycles per instruction. Instruction execution times vary from 5 to 63 internal system clock cycles, with most instructions requiring less than 10 cycles to complete.

Similarly, the number of bytes of program memory required to store an instruction will vary with instruction complexity and addressing mode. TMS370 instructions require from one to five bytes of program memory space, with most instructions occupying one or two bytes.

The *TMS370 Instruction Set Summary*, beginning on page 48, shows the instruction set, the addressing modes, the program memory byte length, and the execution cycle count for each instruction. The Addressing Mode entries are in the format of BYTE LENGTH/CYCLE COUNT. The following symbols and abbreviations are used:

SYMBOL	DEFINITION
s	Source Operand
A	Register A or R0 in Register File
Rs	Source Register in Register File
Ps	Source Register in Peripheral File
Rps	Source Register Pair (Rn, Rn – 1)
Rp	General Purpose Register Pair
iop8	8-bit Immediate Operand
off8	8-bit Signed Offset (label – PCN)
PC	Program Counter
SP	Stack Pointer
#	Immediate Operand
C	Status Register Carry Bit
( )	Contents of

SYMBOL	DEFINITION
d/D	Destination Operand (8-bit/16-bit)
B	Register B or R1 in Register File
Rd	Destination Register in Register File
Pd	Destination Register in Peripheral File
Rpd	Destination Register Pair (Rn, Rn – 1)
label	16-bit Label
iop16	16-bit Immediate Operand
off16	16-bit Signed Offset
PCN	16-bit Address of Next Instruction
ST	Status Register
@	Extended Addressing Operand (Direct, Indirect, Indexed)
→	Is Assigned to

TMS370 Instruction Set Summary

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)	OTHER	
ADC	B, __	1/8										Add with Carry
	Rs, __	2/7	2/7	3/9								(s) + (d) + (C) → (d)
	#lop8, __	2/6	2/6	3/8								
ADD	B, __	1/8										Add
	Rs, __	2/7	2/7	3/9								(s) + (d) → (d)
	#lop8, __	2/6	2/6	3/8								
AND	A, __				2/9							And
	B, __	1/8			2/9							(s). AND. (d) → (d)
	Rs, __	2/7	2/7	3/9								
	#lop8, __	2/6	2/6	3/8	3/10							
BR						3/9	2/8	3/11	4/16			Branch; D → (PC)
BTJO†	A, __,off8				3/10							Bit Test and Jump If One
	B, __,off8	2/10			3/10							If (s). AND. (d) ≠ 0 then
	Rs, __,off8	3/9	3/9	4/11								PCN + offset → (PC)
	#lop8, __,off8	3/8	3/8	4/10	4/11							
BTJZ†	A, __,off8				3/10							Bit Test and Jump If Zero
	B, __,off8	2/10			3/10							If (s).AND.(not d) ≠ 0 then
	Rs, __,off8	3/9	3/9	4/11								PCN + offset → (PC)
	#lop8, __,off8	3/8	3/8	4/10	4/11							
CALL	__					3/13	2/12	3/15	4/20			Call; Push PCN, D → (PC)
CALLR	__					3/15	2/14	3/17	4/22			Call Relative Push PCN, PCN + (d) → (PC)
CLR	__	1/8	1/8	2/6								Clear; 0 → (d)
CLRC											1/19	Clear Carry; 0 → (C)
CMP	__,A					3/11	2/10	3/13	4/18	2/8		Compare
	B, __	1/8										(d) – (s) computed and
	Rs, __	2/7	2/7	3/9								Status Register flags set
	#lop8, __	2/6	2/6	3/8								
CMPBIT	__			3/8	3/10							Complement Bit
CMPL	__	1/8	1/8	2/6								Two's complement; 0100h – (s) → (d)
DAC	B, __	1/10										Decimal Add with Carry
	Rs, __	2/9	2/9	3/11								(s) + (d) + (C) → (d) (BCD)
	#lop8, __	2/8	2/8	3/10								
DEC	__	1/8	1/8	2/6								Decrement; (d) – 1 → (d)
DINT											2/6	Disable Interrupt; 00 → (ST)
DIV	Rs, __	3/55–63‡										Integer Divide; 16 by 8-bit A:B/Rs → A(=quo), B(=rem)

DJNZ† __,off8	2/10	2/1 0	3/8								# cycles depends on operands Decrement and Jump If Not 0 (d) – 1 → (d); If (d) .NE. 0 then PCN + offset → (PC)

† Add 2 to cycle count if jump is taken.
‡ Actual number of cycles is 14 if the quotient is greater than 8 bits (overflow condition).

TMS370 Instruction Set Summary (continued)

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)	OTHER	
DSB	B, __	1/10										Decimal Subtract with Borrow
	RS, __	2/9	2/9	3/11								(d) − (s) − 1 + (C) → (d) (BCD)
	#lop8, __	2/8	2/8	3/10								
EINT											2/6	Enable Interrupts; 0Ch → (ST)
EINTH											2/6	EINT High Priority; 04h → (ST)
EINTL											2/6	EINT Low Priority; 08h → (ST)
IDLE											1/6	Idle Until Interrupt, Low Power entry
INC	__	1/8	1/8	2/6								Increment; (d) + 1 → (d)
INCW	#off8, __			3/11								Increment Word (Rp) + offset → (Rp)
INV	__	1/8	1/8	2/6								Invert; .NOT. (d) → (d)
JMP	__										2/7	Jump; PCN + offset8 → (PC)
JMPL	__					3/9	2/8	3/11	4/16			Jump; PCN + D → (PC)
Jcnd†												Jump Conditional
JN											2/5	Negative
JZ											2/5	Zero
JC											2/5	Carry
JP											2/5	Positive
JPZ											2/5	Positive or Zero
JNZ											2/5	Negative or Zero
JNC											2/5	No Carry
JV											2/5	Overflow, signed
JNV											2/5	No Overflow, signed
JGE											2/5	Greater Than or Equal, signed
JL											2/5	Less Than, signed
JG											2/5	Greater Than, Signed
JLE											2/5	Less Than or Equal, signed
JLO											2/5	Lower Value
JHS											2/5	Higher or Same
JBIT0†	__			4/10	4/11							Jump If Bit = 0
JBIT1 †	__			4/10	4/11							Jump If Bit = 1
LDSP											1/7	Load Stack Pointer; (B) → (SP)
LDST	#lop8										2/6	Load ST Register; (s) → (STP)

MOV	A, __		1/9	2/7	2/8	3/10	2/9	3/12	4/17	2/7		Move; (s) → (d)
	__, A		1/8	2/7	2/8	3/10	2/9	3/12	4/17	2/7		
	B, __	1/8		2/7	2/8							
	Rs, __	2/7	2/7	3/9	3/1							
	Ps, __	2/8	2/8	3/1	0							
	#lop8, __	2/6	2/6	0								
				3/8	3/1							
					0							
MOVW	Rps, __			3/1								Move Word; 16-bit operands
	#lop16, __			2								(s) → (d)
	#lop16(B), __			4/1								
	#off8(Rp), __			3								
				4/1								
				5								
				5/2								
				0								

† Add 2 to cycle count if jump is taken.

TMS370 Instruction Set Summary (concluded)

OPERATION		ADDRESSING MODES										DESCRIPTION
		DIRECT				EXTENDED						
		A	B	Rd	Pd	label	@Rp	label(B)	off8(Rp)	off8(SP)	OTHER	
MPY	B, __	1/47										Multiply
	RS, __	2/46	2/4	3/4								(s) × (d) → (A:B)
	#lop8, __	2/45	6	8								A = MSB, B = LSB
			2/4	3/4								
			5	7								
NOP											1/7	NOP; (PC) +1 → (PC)
OR	A, __				2/9							OR
	B, __	1/8			2/9							(s) .OR. (d) → (d)
	Rs, __	2/7	2/7	3/9								
	#lop8, __	2/6	2/6	3/8	3/1							
					0							
POP	__	1/9	1/9	2/7							1/8	Pop Top of Stack
												((SP)) → (d); (SP) – 1 → (SP)
PUSH	__	1/9	1/9	2/7							1/8	Push onto Stack
												(SP) + 1 → (SP); (s) → ((SP))
RL	__	1/8	1/8	2/6								Rotate Left
RLC	__	1/8	1/8	2/6								Rotate Left Through Carry
RR	__	1/8	1/8	2/6								Rotate Right
RRC	__	1/8	1/8	2/6								Rotate Right Through Carry
RTI											1/12	Return from Interrupt
												Pop PC, Pop ST
RTS											1/9	Return from Subroutine, Pop PC
SBIT0	__			3/8	3/1							Set Bit to 0
					0							
SBIT1	__			3/8	3/1							Set BIT to 1
					0							
SETC											1/7	Set Carry; A0h → (ST)
SBB	B, __	1/8										Subtract with Borrow
	Rs, __	2/7	2/7	3/9								(d) – (s) – 1 + (C) → (d)
	#lop8, __	2/6	2/6	3/8								
STSP											1/8	Store Stack Pointer; (SP) → (B)
SUB	B, __	1/8										Subtract
	Rs, __	2/7	2/7	3/9								(d) – (s) → (d)
	#lop8, __	2/6	2/6	3/8								
SWAP	__	1/11	1/11	2/9								Swap Nibbles
												s(7–4,3–0) → d(3–0,7–4)
TRAPn											1/14	Trap to Subroutine; Push PCN;
												Vector n → (PC)
TST	__	1/9	1/1									Test; Set flags from register
			0									
XCHB	__	1/10	1/1	2/8								Exchange B; (B) ↔ (d)
			0									
XOR	A, __				2/9							Exclusive OR
	B, __	1/8			2/9							(s) .XOR. (d) → (d)
	Rs, __	2/7	2/7	3/9								
	#lop8, __	2/6	2/6	3/8	3/1							
					0							

TMS370 Family OPCODE/Instruction Map

SECOND NIBBLE \ FIRST NIBBLE	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	JMP ra 2/7							INCW #n,Rd 3/11	MOV Ps,A 2/8			CLRC TST A 1/9	MOV A,B 1/9	MOV A,Rd 2/7	TRAP 15 1/14	LDST n 2/6
1	JN ra 2/5		MOV A,Pd 2/8			MOV B,Pd 2/8		MOV Rs,Pd 3/10		MOV Ps,B 2/7				MOV B,Rd 2/7	TRAP 14 1/14	MOV n(SP),A 2/7
2	JZ ra 2/5	MOV Rs,A 2/7	MOV #n,A 2/6	MOV Rs,B 2/7	MOV Rs,Rd 3/9	MOV #n,B 2/6	MOV B,A 1/8	MOV #n,Rd 3/8			MOV Ps,Rd 3/10	DEC A 1/8	DEC B 1/8	DEC Rn 2/6	TRAP 13 1/14	MOV A,n(SP) 2/7
3	JC ra 2/5	AND Rs,A 2/7	AND #n,A 2/6	AND Rs,B 2/7	AND Rs,Rd 3/9	AND #n,B 2/6	AND B,A 1/8	AND #n,Rd 3/8	AND A,Pd 2/9	AND B,Pd 2/9	AND #n,Pd 3/10	INC A 1/8	INC B 1/8	INC Rn 2/6	TRAP 12 1/14	CMP n(SP),A 2/8
4	JP ra 2/5	OR Rs,A 2/7	OR #n,A 2/6	OR Rs,B 2/7	OR Rs,Rd 3/9	OR #n,B 2/6	OR B,A 1/8	OR #n,Rd 3/8	OR A,Pd 2/9	OR B,Pd 2/9	OR #n,Pd 3/10	INV A 1/8	INV B 1/8	INV Rn 2/6	TRAP 11 1/14	extend Inst,2 opcodes
5	JPZ ra 2/5	XOR Rs,A 2/7	XOR #n,A 2/6	XOR Rs,B 2/7	XOR Rs,Rd 3/9	XOR #n,B 2/6	XOR B,A 1/8	XOR #n,Rd 3/8	XOR A,Pd 2/9	XOR B,Pd 2/9	XOR #n,Pd 3/10	CLR A 1/8	CLR B 1/8	CLR Rn 2/6	TRAP 10 1/14	
6	JNZ ra 2/5	BTJO Rs,A 3/9	BTJO #n,A 3/8	BTJO Rs,B 3/9	BTJO Rs,Rd 4/11	BTJO #n,B 3/8	BTJO B,A 2/10	BTJO #n,Rd 4/10	BTJO A,Pd 3/11	BTJO B,Pd 3/10	BTJO #n,Pd 4/11	XCHB A 1/10	XCHB B/ TESTB 1/10	XCHB Rn 2/8	TRAP 9 1/14	IDLE 1/6
7	JNC ra 2/5	BTJZ Rs,A 3/9	BTJZ #n,A 3/8	BTJZ Rs,B 3/9	BTJZ Rs,Rd 4/11	BTJZ #n,B 3/8	BTJZ B,A 2/10	BTJZ #n,Rd 4/10	BTJZ A,Pd 3/10	BTJZ B,Pd 3/10	BTJZ #n,Pd 4/11	SWAP A 1/11	SWAP B 1/11	SWAP Rn 2/9	TRAP 8 1/14	MOV #n,Pd 3/10
8	JV ra 2/5	ADD Rs,A 2/7	ADD #n,A 2/6	ADD Rs,B 2/7	ADD Rs,Rd 3/9	ADD #n,B 2/6	ADD B,A 1/8	ADD, #n,Rd 3/8	MOVW #16,Rd 4/13	MOVW Rs,Rd 3/12	MOVW #16(B),Rd 4/15	PUSH A 1/9	PUSH B 1/9	PUSH Rs 2/7	TRAP 7 1/14	SETC 1/7
9	JL ra 2/5	ADC Rs,A 2/7	ADC #n,A 2/6	ADC Rs,B 2/7	ADC Rs,Rd 3/9	ADC #n,B 2/6	ADC B,A 1/8	ADC #n,Rd 3/8	JMPL lab 3/9	JMPL @Rd 2/8	JMPL lab(B) 3/11	POP A 1/9	POP B 1/9	POP Rd 2/7	TRAP 6 1/14	RTS 1/9
A	JLE ra 2/5	SUB Rs,A 2/7	SUB #n,A 2/6	SUB Rs,B 2/7	SUB Rs,Rd 3/9	SUB #n,B 2/6	SUB B,A 1/8	SUB #n,Rd 3/8	MOV lab,A 3/10	MOV @Rs,A 2/9	MOV lab(B),A 3/12	DJNZ A,ra 2/10	DJNZ B,ra 3/8	DJNZ Rn,ra 3/8	TRAP 5 1/14	RTI 1/12
B	JHS ra 2/5	SBB Rs,A 2/7	SBB #n,A 2/6	SBB Rs,B 2/7	SBB Rs,Rd 3/9	SBB #n,B 2/6	SBB B,A 1/8	SBB #n,Rd 3/8	MOV A,lab 3/10	MOV A,@Rd 2/9	MOV A,lab(B) 3/12	COMPL A 1/8	COMPL B 1/8	COMPL Rn 2/6	TRAP 4 1/14	PUSH ST 1/8
C	JNV ra 2/5	MPY Rs, A 2/46	MPY #n,A 2/45	MPY Rs,B 2/46	MPY Rs,Rd 3/48	MPY #n,B 2/45	MPY B,A 1/47	MPY #n,Rd 3/47	BR lab 3/9	BR @Rd 2/8	BR lab(B) 3/11	RR A 1/8	RR B 1/8	RR Rn 2/6	TRAP 3 1/14	POP ST 1/8
D	JGE ra 2/5	CMP Rs,A 2/7	CMP #n,A 2/6	CMP Rs,B 2/7	CMP Rs,Rd 3/9	CMP #n,B 2/6	CMP B,A 1/8	CMP #n,Rd 3/8	CMP lab,A 3/11	CMP @Rs,A 2/10	CMP lab(B),A 3/15	RRC A 1/8	RRC B 1/8	RRC Rn 2/6	TRAP 2 1/14	LDSP 1/7
E	JG ra 2/5	DAC Rs,A 2/9	DAC #n,A 2/8	DAC Rs,B 2/9	DAC Rs,Rd 3/11	DAC #n,B 2/8	DAC B,A 1/10	DAC #n,Rd 3/10	CALL lab 3/13	CALL @Rd 2/12	CALL lab(B) 3/15	RL A 1/8	RL B 1/8	RL Rn 2/6	TRAP 1 1/14	STSP 1/8
F	JLO ra 2/5	DSB Rs,A 2/9	DSB #n,A 2/8	DSB Rs,B 2/9	DSB Rs,Rd 3/11	DSB #n,B 2/8	DSB B,A 1/10	DSB #n,Rd 3/10	CALLR lab 3/15	CALLR @Rd 2/14	CALLR lab(B) 3/17	RLC A 1/8	RLC B 1/8	RLC Rn 2/6	TRAP 0 1/14	NOP 1/7

Second byte of two-byte instructions (F4xx):

	E	F
8	MOVW n(Rn) 4/15	DIV Rn,A 3/14-63
9	JMPL n(Rn) 4/16	
A	MOV n(Rn),A 4/17	
B	MOV A,n(Rn) 4/17	
C	BR n(Rn) 4/16	
D	CMP n(Rn) 4/18	
E	CALL n(Rn) 4/20	
F	CALLR n(R) 4/22	

ra — relative address
Rn — Register
Rs — Register containing source byte
Rd — Register containing destination byte
Ps — Peripheral register containing source byte
Pd — Peripheral register containing destination byte
Pn — Peripheral register
#n — Immediate 8-bit number
#16 — Immediate 16-bit number
lab — 16-bit label
@Rn — 16-bit address of contents of register pair

development system support

The TMS370 family development support tools include an Assembler, a C-Compiler, a Linker, an In-Circuit emulator (XDS – eXtended Development Support), and an EEPROM/UVEPROM programmer.

- Assembler/Linker (Part No. TMDS3740810-02 for PC, Part No. TMDS3740210-08 for VAX™/VMS™, Part No. TMDS3740510-09 for Sun-3™ or Sun-4™)
  - Extensive macro capability.
  - High-speed operation.
  - Format conversion utilities available for popular formats.
- ANSI C Compiler (Part No. TMDS3740815-02 for PC, Part No. TMDS3740215-08 for VAX™/VMS™, Part No. TMDS3740515-09 for Sun-3™ or Sun-4™)
  - Generates assembly code of the TMS370 that can be easily inspected.
  - The compilation, assembly, and linking steps can all be performed with a single command.
  - Enables the user to directly reference the TMS370's port registers by using a naming convention.
  - Provides flexibility in specifying the storage for data objects.
  - C functions and assembly functions can be easily interfaced.
- CDT370 (Compact Development Tool) In-Circuit Emulator (Part Number EDSCDT370)
  - PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
  - Assembler, Linker, Symbolic debugging.
  - Execute single/multiple instructions, single/multiple statements, until/while condition, or at full speed until breakpoint.
  - Programmation of the EPROM devices.
  - Target cable (Part Number EDSTRG68PLCC)
  - The user needs to provide a regulated 5-V power supply with a 2-A current capability. The board may also be plugged into a PC slot.
- XDS/22 (eXtended Development Support) In-Circuit Emulator (Part Number TMDS3762210 – For PC)
  - Contains all of the features of the XDS/11 described above but does not require an external power supply.
  - Contains sophisticated breakpoint trace and timing hardware that provides up to 2047 qualified trace samples with symbolic disassembly.
  - Allows break points to be qualified by address and/or data on any type of memory acquisition. Up to four levels of events can be combined to cause a breakpoint.
  - Provides timers fo analyzing total and average time in routines.
  - Contains an eight line logic probe for adding visibility of external signals to the breakpoint qualifier and to the trace display.
- EEPROM/EPROM Programmer
  - Base (Part No. TMDS3760500 – base only, requires programmer head)
    - Single unit head (Part No. TMDS3780510).
    - Gang programmer head (Part No. TMDS3780521) supports programming 16 TMS370Cx5x devices in parallel.
  - PC-based, window/function-key oriented user interface for ease of use and a rapid learning environment.
- Design Kit (Part No. TMDS3770110 – For PC)
  - Includes TMS370 Application Board and TMS370 Assembler diskette and documentation.
  - Supports quick evaluation of TMS370 functionality.
  - Capability to upload and download code.
  - Capability to execute programs and software routines, and to single-step executable instructions.
  - Software breakpoints to halt program execution at selected addresses.
  - Wire-wrap prototype area.
  - Reverse assembler.

VMS and **VAX** are trademarks of Digital Equipment Corporation.
Sun-3 and **Sun-4** are trademarks of Sun Microsystems, Inc.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V_{CC1}, V_{CC2}, V_{CC3} (see Note 3) ..... −0.6 V to 7 V
Input voltage range, All pins except MC ..... −0.6 V to 7 V
MC ..... −0.6 V to 14 V
Input clamp current, I_{IK} ($V_I < 0$ or $V_I > V_{CC}$) ..... ±20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{CC}$ ..... ±20 mA
Continuous output current per buffer, I_O (V_O = 0 to V_{CC})‡ ..... ±10 mA
Maximum I_{CC} current ..... 170 mA
Maximum I_{SS} current ..... −170 mA
Continuous power dissipation ..... 1 W
Storage temperature range ..... −65°C to 150 °C

† Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only, and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
‡ Electrical characteristics are specified with all output buffers loaded with specified I_O current. Exceeding the specified I_O current in any buffer may affect the levels on other buffers.

recommended operating conditions (see note 3)

PARAMETER			MIN	NOM	MAX	UNIT
V_{CC1}	Digital logic supply voltage		4.5	5	5.5	V
V_{CC1}	RAM data retention supply voltage (see Note 4)		3		5.5	V
V_{CC2}	Digital I/O supply voltage		4.5	5	5.5	V
V_{CC3}	Analog supply voltage		4.5	5	5.5	V
V_{SS2}	Digital I/O supply ground		−0.3	0	0.3	V
V_{SS3}	Analog supply ground		−0.3	0	0.3	V
V_{IL}	Low-level input voltage	All pins except MC	V_{SS}		0.8	V
		MC	V_{SS}		0.3	
V_{IH}	High-level input voltage	All pins except MC, XTAL2/CLKIN, and $\overline{RESET}$	2		V_{CC}	V
		MC (non-WPO mode)	V_{CC} − 0.3		V_{CC}+0.3	
		XTAL2/CLKIN	0.8 V_{CC}		V_{CC}	
		$\overline{RESET}$	0.7 V_{CC}		V_{CC}	
V_{MC}	MC (mode control) voltage (see Note 5)	EEPROM write protect override	11.7	12	13	V
		Microprocessor	V_{CC} − 0.3		V_{CC}+0.3	
		Microcomputer	V_{SS}		0.3	
		EPROM programming voltage (V_{PP})	12	12.5	13	
T_A	Operating free-air temperature	A version	−40		85	°C
		L version	0		70	°C

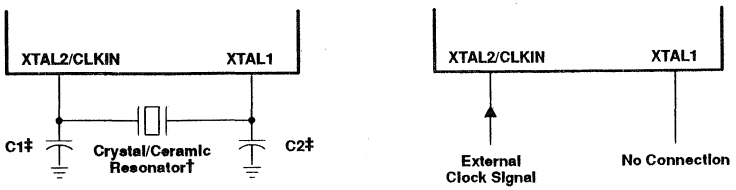
NOTES: 3. All voltage values are with respect to V_{SS1}.
4. $\overline{RESET}$ is externally released while V_{CC} is within the recommended operating range of 4.5 V to 5.5 V and externally activated when V_{CC} < 4.5 or V_{CC} > 5.5 V. RAM data retention is valid throughout the 2 MHz-20 MHz frequency range. An active $\overline{RESET}$ initializes (clears) RAM locations 0000h and 0001h.
5. The basic microcomputer and microprocessor operating modes are selected by the voltage level applied to the dedicated MC pin 2 µs before the $\overline{RESET}$ pin goes inactive (high). The WPO mode may be selected anytime a sufficient voltage is present on the MC pin.

electrical characteristics over recommended operating free-air temperature range (unless otherwise noted)

PARAMETER			TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OL}	Low-level output voltage	Ports A, B, C, D, and $\overline{RESET}$	I_{OL} = 2 mA			0.4	V
		Other outputs	I_{OL} = 1.4 mA			0.4	
V_{OH}	High-level output voltage		I_{OH} = −50 µA	0.9 V_{CC}			V
			I_{OH} = −2 mA	2.4			
I_I	Input current	MC	0 V ≤ V_I ≤ 0.3 V			10	µA
			0.3 V < V_I ≤ V_{CC} − 0.3			50	
			V_{CC} − 0.3 V ≤ V_I ≤ V_{CC} + 0.3 V			10	
			V_{CC}+0.3 V < V_I ≤ 13 V			650	
		I/O pins	0V ≤ V_I ≤ V_{CC}			±10	µA
I_{OL}	Low-level output current	Ports A, B, C, D and $\overline{RESET}$	V_{OL} = 0.4 V	.2			mA
		Other outputs	V_{OL} = 0.4 V	1.4			
I_{OH}	High-level output current		V_{OH} = 0.9 V_{CC}	−50			µA
			V_{OH} = 2.4 V	−2			mA
I_{CC}	Supply current (Operating mode) Osc Power bit = 0 (see Note 8)	TMS370Cx50, TMS370Cx52	Notes 6 and 7 CLKIN = 20 MHz		30	45	mA
		TMS370Cx56			35	56	
		TMS370Cx58			39	62	
		TMS370Cx50, TMS370Cx52	Notes 6 and 7 CLKIN = 12 MHz		20	30	
		TMS370Cx56			22	36	
		TMS370Cx58			24	40	
		TMS370Cx5x	Notes 6 and 7 CLKIN = 2 MHz		7	11	
I_{CC}	Supply current (Standby mode) Osc Power bit = 0 (see Note 9)		Notes 6 and 7 CLKIN = 20 MHz		12	17	mA
			Notes 6 and 7 CLKIN = 12 MHz		8	11	
			Notes 6 and 7 CLKIN = 2 MHz		2.5	3.5	
I_{CC}	Supply current (Standby mode) Osc Power bit = 1 (see Note 10)		Notes 6 and 7 CLKIN = 12 MHz		6	8.6	mA
I_{CC}	Supply current (Halt mode)		Notes 6 and 7 CLKIN = 2 MHz		2	3	
			Note 6 XTAL2/CLKIN < 0.2 V		2	30	µA

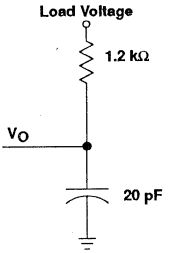
NOTES: 6. Single chip mode, ports configured as inputs, or outputs with no load. All inputs ≤ 0.2 V or ≥ V_{CC} −0.2 V.
7. XTAL2/CLKIN is driven with an external square wave signal with 50% duty cycle and rise and fall times less than 10 ns. Currents may be higher with a crystal oscillator. At 20 MHz this extra current = .01 mA × (total load capacitance + crystal capacitance in pF).
8. Maximum operating current for TMS370Cx50 and TMS370Cx52 = 1.9 (CLKIN) + 7 mA.
Maximum operating current for TMS370Cx56 = 2.5 (CLKIN) + 5.8 mA.
9. Maximum standby current for TMS370Cx5x = 0.75 (CLKIN) + 2 mA.
10. Maximum standby current for TMS370Cx5x = 0.56 (CLKIN) + 1.9 mA. (Osc power bit valid only from 2 MHz to 12 MHz.)

Recommended Crystal/Clock Connections



† The crystal/ceramic resonator frequency is four times the reciprocal of the system clock period.
‡ The values of C1 and C2 should be the values recommended by the crystal/ceramic resonator manufacturer.

Typical Output Load Circuit§



Case 1: $V_O = V_{OH} = 2.4$ V; Load Voltage = 0 V
Case 2: $V_O = V_{OL} = 0.4$ V; Load Voltage = 2.8 V for Ports A, B, C, D, and $\overline{\text{RESET}}$
Load Voltage = 2.1 for other Outputs

§ All measurements are made with the pin loading as shown unless otherwise noted. All measurements are made with XTAL2/CLKIN driven by an external square wave signal with a 50% duty cycle and rise and fall times less than 10 ns unless otherwise stated.

timing parameter symbology

Timing parameter symbols have been created in accordance with JEDEC Standard 100. In order to shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

A	Address	R	Read
AR	Array	RXD	SCIRXD
B	Byte	S	Slave mode
CI	XTAL2/CLKIN	SCC	SCICLK
CO	CLKOUT	SIMO	SPISIMO
D	Data	SOMI	SPISOMI
E	$\overline{EDS}$	SPC	SPICLK
PGM	Program	W	Write
		WT	$\overline{WAIT}$

Lowercase subscripts and their meanings are:

c	cycle time (period)	r	rise time
d	delay time	su	setup time
f	fall time	v	valid time
h	hold time	w	pulse duration (width)

The following additional letters are used with these meanings:

H	High	V	Valid
L	Low	Z	High Impedance

PARAMETER MEASUREMENT INFORMATION

All timings are measured between high and low measurement points as indicated in the figures below.

XTAL2/CLKIN Measurment Points

General Measurement Points

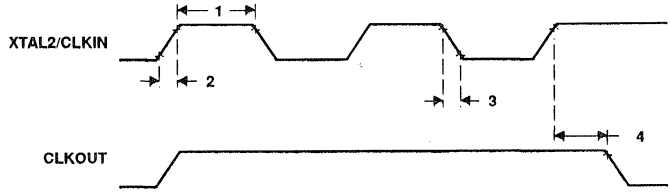
external clocking requirements†

NO.	PARAMETER		MIN	NOM	MAX	UNIT
1	$t_{w(CI)}$	XTAL2/CLKIN pulse duration (see Note 11)	20			ns
2	$t_{r(CI)}$	XTAL2/CLKIN rise time			30	ns
3	$t_{f(CI)}$	XTAL2/CLKIN fall time			30	ns
4	$t_{d(CIH-COL)}$	Delay time, XTAL2/CLKIN rise to CLKOUT fall			100	ns
	f_X	Crystal operating frequency	2		20	MHz

† For V_{IL} and V_{IH}, refer to "Recommended Operating Conditions".

NOTE 11: This pulse may be either a high pulse, as illustrated below, which extends from the earliest valid high to the final valid high in an XTAL2/CLKIN cycle, or a low pulse which extends from the earliest valid low to the final valid low in an XTAL2/CLKIN cycle.

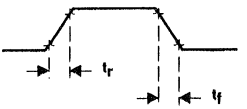
external clock timing



EXPANSION MODE OUTPUT

general purpose output signal switching time requirements

		MIN	NOM	MAX	UNIT
t_r Rise time	A, B, C, D, and $\overline{\text{RESET}}$		15		ns
	INT2 , INT3, SPISOMI, SPISIMO, SPICLK, T1IC/CR, T1PWM, T1EVT, T2IC1/CR, T2IC2/PWM, T2EVT, SCITXD, SCIRXD, SCICLK		30		ns
t_f Fall time	A, B, C, D, and $\overline{\text{RESET}}$		15		ns
	INT2, INT3, SPISOMI, SPISIMO, SPICLK, T1IC/CR, T1PWM, T1EVT, T2IC1/CR, T2IC2/PWM, T2EVT, SCITXD, SCIRXD, SCICLK		30		ns



recommended EPROM operating conditions for programming

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.75	5.5	6.0	V
V_{PP}	Supply voltage at MC pin	12	12.5	13	V
I_{PP}	Supply current at MC pin during programming (V_{PP} = 13 V)		30	50	mA
CLKIN	Operating crystal frequency	2		20	MHz

recommended EPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(IEPGM)}$	Initial programming pulse (see Note 12)	0.95	1	1.05	ms
$t_{w(FEPGM)}$	Final programming pulse	2.85		78.75	ms

NOTE 12: Programming pulse is active when both EXE (EPCTL.0) and VPPS (EPCTL.6) are set.

recommended EEPROM timing requirements for programming

		MIN	NOM	MAX	UNIT
$t_{w(PGM)B}$	Programming signal pulse duration to insure valid data is stored (byte mode)	10			ms
tw(PGM)AR	Programming signal pulse duration to insure valid data is stored (array mode)	20			ms

switching characteristics and timing requirements

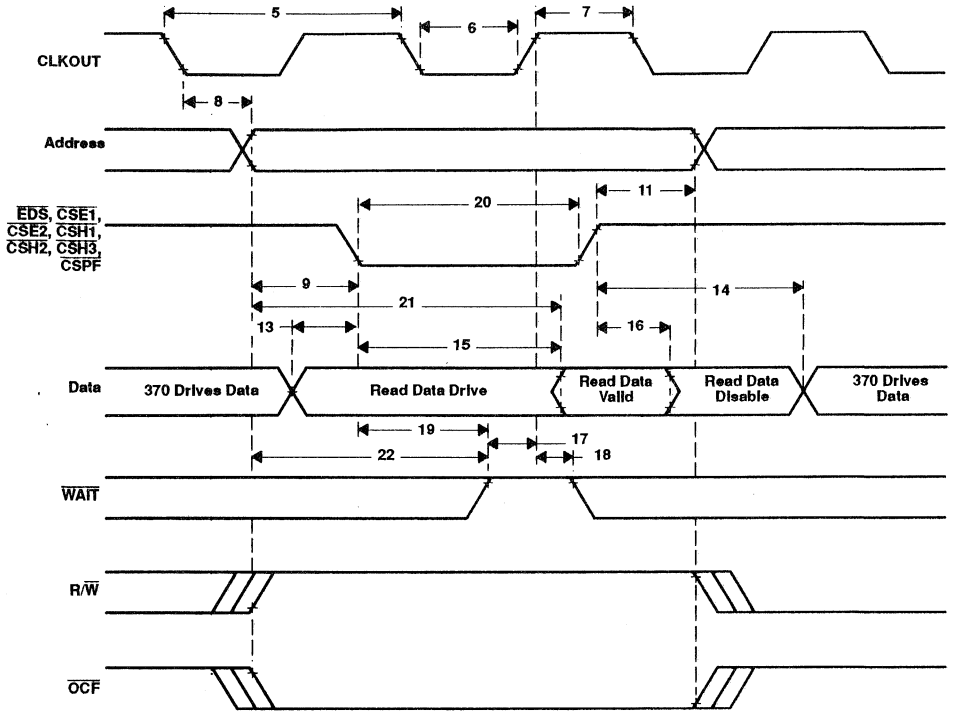
NO.	PARAMETER		MIN	MAX	UNIT
5	t_c	CLKOUT (system clock) cycle time (see Note 13)	200	2000	ns
6	$t_{w(COL)}$	CLKOUT low pulse duration	$0.5t_c - 20$	$0.5t_c$	ns
7	$t_{w(COH)}$	CLKOUT high pulse duration	$0.5t_c$	$0.5t_c+20$	ns
8	$t_{d(COL-A)}$	Delay time, CLKOUT low to address, R/$\overline{W}$, and $\overline{OCF}$ valid		$0.25t_c+40$	ns
9	$t_{v(A)}$	Address valid to $\overline{EDS}$, $\overline{CSE1}$, $\overline{CSE2}$, $\overline{CSH1}$, $\overline{CSH2}$, $\overline{CSH3}$, and $\overline{CSPF}$ low	$0.5t_c - 50$		ns
10	$t_{su(D)}$	Write data setup time to $\overline{EDS}$ high	$0.75t_c - 40$†		ns
11	$t_{h(EH-A)}$	Address, R/$\overline{W}$, and $\overline{OCF}$ hold time from $\overline{EDS}$, $\overline{CSE1}$, $\overline{CSE2}$, $\overline{CSH1}$, $\overline{CSH2}$, $\overline{CSH3}$, and $\overline{CSPF}$ high	$0.5t_c - 40$		ns
12	$t_{h(EH-D)W}$	Write data hold time from $\overline{EDS}$ high	$0.75t_c+15$		ns
13	$t_{d(DZ-EL)}$	Delay time, data bus high impedance to $\overline{EDS}$ low (read cycle)	$0.25t_c - 30$		ns
14	$t_{d(EH-D)}$	Delay time, $\overline{EDS}$ high to data bus enable (read cycle)	$1.25t_c - 40$		ns
15	$t_{d(EL-DV)}$	Delay time, $\overline{EDS}$ low to read data valid		$t_c - 65$†	ns
16	$t_{(EH-D)R}$	Read data hold time from $\overline{EDS}$ high	0		ns
17	$t_{su(WT-COH)}$	$\overline{WAIT}$ setup time to CLKOUT high	$0.25t_c+75$‡		ns
18	$t_{h(COH-WT)}$	$\overline{WAIT}$ hold time from CLKOUT high	0		ns
19	$t_{d(EL-WTV)}$	Delay time, $\overline{EDS}$ low to $\overline{WAIT}$ valid		$0.5t_c - 70$	ns
20	t_w	Pulse duration; $\overline{EDS}$, $\overline{CSE1}$, $\overline{CSE2}$, $\overline{CSH1}$, $\overline{CSH2}$, $\overline{CSH3}$, and $\overline{CSPF}$ low	$t_c - 40$†	t_c+40†	ns
21	$t_{d(AV-DV)R}$	Delay time, address valid to read data valid		$1.5t_c - 75$†	ns
22	$t_{d(AV-WTV)}$	Delay time, address valid to $\overline{WAIT}$ valid		$t_c - 85$	ns
23	$t_{d(AV-EH)}$	Delay time, address valid to $\overline{EDS}$ high (end of write)	$1.5t_c - 40$†		ns

NOTE 13: t_c = system clock cycle time = $4/f_x$.

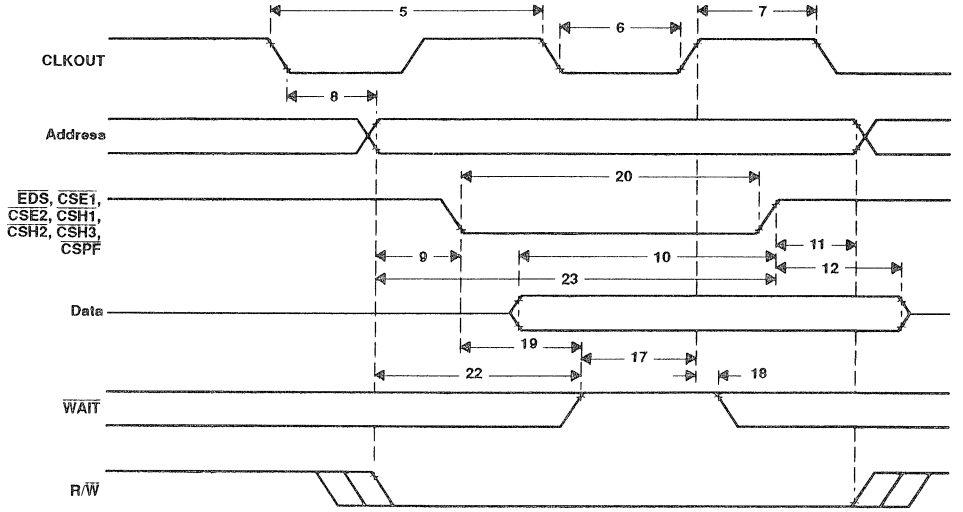
† If wait states, PFWait, or the Auto-Wait feature are used, add t_c to this value for each wait state invoked.

‡ If the Auto-Wait feature is enabled, the $\overline{WAIT}$ input may assume a "Don't Care" condition until the third cycle of the access. The $\overline{WAIT}$ signal needs to be synchronized with the high pulse of the CLKOUT signal while still observing the minimum setup time.

external read timing



external write timing



SERIAL COMMUNICATIONS INTERFACE (SCI) INTERNAL CLOCK ISOSYNCHRONOUS MODE I/O TIMING

SCI isosynchronous mode timing characteristics for internal clock (see Note 13)

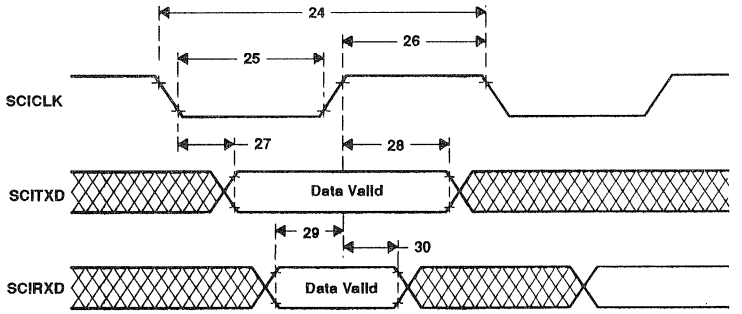
NO.	PARAMETER		MIN	MAX	UNIT
24	$t_{c(SCC)}$	SCICLK cycle time	$2t_c$	$131{,}072t_c$	ns
25	$t_{w(SCCL)}$	SCICLK low pulse duration	$t_c - 45$	$0.5t_{c(SCC)}+45$	ns
26	$t_{w(SCCH)}$	SCICLK high pulse duration	$t_c - 45$	$0.5t_{c(SCC)}+45$	ns
27	$t_{d(SCCL-TXDV)}$	Delay time, SCITXD valid after SCICLK low	− 50	50	ns
28	$t_{v(SCCH-TXD)}$	SCITXD data valid after SCICLK high	$t_{w(SCCH)} - 50$		ns

SCI isosynchronous mode timing requirements for internal clock (see Note 13)

NO.	PARAMETER		MIN	MAX	UNIT
29	$t_{su(RXD-SCCH)}$	SCIRXD setup time to SCICLK high	$0.25t_c+145$		ns
30	$t_{v(SCCH-RXD)}$	SCIRXD data valid after SCICLK high	0		ns

NOTE 13: t_c = system clock period time = $4/f_x$.

SCI isosynchronous mode timing diagram for internal clock



SERIAL COMMUNICATIONS INTERFACE (SCI) EXTERNAL CLOCK ISOSYNCHRONOUS MODE I/O TIMING

SCI Isosynchronous mode timing characteristics for external clock (see Note 13)

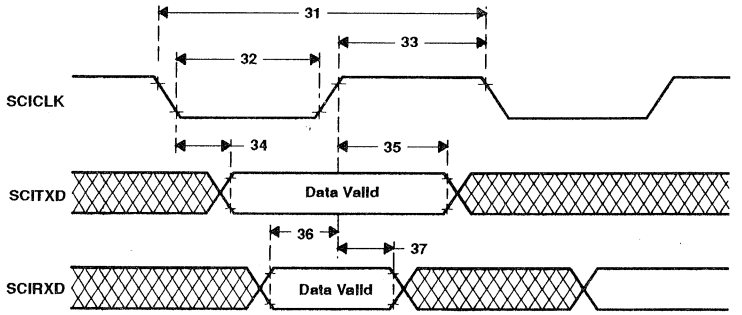
NO.	PARAMETER		MIN	MAX	UNIT
34	$t_{d(SCCL-TXDV)}$	Delay time, SCITXD valid after SCICLK low		$40.25t_c+145$	ns
35	$t_{v(SCCH-TXD)}$	SCITXD data valid after SCICLK high	$t_{w(SCCH)}$		ns

SCI Isosynchronous mode timing requirements for external clock (see Note 13)

NO.			MIN	MAX	UNIT
31	$t_{c(SCC)}$	SCICLK cycle time	$10t_c$		ns
32	$t_{w(SCCL)}$	SCICLK low pulse duration	$4.25t_c+120$		ns
33	$t_{w(SCCH)}$	SCICLK high pulse duration	t_c+120		ns
36	$t_{su(RXD-SCCH)}$	SCIRXD setup time to SCICLK high	40		ns
37	$t_{v(SCCH-RXD)}$	SCIRXD data valid after SCICLK high	$2t_c$		ns

NOTE 13: t_c = system clock period time = $4/f_x$.

SCI Isosynchronous mode timing diagram for external clock



SERIAL PERIPHERAL INTERFACE (SPI) TIMING

SPI master external timing characteristics (see Note 13)

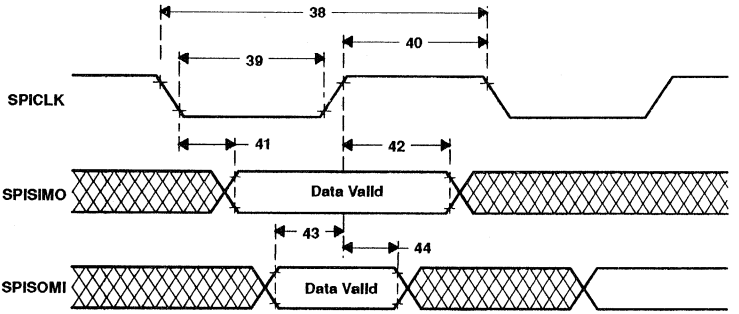
NO.	PARAMETER		MIN	MAX	UNIT
38	$t_{c(SPC)}$	SPICLK cycle time	$2t_c$	$256t_c$	ns
39	$t_{w(SPCL)}$	SPICLK low pulse duration	$t_c - 45$	$0.5t_{c(SPC)}+45$	ns
40	$t_{w(SPCH)}$	SPICLK high pulse duration	$t_c - 45$	$0.5t_{c(SPC)}+45$	ns
41	$t_{d(SPCL-SIMOV)}$	Delay time, SPISIMO valid after SPICLK low (Polarity = 1)	− 50	50	ns
42	$t_{v(SPCH-SIMO)}$	SPISIMO data valid after SPICLK high (Polarity = 1)	$t_{w(SPCH)} - 50$		ns

SPI master external timing requiremtnts (see Note 13)

NO.	PARAMETER		MIN	MAX	UNIT
43	$t_{su(SOMI-SPCH)}$	SPISOMI setup time to SPICLK high (Polarity =1)	$0.25t_c+150$		ns
44	$t_{v(SPCH-SOMI)}$	SPISOMI data valid after SPICLK high (Polarity =1)	0		ns

NOTE 13: t_c = system clock period time = $4/f_x$.

SPI master external timing



NOTE 14: The diagram above is for Polarity = 1. SPICLK is inverted from above diagram when Polarity = 0.

SERIAL PERIPHERAL INTERFACE (SPI) TIMING

SPI master external timing characteristics (see Note 13)

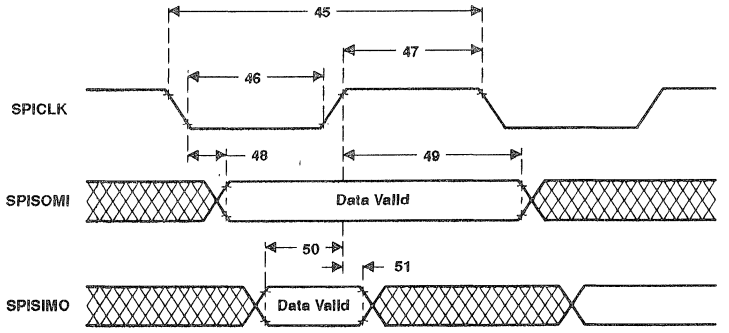
NO.	PARAMETER		MIN	MAX	UNIT
48	$t_{d(SPCL-SOMIV)S}$	Delay time, SPISOMI valid after SPICLK low (Polarity = 1)		$3.25t_c + 125$	ns
49	$t_{v(SPCH-SOMI)S}$	SPISOMI data valid after SPICLK high (Polarity =1)	$t_{w(SPCH)S}$		ns

SPI slave external timing requirements (see Note 13)

NO.			MIN	MAX	UNIT
45	$t_{c(SPC)S}$	SPICLK cycle time	$8t_c$		ns
46	$t_{w(SPCL)S}$	SPICLK low pulse duration	$4t_c - 45$	$0.5t_{c(SPC)S}+45$	ns
47	$t_{w(SPCH)S}$	SPICLK high pulse duration	$4t_c - 45$	$0.5t_{c(SPC)S}+45$	ns
50	$t_{su(SIMO-SPCH)S}$	SPISIMO setup time to SPICLK high (Polarity = 1)	0		ns
51	$t_{v(SPCH-SIMO)S}$	SPISIMO data valid after SPICLK high (Polarity = 1)	$3t_c+100$		ns

NOTE 13: t_c = system clock period time = $4/f_x$.

SPI slave external timing



NOTES: 14. The diagram above is for Polarity = 1. SPICLK is inverted from above diagram when Polarity = 0.
15. As a slave, the SPICLK pin is used as the input for the serial clock, which is supplied from the network master.

A/D Converter

The A/D Converter has a separate power bus for its analog circuitry. These pins are referred to as V_{CC3} and V_{SS3}. The purpose is to enhance A/D performance by preventing digital switching noise on the logic circuitry which could be present on V_{SS} and V_{CC} from coupling into the A/D analog stage. All A/D specifications will be given with respect to V_{SS3} unless otherwise noted.

Resolution ..... 8 bits (256 values)
Monotonic ..... Yes
Output conversion code ..... 00h to FFh (00 for $V_I \leq V_{SS3}$; FF for $V_I \geq V_{ref}$)
Conversion time (excluding sample time) ..... $164t_c$

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC3}	Analog supply voltage	4.5	5	5.5	V
		$V_{CC} - 0.3$		$V_{CC} + 0.3$	
V_{SS3}	Analog ground	$V_{SS} - 0.3$		$V_{SS} + 0.3$	V
V_{ref}	Non-V_{CC3} reference (See Note 16)	2.5	V_{CC3}	$V_{CC3} + 0.1$	V
	Analog input for conversion	V_{SS3}		V_{ref}	V

NOTE 16: V_{ref} must be stable, within ± 1/2 LSB of the required resolution, during the entire conversion time.

operating characteristics over full ranges of recommended operating conditions

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
	Absolute accuracy (see Note 17)	V_{CC3} = 5.5 V, V_{ref} = 5.1 V			±1	LSB
	Differential/integral linearity error (see Notes 17 and 18)	V_{CC3} = 5.5 V, V_{ref} = 5.1 V			±0.5	LSB
I_{CC3}	Analog supply current	Converting			2	mA
		Non-Converting			5	µA
I_I	Input current, AN0–AN7	$0 V \leq V_I \leq 5.5 V$			2	µA
	V_{ref} input charge current				1	mA
Z_{ref}	Source impedance of V_{ref}	XTAL2/CLKIN ≤ 12 MHz			24	kΩ
		12 MHz < XTAL2/CLKIN ≤ 20 MHz			10	kΩ

NOTES: 17. Absolute resolution = 20 mV. At V_{ref} = 5 V, this is 1 LSB. As V_{ref} decreases, LSB size decreases and thus absolute accuracy and differential/integral linearity errors in terms of LSBs increases.
18. Excluding quantization error of 1/2 LSB.

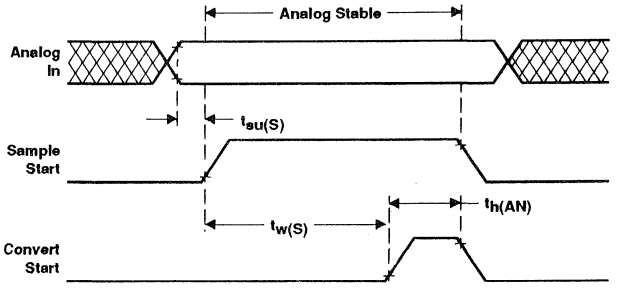
The A/D module allows complete freedom in design of the sources for the analog inputs. The period of the sample time is user-defined such that high impedance sources can be accommodated without penalty to low-impedance sources. The sample period begins when the SAMPLE START bit of the A/D Control Register (ADCTL) is set to 1. The end of the signal sample period occurs when the conversion bit (CONVERT START) of the ADCTL is set to 1. After a hold time, the converter will reset the SAMPLE START and CONVERT START bits, signaling that a conversion has started and the analog signal can be removed.

analog timing requirements

		MIN	NOM	MAX	UNIT
$t_{su(S)}$	Analog input setup to sample command	0			ns
$t_{h(AN)}$	Analog input hold from start of conversion	$18t_c$			ns
$t_{w(S)}$	Duration of sample time per kilohm of source impedance (see Note 19)	1			µs/kΩ

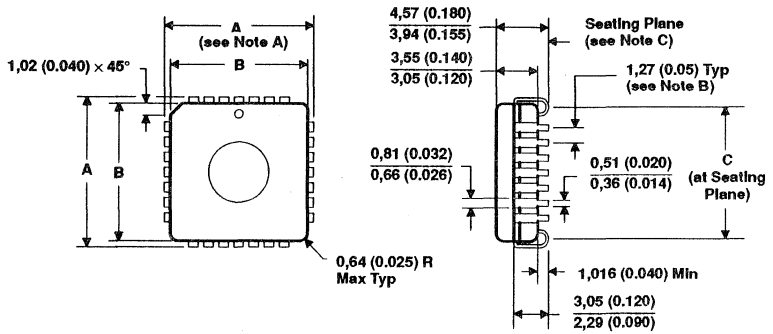
NOTE 19: The value given is valid for a signal with a source impedance greater than 1 kΩ. If the source impedance is less than 1 kΩ, use a minimum sampling time of 1 µs.

analog timing



MECHANICAL DATA

68-lead FZ cerquad chip carrier package

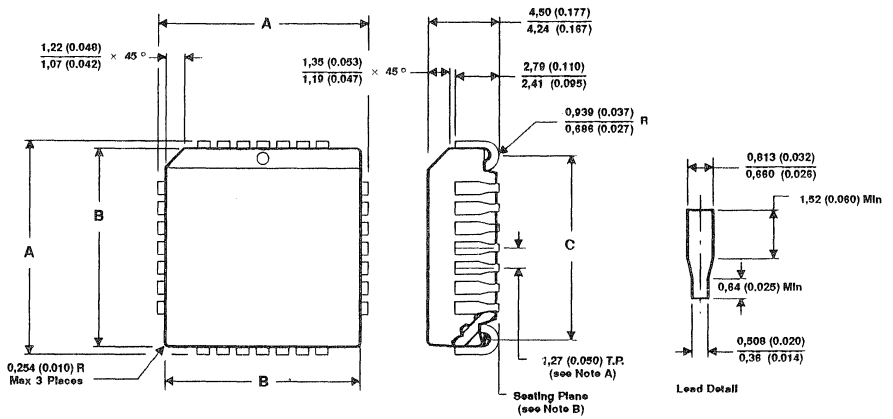


JEDEC OUTLINE	NO. OF TERMINALS	A		B		C	
		MIN	MAX	MIN	MAX	MIN	MAX
MO-087AA	28	12,32 (0.485)	12,57 (0.495)	10,92 (0.430)	11,56 (0.455)	10,41 (0.410)	10,92 (0.430)
MO-087AB	44	17,40 (0.685)	17,65 (0.695)	16,00 (0.630)	16,64 (0.655)	15,49 0.610)	16,00 (0.630)
MO-087AD	68	25,02 (0.985)	25,27 (0.995)	23,62 (0.930)	24,26 (0.955)	23,11 (0.910)	23,62 (0.930)

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTES: A. Center line of center pin each side is within 0,10 (0.004) of package centerline as determined by dimension B.
B. Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
C. The lead contact points are planar with 0,15 (0.006).

68-pin plastic leaded chip carrier package (FN suffix)



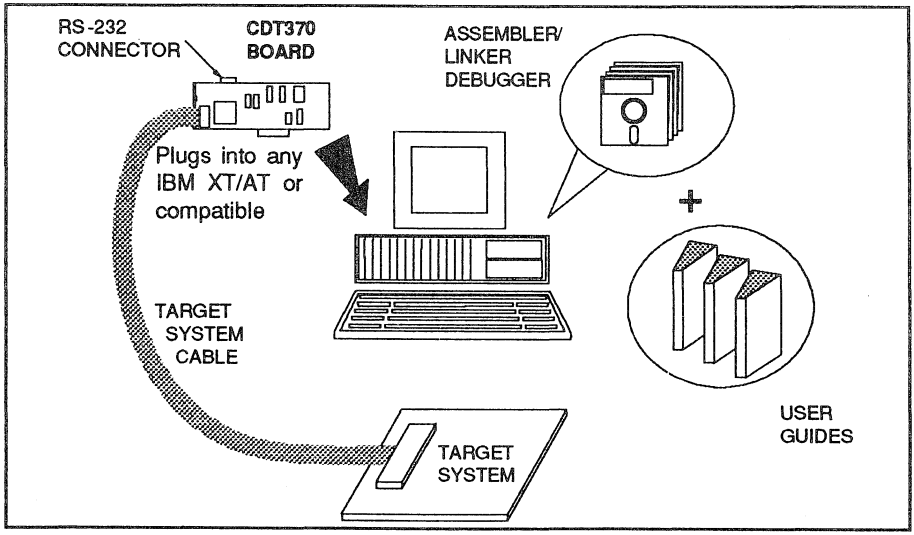
JEDEC OUTLINE	NO. OF TERMINALS	A		B		C	
		MIN	MAX	MIN	MAX	MIN	MAX
M0-047AB	28	12,32 (0.485)	12,57 (0.495)	10,92 (0.430)	11,56 (0.455)	10,41 (0.410)	10,92 (0.430)
M0-047AC	44	17,40 (0.685)	17,65 (0.695)	16,00 (0.630)	16,64 (0.655)	15,49 (0.610)	16,00 (0.630)
M0-047AE	68	25,02 (0.985)	25,27 (0.995)	23,62 (0.930)	24,26 (0.955)	23,11 (0.910)	23,62 (0.930)

ALL LINEAR DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

NOTES: A. Location of each pin is within 0,127 (0.005) of true position with respect to center pin on each side.
B. The lead contact points are planar within 0,101(0.004).

CDT370
TMS370 Compact Development Tool

- Complete Development Solution in One Package.
- Plugs Into IBM XT/AT or Compatible.
- Emulates TMS370 Family Members.
- Integrated EEPROM Programmer.
- Interactive, Windowed Debugger.
- Real-Time Emulation.
- Shorter Development Cycle.
- Unpack and Begin - Everything is Provided.
- Quick and Easy to Configure No Delays.
- Compact and Economic Development Solution.
- Supports FPM Device Programmation.
- Optimized User Interface, Enhances Productivity.
- When Only Real-Time will Do.
- Faster to Market.



CDT370 Configuration

TEXAS INSTRUMENTS

About the CDT370

The CDT370 offers a low cost but highly efficient route to TMS370 family development. In addition, the CDT370 supports programmation of the new Field Programmable Microcontroller (FPM) family members. Features such as the new interactive windowed debugger, real-time emulation and integrated EEPROM programmer, all contribute to enhanced user productivity and consequently a shorter design cycle. A 1024 program step real-time trace facility is complemented by an on-board timer (up to 3.5 S.) for measuring program execution time.

The CDT370 Compact Development Tool comprises:

- CDT370 Emulator Board.
- Assembler/ Linker for IBM XT/AT or compatible.
- New windowed debugger for IBM XT/ AT or compatible.
- Complete supporting documentation.
- Target Cable. (specific to each device).

Once the CDT370 is unpacked, development can start immediately. Everthing required to emulate the TMS370 family single-chip mode is provided.

The CDT370 is a single-board Compact Development Tool (CDT) for the TMS370 family (for the TMS370C732 PACT device, use of the XDS22 eXtended Development System is recommended). It is designed to plug-in to the expansion chassis of any IBM XT/AT or compatible PC and will run under MS-DOS. In this mode, the CDT370 will function without any external power supply. Once the board is installed in the PC there remains only the target system cable to connect and development can begin. Should there be no free slots in the PC, an RS-232 serial interface is provided. In this case an external +5V supply voltage is required.

Description

The CDT370 comprises two major functional areas: the master CPU and the SE370C050 emulator chip. A functional block diagram of the CDT370 is shown in Figure 1.

Basic control of the CDT370 board is maintained by the TMS9995 16-bit CPU. This device communicates with the SE370C050 emulator chip over an 8-bit data bus buffered by an octal tranceiver. Resident firmware is stored in a 256K byte EPROM and an 8K byte RAM provides the necessary workspace. Communication with the PC is implemented via a TMS9901 Programmable Systems Interface (PSI) and TMS9902 Asynchronous Communications Controller (ACC). The ACC accepts serial data from either a UART connected directly to the PC bus or an external RS-232 connector. The PSI provides interrupt control facilities as well as a serial communications link to the CPU. The data rate automatically defaults to 19,200 Baud but is also software selectable by the user.

The SE370C050 emulator chip emulates all the features of the TMS370 family (except the TMS370C732 PACT device). These include four 8-bit ports, 2 timers, A/D converter, SPI and SCI. Real-time EEPROM emulation, implemented in logic is also provided. This may be mapped to any part of the address space and used instead of target system memory. The SE370C050 may be driven by either the on-board clock or target cable clock, at the discretion of the user.

For the FPM family members, an EPROM and data EEPROM programmation interface generates all the necessary signals. These are subsequently fed into the target cable in order that the target system connector may be used for programmation.

Operating System

The CDT operating system software is delivered on three diskettes and comprises the following major parts:

- Assembler/Linker
- Archiver
- Code Conversion Utility
- Interactive Debug monitor and EPROM programmer.

All the software runs under MS-DOS and is designed to interact with the user to provide a complete, powerful and easy-to-use development tool.

The interactive nature of the software enables

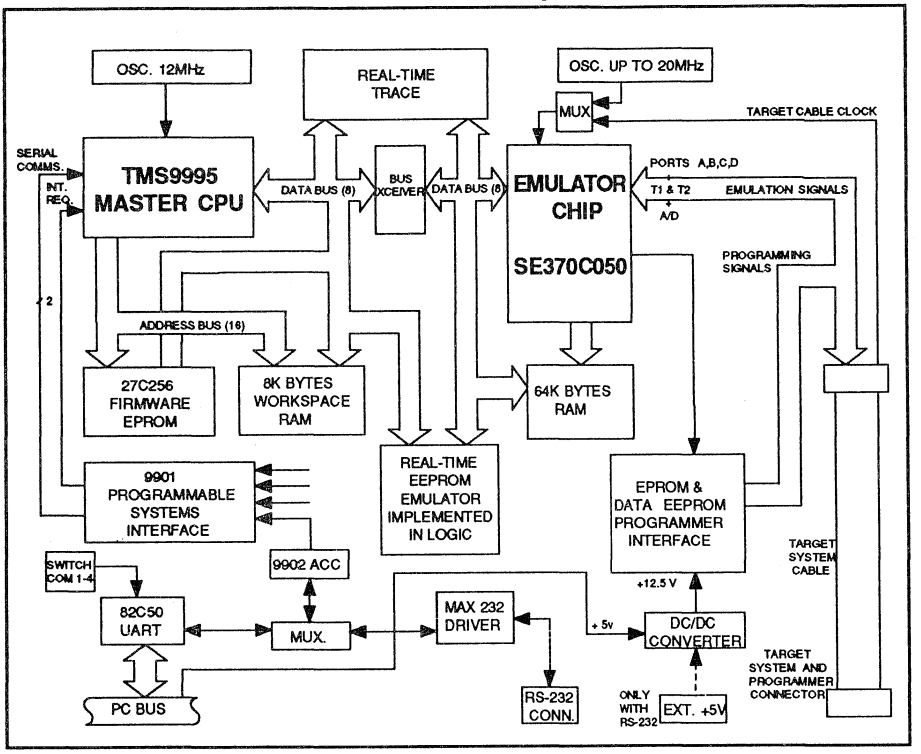


Figure 1- CDT370 Block Diagram

the user to exploit the features of the CDT at maximum efficiency. Optimization of the man-machine interface is achieved through an enhanced level of communication, resulting in better user productivity.

The TMS370 system designer can use any text editor to generate the assembler source code, then use the assembly language tools to assemble and link the executable object code modules. The debugger is used to download the code into target system or emulator memory for subsequent execution. After downloading, the powerful window orientated debugger user interface permits a wide variety of operations to aid the development process.

Assembler/Linker

The TMS370 assembler is a two-pass assembler, creating object code modules in Common Object File Format (COFF). The COFF encourages and facilitates modular programming by allowing the assembler to maintain a Section Program Counter (SPC) for each section of object code generated.

Powerful assembler directives control various aspects of the assembly process. These include source listing format, symbol definition, conditional assembly blocks, macro library definition and how the machine code is placed in TMS370 memory space.

Symbol tables contained in the COFF object files allow the debugger to provide the user with symbolic debugging. This supports the setting of breakpoints using addresses, line numbers or symbols.

The Linker creates executable object modules

by combining the COFF object files. Several file types are accepted as input:

- Relocatable COFF object files.
- Command files.
- Archive object libraries.
- Output modules created by a previous linker.

The linker supports a "C" like command language with powerful directives that control memory configuration, section definition and address binding.

Archiver

The archiver provides file management by allowing a group of files to be collected into a single library. Macros can be collected by the archiver, then fetched by the assembler as directed by the source file. Object modules can also be collected into a library for convenient access by the linker.

While not mandatory for program development, the archiver is a valuable organizational tool in the building of an executable COFF object file.

Interactive Debugger

Once an executable object module has been produced by the TMS370 assembler and linker, the debugger is used to load it into emulator memory and execute it.

Debugging a system may require intervention in a number of different areas: the code being executed; the registers of the target machine; the variables in the progam; etc. The TMS370 Debugger displays this data in a set of windows on the screen.

TMS370 Interactive Debugger Features

- Window orientated user interface with menu-driven command structure.
- Ability to change and display registers and memory.
- Full access to symbol tables.
- A line-by-line patch assembler.
- A symbolic reverse assembler that displays object code.
- Full symbolic expression analysis that recognizes all assembly language operators.
- Full control of microcomputer execution, including single-step.
- Software breakpoints that halt execution at selected addresses.
- 1024 Program step trace.
- Continuous run mode, allowing above features to be used while the MCU is running.
- Memory mapping that allows appropriate configuration of emulator memory.
- Timer/Counter upto 3.5 seconds.

Each window contains information pertinent to one aspect of the debugging process. The user can move from window to window to perform specific operations such as moving to the code window to examine code or moving to the CPU register window to clear a register.

The windows are designated:

Code	Display	CPU Registers .
R file	Stack	EPROM Programmer

The windows are based on virtual buffers in the debugger. The debugger therefore keeps track of more information than is actually displayed in a given window. This allows the user to scroll the window up or down very quickly because the debugger does not need to request the required data from the CDT. The windows are automatically updated whenever the microcontroller stops running, or by the modify command. When the values change in a window conaining register or memory information, the new values are highlighted so that they are easily identified on the screen.

In addition to the various windows, the top line of the entry level debugging screen shows a menu of single-letter command options. In the second line a display of current system status is shown The bottom line shows the active function key assignements.

The command language is designed to be both simple for the inexperienced user and efficient for the expert. This is accomplished by limiting command menus to just one or two levels, so that almost all debugger functions can be controlled by a simple two-letter command with no wasted keystrokes.

The ten function keys complement the command language by providing additional control of the debugger itself. Function keys are used to scroll windows up and down, and to move between win-

```
Debug: Display  eXecute  Halt  Mem  Bpoint  Config  Trace  Load  mOdule  Reg  Eval  Prog -->
                                        CPU : HALTED
═══════ code ═══════                                  ═══ cpu registers ═══
WRLOOP:                                      PC  702A  SP 42    ST   cnzv21
    7027 AB0100     MOV A, : 00100H(B)       A   3A    B  47    00   000000
    702A B3         INC  A                  ═══ r file ═══   ═ stack ═
    702B C3         INC  B                  R02  EE  R03  F3   SP   (42) 1E
    702C 5DFC       CMP  #0FCh,B            R04  00  R05  00   -    1(41) 70
    702E 0657       JNZ  WRLOOP             R06  00  R07  00   -    2(40) 00
    7030 420302     MOV  R03, R02           R08  00  R09  00   -    3(3F) 00
    7033 C5         CLR  B                  R0A  00  R0B  00   -    4(3E) 00
RDLOOP:                                     R0C  00  R0D  00   -    5(3D) 00
    7034 AA0100     MOV  :00100H(B),A       R0E  00  R0F  00   -    6(3C) 00
═══════ display ═══════                     R010 00  R011 00   -    7(3B) 00
                                            R012 00  R013 00   -    8(3A) 00
                                            R014 00  R015 00   -    9(39) 00
                                                  ═══ expressions ═══
    TMS370 :   370 USI Debugger v1.00       CP21                    04Ah
               370 CDT v1.00                CP43                    04Bh
                                            CP65                    04Ch
    Copyright © 1991 Texas Instruments, Inc. INPUTCTL                04Dh
                                            WDKEY                   04Eh
                                            RESOL                   04Fh
                                            ERRORMSG                000h
F1 Inspect                    F9 Update  F10 Help
```

Interactive Debugger - Top Level Screen

dows etc. The actual function keys displayed and available at any particular time are activity dependent.

During execution, several modes of fixed display are available - a hex display of all register and peripheral files for example, or a binary display of the peripheral ports. While using a fixed display, subsequent execution to a breakpoint or execution of a single instruction step will overwrite the old data on the screen with new data. A programmable line of up to six registers or peripheral locations is provided for display with breakpoints and instruction steps.

Code Window

Located in the upper left-hand corner of the screen, the Code Window is dedicated to displaying the code being debugged. The code in this window is disassembled from object code in memory.

The disassembler uses the symbol table and with the exception of relocatable register symbols, inserts labels into the disassembly, whenever possible. The instruction at the current PC value is identified with a highlighted address. Instructions at which simple breakpoints have been set are also identified.

CPU Registers Window

Five TMS370 registers (A, B, PC, SP and ST) are displayed in the CPU registers window located in the upper right hand corner of the screen. This window cannot be scrolled but the register contents may be modified.

Display Window

This window is located in the lower left corner The functions (F Keys) available depend on what is currently being displayed.

Configure Window Displayed Items	
Inspect or Change	
Run Mode	(run/continuous)
Clock Source	(target/osc)
Inspect Only	
Device Mode	(μprocessor/μcomputer)
Clock Period	
Configure Commands	
Memory Map	Configure the memory map
Load	Load memory map from file
Save	Save memory map to a file
Reset	Reset all memory ranges to device default
Device Select	Select a standard TMS370xx device
Reset	Reset the TMS370 device
Colours	Run the Colour Setup utility (DBSETUP)
Programmation	**Device Programmation Menu**
Start	Start device programmation
Abort	Abort device programmation
Verify	Verify device memory
Dump	Dump device memory
Blank	Check device memory is blank

TMS370 Debugger - Configuration and Programmation Commands

Register File Window

The register file window is located on the left, under the CPU registers and displays the contents of 20 of the registers from the register file. This window can be scrolled up and down in order to see different areas of the register file as well as modify the contents of any register shown.

Stack Window

The stack window, located to the right of the register file window, contains the contents of the current program stack. The stack window differs from the register file window in that a) when updated, the stack window automatically changes the display to reflect the offset of each register from the current top of stack, and b) the registers are displayed in reverse order, so that "higher" on the stack (ie. closer to the top-of-stack) corresponds to higher in the window.

Expression Window

The expression window, located in the lower right hand corner of the screen is used to display expressions specified by the user.

Top Level Commands	
Display	Fill the Display Window
Memory	Display any memory location
Pregs	Display peripheral registers
File	Display a file
Symbol	Display current module's symbols
mOdules	Display modules
Clear	Display system revisions
eXecute	Run or Single-step the CPU
Instruction	Single-step instruction
Statement	Single-step statement (special for CALL)
Loop	Single-step once through a loop
While	Single-step while expression is true
Until	Single-step until expression is true
Function	Single-step until RTS or RTI is encountered
Go	Run from current PC
Reset	Software reset and run
tArget reset/run	Wait for target to be reset, then run
Trace/Timer	Inspect trace samples

TMS370 Debugger - Top Level Commands

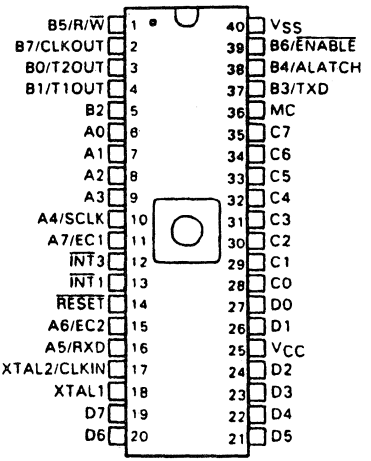
Register	Modify a register
Memory	Modify or fill memory
Modify	Modify a memory location
Fill	Fill a range of memory with a value
Assemble	Invoke symbolic line by line assembler
Breakpoint	Change code breakpoint settings
Add	Add a simple breakpoint
Delete	Delete a simple breakpoint
Remove all	Remove all simple breakpoints
Load	Load simple breakpoints from file
Save	Save simple breakpoints to a file
Evaluate	Evalute an expression
Configure	Configure the debugger (see below)
Trace/Timer	Inspect trace samples
Position	Position trace screen at a specific sample
Top	Position trace screen at top of trace buffer
Bottom	Position trace screen at bottom of trace buffer
Save	Save trace buffer to a file
eXecute	Start the CPU
Load	Load a file
mOdule	Set the current module
Halt	Halt the CPU if running
System	Temporary escape to the operating system
Quit	Exit from the debugger

TMS370 Debugger - Top Level Commands Cont.

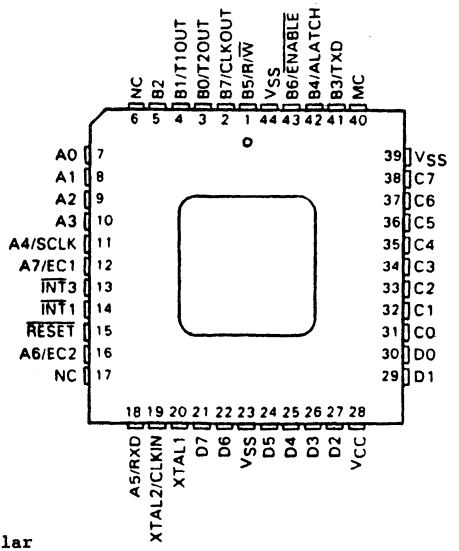
TMS77C82
8 BIT CMOS EPROM MICROCOMPUTER

- Additional member of TMS7000 Family
  - TMS70C82 Prototyping Device
  - Low-Volume Production Support Option
  - Register-to Register Architecture
  - TMS7000 Instruction Set Compatible
  - Eight Powerful Addressing Formats
- CMOS Technology
  - Low Power Modes
- 8K-Byte On-Chip EPROM with Programming Procedure Compatible with TMS27C64
- Operating Range
  - Voltage (Vcc) 3V - 6V
  - Frequency 0.5 MHz to 6.0 MHz
  - Temperature -40°C to 85°C
- Flexible Memory Configuration
  - 256-Byte On-Chip RAM Register File
  - Memory-Mapped ports
  - Memory Expansion to 64K Bytes
- 32 CMOS Compatible I/O Pins
  - 24 Bidirectional Pins
  - 8 Output Pins
- Three On-Chip Timers:
  - Two cascadable 16-Bit Timers with 5-Bit Prescale and 16-Bit Capture Latch
  - One 8-Bit Timer with 2-Bit Prescale
  - Internal Interrupt with Timer Reload
- On-Chip Serial Port
  - Flexible Data Protocols
  - Internal or External Baud Generator
  - Asynchronous, Isoynchronous and Serial I/O Modes
  - Two Multiprocessor Communication Formats
- Flexible Interrupt Handling
  - External Interrupts Programmable for Edge or Edge/Level Triggerring and Rising or Falling Edge Detection
  - Priority Servicing of Simultaneous Interrupts
  - Global and Individual Interrupt Masking
- Development Support
  - Low Cost Evaluation Module
  - Full feature Development System
  - Assembler/Linker Cross Support for Popular Hosts

JD PACKAGE
(TOP VIEW)



FJ PACKAGE
(TOP VIEW)



TEXAS INSTRUMENTS

description

The TMS77C82 is an EPROM version of the 8-bit TMS70C82 microcomputer. The TMS77C82 contains 8K bytes of on-chip EPROM and is completely software and pin compatible with the TMS70C82. Other features include 256 bytes of on-chip RAM, a flexible serial port (UART), three timers programmable sense interrupts, eight addressing formats, and the same advanced register-to-register architecture that allows direct register arithmetic and logical operations without requiring the use of an accumulator (e.g Add R37, R228; add register 37 to register 228 and store the result in register 228).

The TMS77C82 is ideal for low-power applications, and for designs where program constraints are likely to change periodically. The CMOS fabrication, coupled with high performance CPU and internal peripherals, allows flexible system designs in industrial, automotive, computer and telecommunications applications. Other uses of this device include prototyping capabilities for the TMS70C82 and a low-volume alternative to masked ROM parts.

The 16-bit timers, with their associated 5-bit prescale, 16-bit capture latch, and timer outputs, simplify A/D conversions, pulse width measurements and other time-critical application designs. For real-time applications where accuracy over long periods is essential, the Timer 1 output may be cascaded into the Timer 2 inputs to effectively form one 42-bit timer.

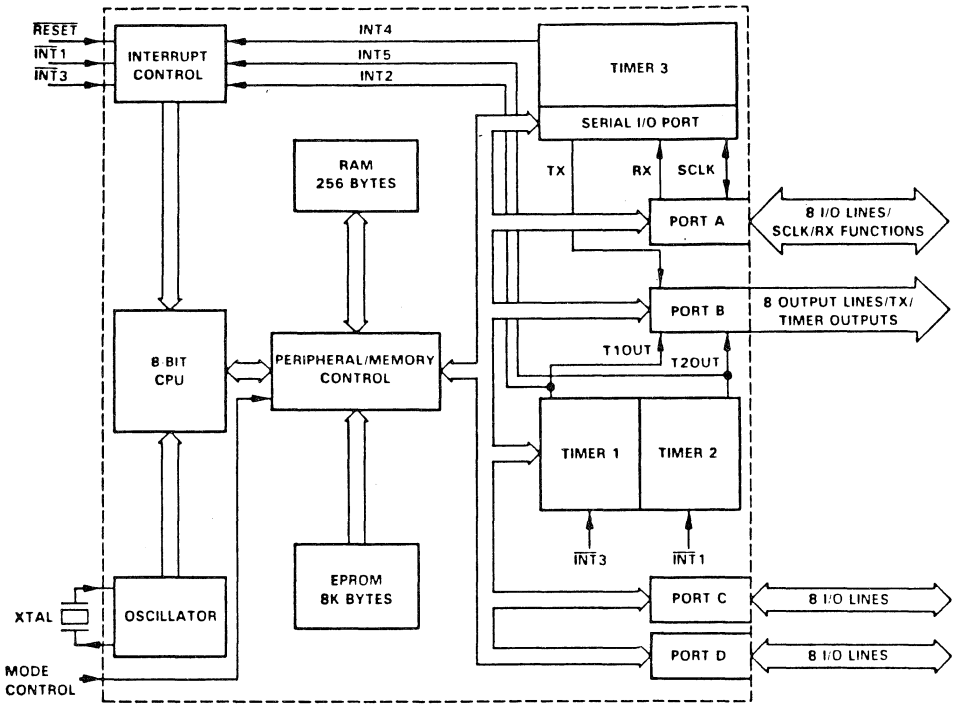
The unique serial port can operate in any one of these three modes: Asynchronous, Isosynchronous, or Serial I/O. Additional features of the serial port include a selectable protocol (data bits, parity, and stop bits). Internal or external baud rate generation and error detection flags. Direct networking for processor-to-processor communications is also supported through two multiprocessor protocols.

There are six prioritised interrupt levels on the TMS77C82. Level 0 is the nonmaskable reset, level 2 is associated with Timer 1, level 4 is associated with the serial port (receive, transmit and Timer 3), and level 5 is generated by Timer 2. Levels 1 and 3 are external interrupts with programmable edge/edge and level triggering, and rising/falling sense detection. All interrupts are routed through a user-defined vector to the appropriate service routine; therefore, each service routine can be located anywhere in the TMS77C82 address space. There is a global interrupt enable bit in the status register as well as individual interrupt enable bits for interrupts 1 through 5. The TMS77C82 can be programmed like any Texas Instruments TMS27C64 on a wide variety of PROM program with the aid of an adaptor socket (see pages 40 and 41 for schematics. Contact your PROM programmer manufacturer or local TI field sales office for programming support. The TMS77C82 also contains an EPROM integrity feature called the R bit, which may be used to disable external access to the EPROM. Once the Rbit has been programmed, the contents of the EPROM cannot be modified and only 1s can be read externally from the EPROM. The only way to modify the R bit protection is to completely erase the contents.

If power consumption is critical, the TMS77C82 can idle selectable sections of the microcomputer(e.g. Timer 1, Timer 2, or UART) and use power only when needed. Also the entire processor can be halted while retaining the 256 bytes of internal RAM.

The TMS77C82 instruction set is identical to that of all TMS7000 family members, allowing easy transition between members.

functional block diagram



pin descriptions

OPERATION MODES					EPROM MODE		
SIGNAL	PIN NO. LCC	PIN NO. DIP	I/O	DESCRIPTION	SIGNAL	I/O	DESCRIPTION
A0 (LSb)	7	6	I/O	A0-A7 are general-purpose bidirectional pins.	A7	I	A3-A7, A12 are address
A1	8	7	I/O		A6	I	lines.
A2	9	8	I/O		A5	I	
A3	10	9	I/O		A4	I	
A4/SCLK	11	10	I/O	Data I/O/Serial port clock	A3	I	
A5/RXD	18	16	I/O	Data I/O/Serial port receiver	A12	I	
A6/EC2	16	15	I/O	Data I/O/Timer 2 event counter	$\overline{PGM}$	I	Program
A7/EC1	12	11	I/O	Data I/O/Timer 1 event counter	$\overline{G}$	I	Output enable
B0/T2OUT	3	3	O	B0-B3 are outputs. B4-B7 are outputs in Single-Chip mode and memory interface pins in			
B1/T1OUT	4	4	O	all other modes. B0 and B1 are outputs for Timer 2 and Timer 1.			
B2	5	5	O				
B3/TXD	41	37	O	Data output/Serial port transmitter			
B4/ALATCH	42	38	O	Data output/Memory interface address latch strobe			
B5/R/$\overline{W}$	1	1	O	Data output/Memory read/write signal			
B6/$\overline{ENABLE}$	43	39	O	Data output/Memory interface enable strobe			
B7/CLKOUT	2	2	O	Data output/Internal clockout			
C0	31	28	I/O	Port C is a bidirectional data port. In	Q1	I/O	Q1-Q8 are bidirectional
C1	32	29	I/O	Microprocessor, Peripheral-Expansion, and	Q2	I/O	data lines.
C2	33	30	I/O	Full-Expansion modes, Port C is a multiplexed	Q3	I/O	
C3	34	31	I/O	low address and data bus.	Q4	I/O	
C4	35	32	I/O		Q5	I/O	
C5	36	33	I/O		Q6	I/O	
C6	37	34	I/O		Q7	I/O	
C7	38	35	I/O		Q8	I/O	
D0	30	27	I/O	Port D is a bidirectional data port. In	A8	I	A0-A2 and
D1	29	26	I/O	Microprocessor and Full-Expansion modes,	A9	I	A8-A11 are
D2	27	24	I/O	it is the high address bus.	A11	I	address lines.
D3	26	23	I/O		A10	I	
D4	25	22	I/O		$\overline{E}$	I	Chip enable
D5	24	21	I/O		A0	I	
D6	22	20	I/O		A1	I	
D7	21	19	I/O		A2	I	
$\overline{INT1}$	14	13	I	Highest priority maskable external interrupt			
$\overline{INT3}$	13	12	I	Lowest priority maskable external interrupt			
$\overline{RESET}$	15	14	I	Reset	GND		V_{SS} for EPROM mode
MC	40	36	I	Mode control pin, V_{CC} for Microprocessor mode	V_{PP}		Program enable 12.5 V to program (0 V to verify)
XTAL2/CLKIN	19	17	I	Crystal input for control of internal oscillator	GND		V_{SS} for EPROM mode
XTAL1	20	18	O	Crystal output for control of internal oscillator			
V_{CC}	28	25		Supply voltage (positive)	V_{CC}		Supply voltage (6 V)
V_{SS}	23, 39, 44	40		Ground reference	GND		Ground reference

architecture

memory modes

The TMS77C82 has four different operating modes, allowing the optimization of the on-chip versus off-chip memory for each application. These modes are Single-Chip, Peripheral Expansion, Full Expansion, and Microprocessor. The tables below show the pin conditions that must be met for each mode, the number of I/O pins, and the amount of external address space available in each of the different modes. To enter the EPROM mode, the $\overline{\text{RESET}}$ and XTAL2 pins must be held low.

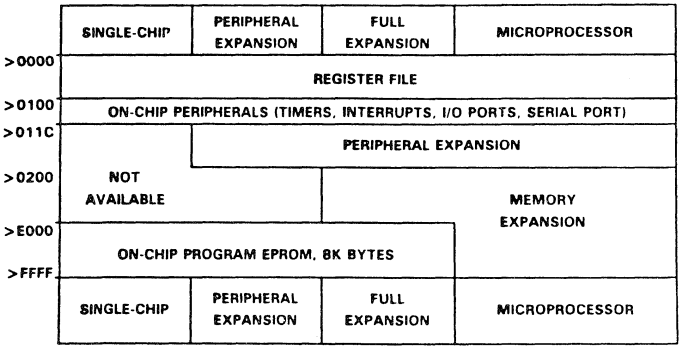
MODE SELECT		OPERATION MODES				EPROM PROGRAMMING MODE	EPROM VERIFY MODE
		SINGLE-CHIP	PERIPHERAL EXPANSION	FULL EXPANSION	MICROPROCESSOR		
I/O CONTROL REGISTER	BIT 7	0	0	1	X	X	X
	BIT 6	0	1	0	X	X	X
MODE CONTROL PIN		V_{SS}	V_{SS}	V_{SS}	V_{CC}	V_{PP}	V_{SS}
$\overline{\text{RESET}}$ PIN		V_{CC}	V_{CC}	V_{CC}	V_{CC}	V_{SS}	V_{SS}
XTAL2 PIN		N/A	N/A	N/A	N/A	V_{SS}	V_{SS}

X — Don't care
N/A — Not applicable

	SINGLE-CHIP	PERIPHERAL EXPANSION	FULL EXPANSION	MICROPROCESSOR
I/O Pins:				
Bidirectional	24	16	8	8
Output only	8	4	4	4
Expansion Bus:				
Address only lines	0	0	8	8
Multiplexed Address/Data lines	0	8	8	8
Control lines	0	4	4	4
Memory Space:				
RAM	256	256	256	256
EPROM†	8192	8192	8192	0
Internal Peripheral File	28	25	23	23
External Peripheral File	0	231	233	233
External Memory	0	0	56832	65024

†The first six bytes of masked ROM are reserved for TI internal use.

memory map


SINGLE-CHIP
PERIPHERAL EXPANSION
FULL EXPANSION
MICROPROCESSOR
>0000
REGISTER FILE
>0100
ON-CHIP PERIPHERALS (TIMERS, INTERRUPTS, I/O PORTS, SERIAL PORT)
>011C
PERIPHERAL EXPANSION
>0200
NOT AVAILABLE
MEMORY EXPANSION
>E000
ON-CHIP PROGRAM EPROM, 8K BYTES
>FFFF
SINGLE-CHIP
PERIPHERAL EXPANSION
FULL EXPANSION
MICROPROCESSOR

peripheral memory map

REGISTER	ADDRESS	NAME	NOTE	FUNCTION
P0	>0100	IOCNT0	3	Interrupts 1, 2, and 3, expansion mode control
P1	>0101	IOCNT2		Polarity and edge/level control for $\overline{INT1}$ and $\overline{INT3}$
P2	>0102	IOCNT1	3	Interrupts 4 and 5
P3	>0103	–		Reserved
P4	>0104	APORT		A port data value
P5	>0105	ADDR		A port direction register
P6	>0106	BPORT	1	B port data value
P7	>0107	–		Reserved
P8	>0108	CPORT	1	C port data value
P9	>0109	CDDR	1	C port direction register
P10	>010A	DPORT	2	D port data value
P11	>010B	DDDR	2	D port direction register
P12	>010C	T1MSDATA	3	Timer 1 MSB reload register/MSB readout latch
P13	>010D	T1LSDATA	3	Timer 1 LSB reload register/LSB decrementer value
P14	>010E	T1CTL1	3	Timer 1 control register 1/MSB readout latch
P15	>010F	T1CTL0	3	Timer 1 control register 0/LSB capture latch value
P16	>0110	T2MSDATA	3	Timer 2 MSB reload register/MSB readout latch
P17	>0111	T2LSDATA	3	Timer 2 LSB reload register/LSB decrementer value
P18	>0112	T2CTL1	3	Timer 2 control register 1/MSB readout latch
P19	>0113	T2CTL0	3	Timer 2 control register 0/LSB capture latch value
P20	>0114	SMODE		Serial port mode control register
P21	>0115	SCTL0		Serial port control register 0
P22	>0116	SSTAT		Serial port status register
P23	>0117	T3DATA	3	Timer 3 reload register/decrementer value
P24	>0118	SCTL1		Serial port control register 1
P25	>0119	RXBUF		Receiver buffer
P26	>011A	TXBUF		Transmitter buffer
P27	>011B	–		Reserved
P28-P255	>011C >01FF	–		Peripheral expansion

NOTES 1. P8, P9, and the most significant nibble of P6 become off-chip in Peripheral Expansion, Full Expansion, and Microprocessor modes
2. P10 and P11 become off-chip in Full Expansion and Microprocessor modes. All other addresses between P0 and P27 inclusive remain on-chip in all expansion modes.
3. Exercise caution when using logical instructions (e.g., ANDP, ORP, XORP) on these registers because of the different read/write functions.

Interrupt priorities

The TMS77C82 has five interrupt levels plus $\overline{RESET}$. These levels are defined as follows:

Level 0: $\overline{RESET}$ (highest priority)
Level 1 ($\overline{INT1}$): External, user-defined, software programmable control over edge/level triggering and polarity
Level 2 (INT2): Timer 1
Level 3 ($\overline{INT3}$): External, user-defined, software programmable contol over edge/triggering and polarity
Level 4 (INT4): Serial port TX ready, RX full, or Timer 3
Level 5 (INT5): Timer 2

device initialization

Interrupt level 0 ($\overline{\text{RESET}}$) cannot be masked and will be recognized immediately, even in the middle of an instruction. To execute the level 0 interrupt, the $\overline{\text{RESET}}$ pin must be held low for a minimum of 1.25 internal clock cycles ($t_{c(C)}$) to guarantee recognition by the device. During assertion of the $\overline{\text{RESET}}$ pin, the following conditions for the indicated locations occur.

PF LOCATIONS	LOCATION	RESET RESULT
P5, P9, P11	Data Direction Registers	Set to all 0s (ports are inputs)
P4, P8, P10	Port A, C, D Output Data Flip Flops	Not affected
P6	Port B Output Data Flip Flops	Set to all 1s
P0, P1	IOCNT0, IOCNT2	Set to all 0s (Bits 7, 6, 3, 2, 0, and IOCNT2 are indeterminate) NOTE: INT1FLG-INT3FLG are cleared
P2	IOCNT1	Bits 3, 2, 0 set to 0 Bits 7, 6, 5, 4, 1 not affected NOTE: INT4FLG is not cleared NOTE: INT5FLG is cleared
P21	SCTL0	Bits 7, 2, 1, 0 set to 0 Bit 6 set to 1 Bits 5, 4, 3 not affected
P22	SSTAT	Bits 6, 1 set to 0 Bits 2, 0 set to 1 Bits 7, 5, 4, 3 not affected
P24	SCTL1	Bits 6, 5, 4, 3, 2 set to 0 Bits 7, 1, 0 not affected
P14	T1CTL1	Bit 6 set to 0 All others not affected
P18	T2CTL1	Bits 7, 6 set to 0 All others not affected

CPU REGISTERS	RESET RESULT
Status Register	Cleared
Stack Pointer	Loaded with >01
Program Counter	Old MSB, LSB loaded into Register A and B PC loaded with reset vector

I/O control registers

The I/O control registers are located in the Peripheral File and are responsible for memory mode definition and interrupt control. In the following figures, each bit in the I/O control registers is defined.

The INTn FLAG values are independent of the INTn ENABLE values. Writing a 1 to the INTn ENABLE will not clear the INTn FLAG. Writing a 1 to the INTn CLEAR bit will clear the corresponding INTn FLAG, but writing a 0 to the INTn CLEAR bit has no effect on the bit.

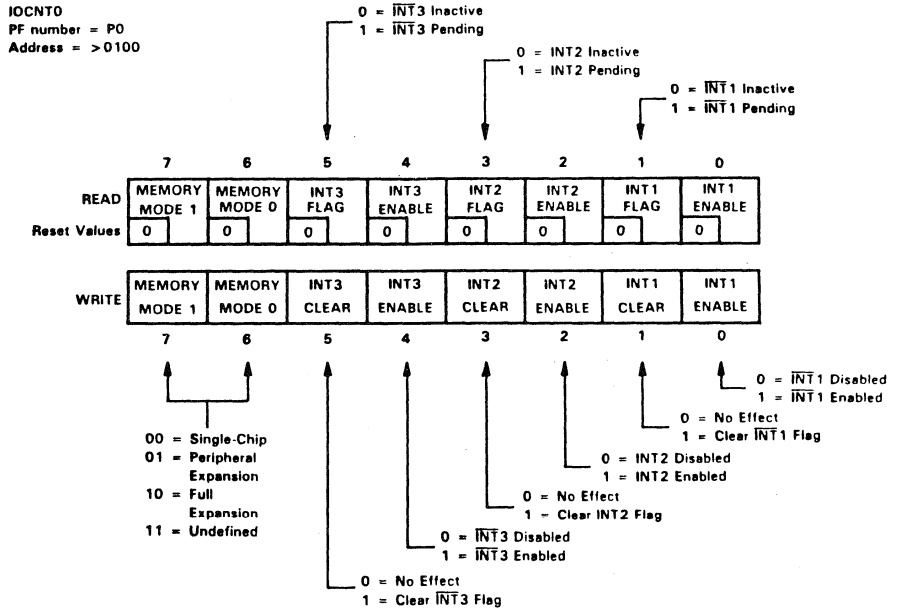


FIGURE 1. I/O CONTROL REGISTER 0 (IOCNT0)

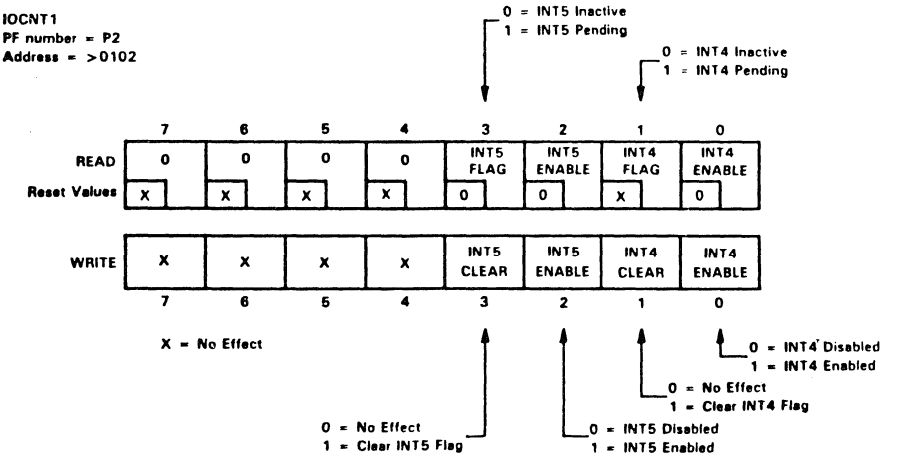
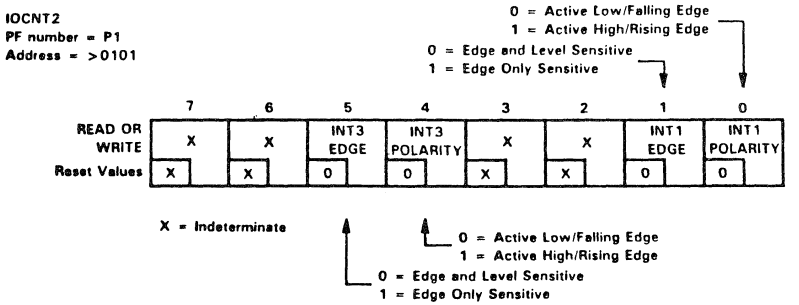


FIGURE 2. I/O CONTROL REGISTER 1 (IOCNT1)



NOTE 4: When changing the sense of $\overline{INT1}$ or $\overline{INT3}$, the interrupt will become active (set the FLAG bit and interrupt the CPU if enabled) if the level present at the interrupt pin corresponds with the new sense selected. Also, the corresponding capture latch will be loaded.

FIGURE 3. I/O CONTROL REGISTER 2 (IOCNT2)

programmable timer/event counter

The TMS77C82 features three on-chip timers with individual start/stop control bits. Timer 1 (shown in Figure 4) and Timer 2 (shown in Figure 7) consist of a 16-bit readable decrementer with a 16-bit reload register, a 16-bit capture latch, and a 5-bit prescaler with a 5-bit reload register. Timer 3 consists of an 8-bit readable decrementer with an 8-bit reload register and a 2-bit prescaler with a 2-bit reload register. Timer 3 can be used as a general-purpose timer or as a baud rate generator for the serial port.

most significant byte readout latch

This latch is shared between the most significant byte (MSB) of the decrementer and the MSB of the capture latch. it allows the complete 16-bit value of the decrementer or the capture latch to be sampled at one moment. The least significant byte (LSB) must be read first, which causes the MSB to be simultaneously loaded into the readout latch.

There is only one readout latch for each timer, but the same latch can be read from two addresses for easier programming (see the diagrams for Timer 1 and Timer 2).

Timer 1 MSB readout latch can be read from both P12 (>010C) and P14 (>010E). Similarly, Timer 2 MSB readout latch can be read from both P16 (>0110) and P18 (>0112).

Reading the LSB of the decrementer or capture latch will always update the contents of the readout latch. In order to correctly read the entire 16-bit value of the decrementer or capture latch, the LSB must be read first, which will load the MSB readout latch. The MSB readout latch must be read and stored before reading the LSB of either the decrementer or capture latch.

The order of 16-bit read operations should be:

Timer 1:
Decrementer: P13 then P12, or P13 then P14
Capture latch: P15 then P12, or P15 then P14

Timer 2:
Decrementer: P17 then P16, or P17 then P18
Capture latch: P19 then P16, or P19 then P18

timer 1 schematic diagram

A schematic diagram of Timer 1 is shown below. For clarity the details of the clock source selection and the power reduction mechanism are covered in Figure 6, Timer 1 Control Registers.

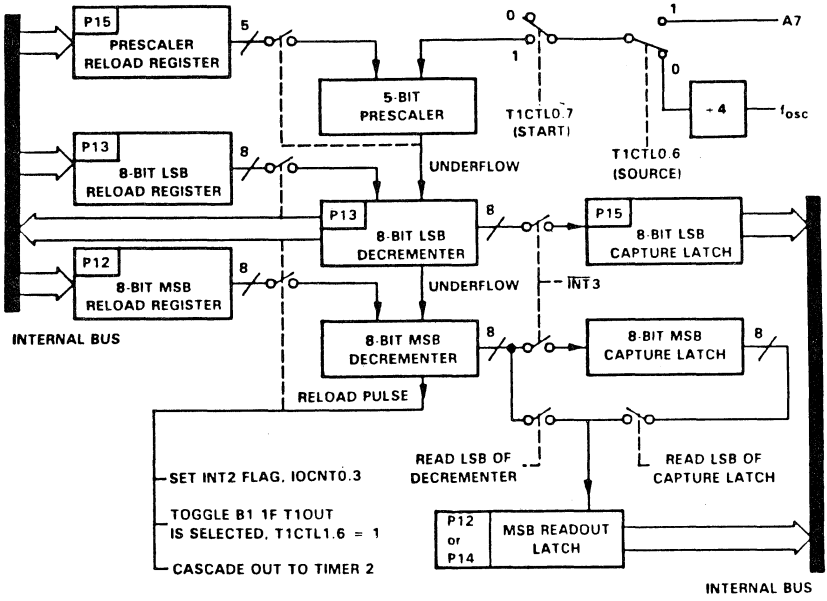


FIGURE 4. TIMER 1 SCHEMATIC DIAGRAM

timer 1 control registers

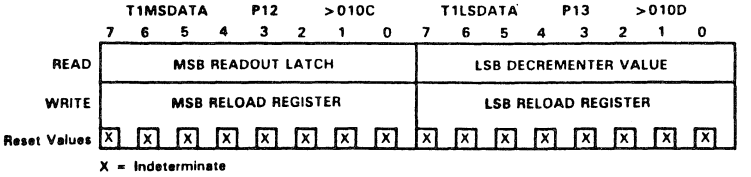


FIGURE 5. TIMER 1 DATA REGISTERS

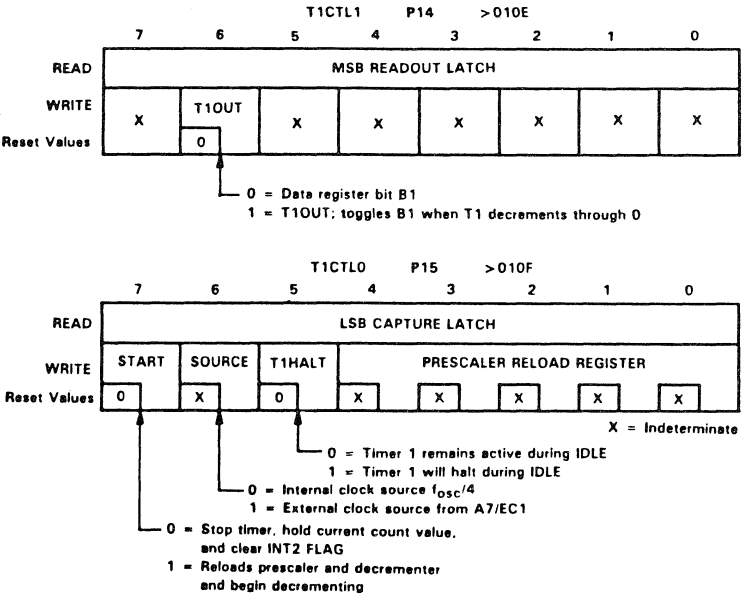


FIGURE 6. TIMER 1 CONTROL REGISTERS

timer 2 schematic diagram

A schematic diagram of Timer 2 is shown below. For clarity the details of the clock source selection and the power reduction mechanism are covered in Figure 9, Timer 2 Control Registers.

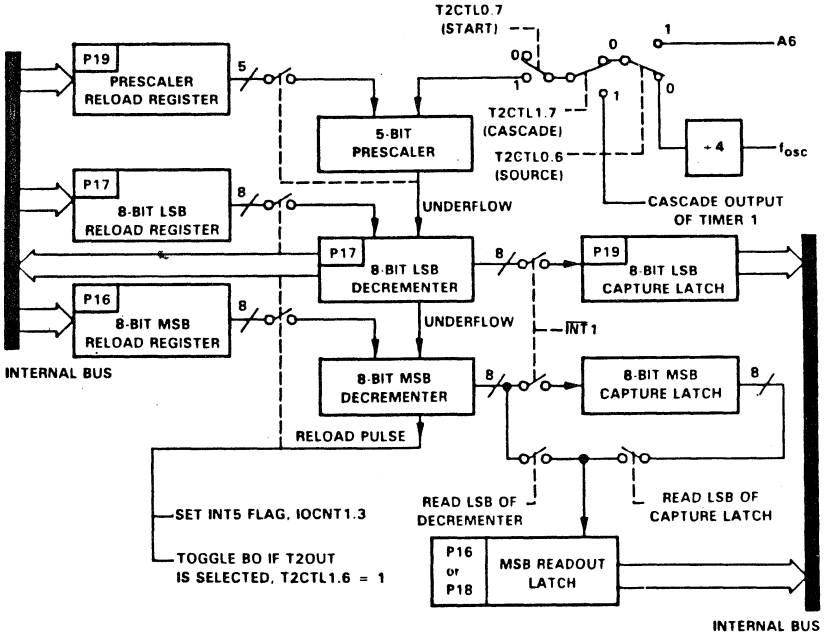


FIGURE 7. TIMER 2 SCHEMATIC DIAGRAM

timer 2 control registers

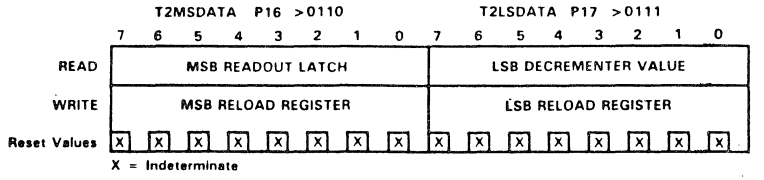
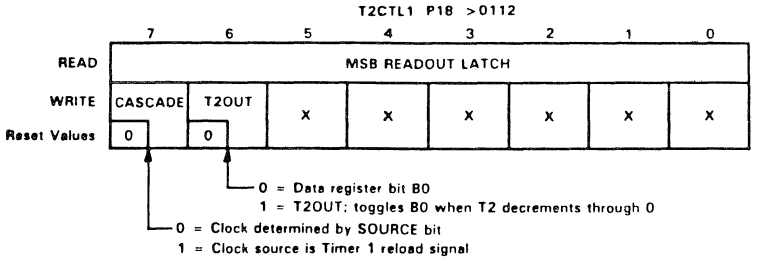


FIGURE 8. TIMER 2 DATA REGISTERS



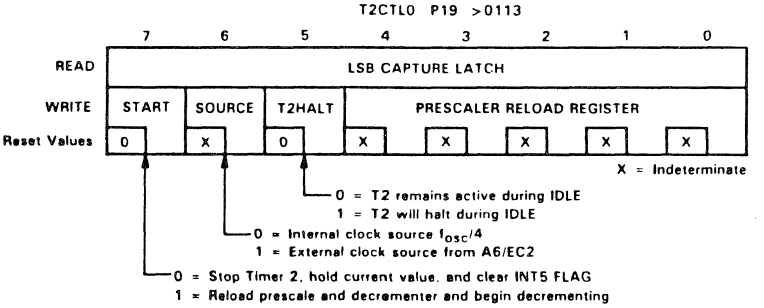


FIGURE 9. TIMER 2 CONTROL REGISTERS

timer 1 and timer 2 clock source

TIMER 1 CLOCK SOURCES

T1CTL0 BIT 6 (SOURCE)	CLOCK SOURCE	MODE
0	$f_{osc}/4$	RTC (Real Time Clock)
1	A7, External	EC (Event Counter)

TIMER 2 CLOCK SOURCES

T2CTL0 BIT 6 (SOURCE)	T2CTL1 BIT 7 (CASCADE)	CLOCK SOURCE	MODE
0	0	$f_{osc}/4$	RTC
1	0	A6, External	EC
X	1	Reload Signal of Timer 1	CASCADE

Bit 7 of timer control registers T1CTL0 and T2CTL0 is the START bit for Timer 1 and Timer 2, respectively. When a 0 is written to the START bit, the timer chain is disabled and frozen at the current count value, and Timer 1's INT2 FLAG or Timer 2's INT5 FLAG is set to 0. When a 1 is written to the START bit, regardless of whether it was a 0 or a 1 before, the prescaler and counter decrementers are loaded with the corresponding latch values, and the Timer/Event Counter operation begins.

When the prescaler and counter decrement through zero together, an interrupt flag is set and the prescaler and counter decrementers are immediately and automatically reloaded with the corresponding values from the reload registers, and counting continues. The interrupts generated by the timers are INT2 for Timer 1 and INT5 for Timer 2.

Timers 1 and 2 each have a 16-bit capture latch which "captures" the current value of the counter whenever the appropriate input capture signal is generated. The capture latch values for Timer 1 and Timer 2 are loaded on the active edges of $\overline{INT3}$ and $\overline{INT1}$, repsectively, whether or not the interrupts are enabled. Both capture latches are disabled during the IDLE instruction when their corresponding timer HALT bits are 1.

event counter (EC)

When Timer 1 or Timer 2 is in the EC mode, pins A7 and A6 are the decrementer clock sources for Timer 1 and Timer 2, respectively. The maximum clock frequency on A7 or A6 in the EC mode must not be greater than $f_{osc}/4$. The minimum pulse width must not be less than 1.25 machine cycles ($t_{c(C)}$). Each positive pulse transition decrements the count chain.

timer output function

A timer output function exists on both Timer 1 and Timer 2 that allows the B1 and B0 outputs, respectively, to be toggled every time the timer decrements through zero. This function is enabled by the T1OUT bit and T2OUT bit (bit 6) in timer control registers T1CTL1 and T2CTL1.

When operating in the timer output mode, the B0 and/or B1 output cannot be changed by writing to the B port data register. Writing to the respective timer's START bit will reload and start the timer, but will not toggle the output. The output will toggle only when the timer decrements through zero. The timer output feature is independent of INT2 and INT5 and, therefore, will operate with INT2 and INT5 enabled or disabled. Also, if the timer is active during the IDLE instruction, the timer output feature will continue to operate.

Whenever the T2OUT or T1OUT bit is returned to 0, B0 or B1 will become an output-only pin, like B2. The value in the B0 or B1 data register will be the last value output by the timer output function, so that B0 or B1 will not change as the T2OUT or T1OUT bit is returned to 0.

Whenever a read of BPORT is performed, the values on the B0 pin and B1 pin will always be returned, so the current timer output values can be read by reading BPORT.

The T1OUT and T2OUT bits are set to 0 by $\overline{\text{RESET}}$, so the timer output function will not be enabled unless the user sets the T1OUT or T2OUT bit to 1.

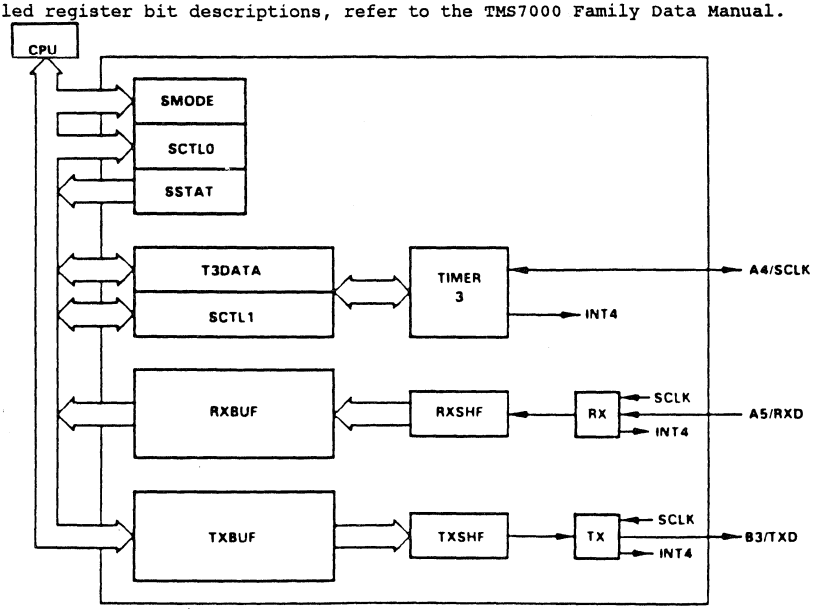
The Timer 2 output (T2OUT) cannot be used if Timer 1 and Timer 2 are cascaded together (CASCADE bit of T2CTL1 set to 1).

serial port

The TMS77C82 contains a serial port which greatly enhances its I/O and communication capability. The serial port can operate in several modes which permit the TMS77C82 to interface with Universal Asynchronous Receiver, Transmitter (uart) peripheral devices as well as several microcomputers (e.g. TMS77C82, TMS70C82, 6801, 8051). The serial port consists of a receiver (RX), transmitter (TX) and baud rate generator (Timer 3, T3). It is controlled and accesses through the following registers in the peripheral file:

REGISTER	ADDRESS	NAME	TYPE	FUNCTIONS
P20	>0114	SMODE	R/W	Serial Port Mode
P21	>0115	SCTLO	R/W	Serial Port Control O
P22	>0116	SSTAT	READ	Serial Port Status
P23	>0117	T3DATA	R/W	Timer 3 Data
P24	>0118	SCTL1	R/W	Serial Port Control 1
P25	>0119	RXBUF	READ	Receiver Buffer
P26	>011A	TXBUF	WRITE	Transmission Buffer

For detailed register bit descriptions, refer to the TMS7000 Family Data Manual.



Note 5: The INT4 sources are effectively wire-ORed together to generate only one INT4 input. The SCLK sources are wired together to generate only one SCLK input.

FIGURE 10. SERIAL PORT FUNCTIONAL BLOCKS

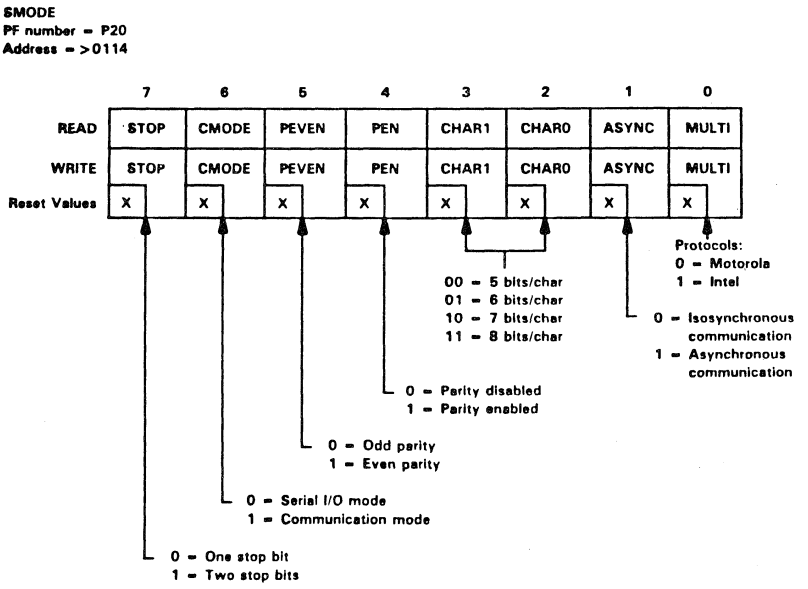


FIGURE 11. SERIAL PORT MODE (SMODE)

SMODE is the RX/TX control register that describes the character format and type of communication mode.

SCTL0
PF number = P21
Address = >0115

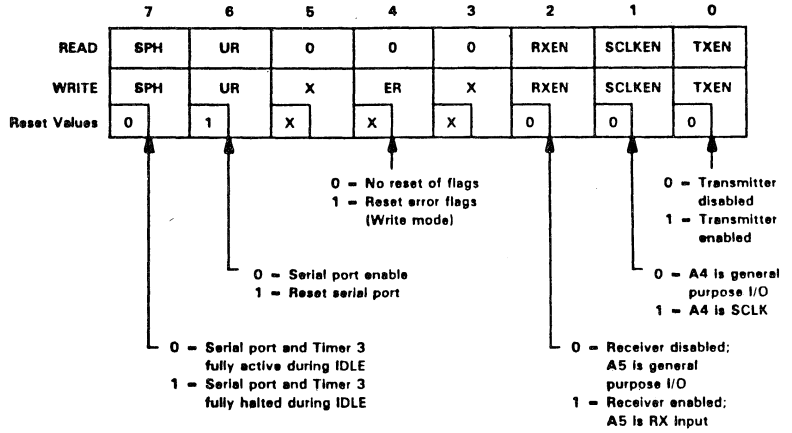


FIGURE 12. SERIAL PORT CONTROL REGISTER 0 (SCTL0)

SCTL0 is RX/TX control register used to control the serial port functions such as TX and RX enable, clearing of error flags, and software enable.

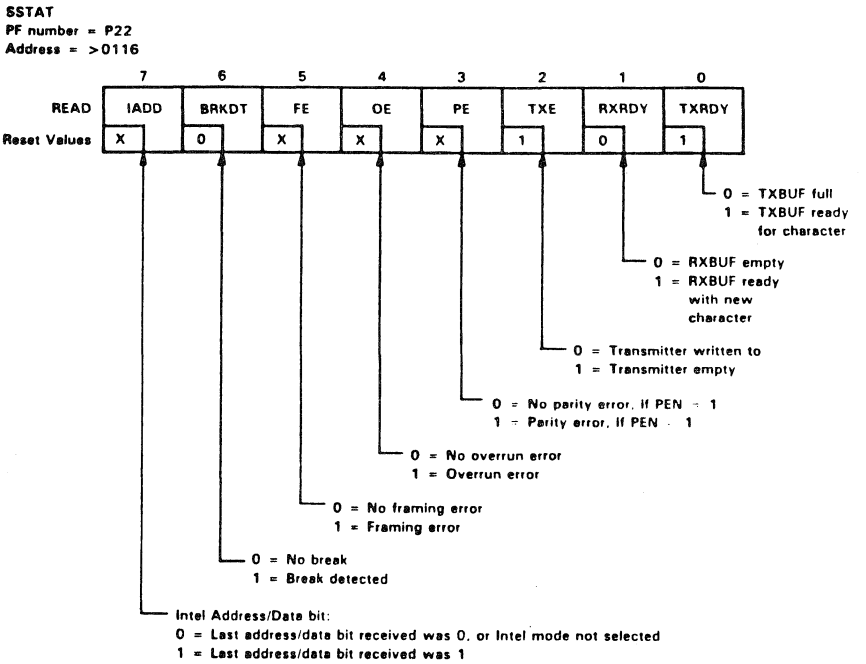


FIGURE 13. SERIAL PORT STATUS (SSTAT)

SSTAT is the read-only register used to report the status of the serial port. Bit 7 (IADD) stores the value of the last address/data bit received when using the Intel multiprocessor mode.

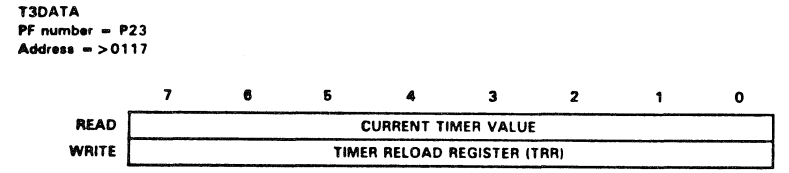


FIGURE 14. TIMER 3 DATA REGISTER (T3DATA)

SCTL1
PF number = P24
Address = >0118

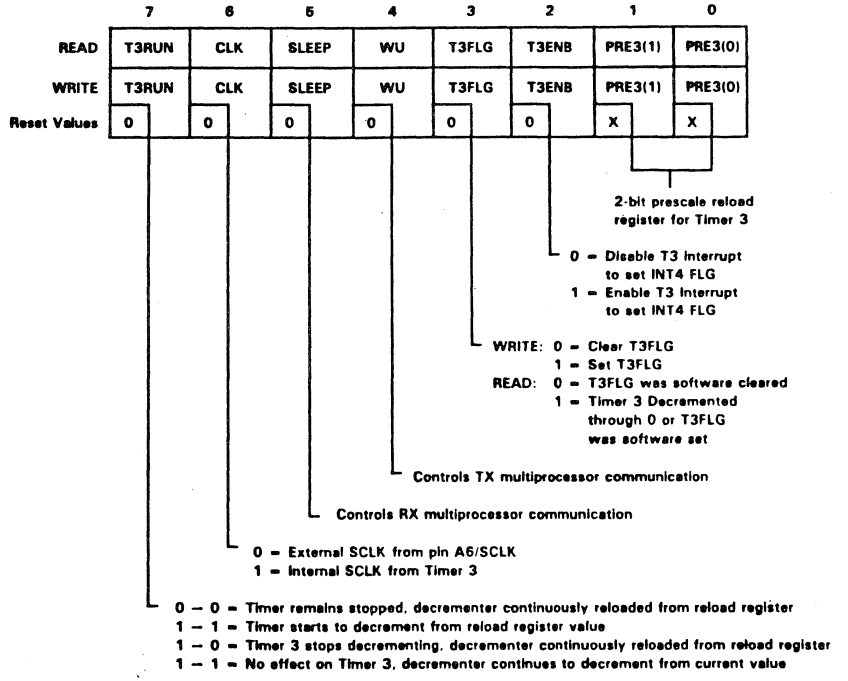


FIGURE 15. SERIAL PORT CONTROL REGISTER (SCTL1)

For a description of the individual timer bits in SCTL1, see the Timer 3 section of this document.

RXBUF
PF number = P25
Address = >0119

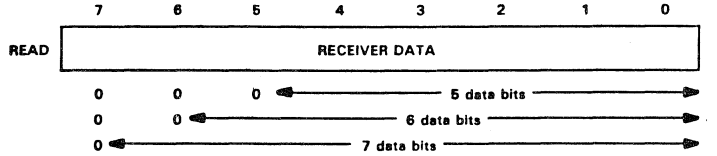
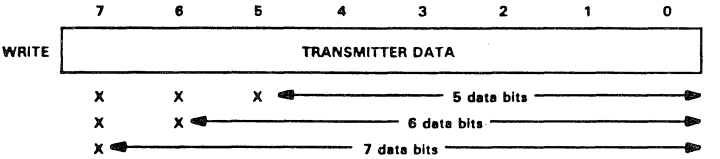


FIGURE 16. RECEIVER BUFFER (RXBUF)

TXBUF
PF number = P26
Address = >011A



X = Ignored

FIGURE 17. TRANSMITTER BUFFER (TXBUF)

serial port clock sources

The serial port can be clocked by Timer 3 or an external baud rate generator. The source of the serial clock (SCLK) is determined by the CLK bit (SCTL1 bit 6) and the SCLKEN bit (SCTL0 bit 1).

SCLKEN	CLK	Serial Port Clock Operation
1	1	A4 is forced to output mode, independent of the data direction register (P5). Timer 3 provides the clock for the Serial Port which is output as SCLK on A4.
1	0	A4 is forced to input mode, independent of the data direction register (P5). An external signal applied to A4 provides the baud rate clock for the Serial Port.
0	1	A4 is available for general-purpose I/O. The clock for the serial port is provided by Timer 3 but is not output on any pin.
0	0	A4 is selected as general-purpose I/O with its direction register controlling the direction of A4. The serial port clock is taken from the A4 pin, so the clock can be provided by an external signal if the pin is in input mode (the same as the SCLKEN = 1, CLK = 0 option above), or by software if the pin is in output mode by writing to the A4 data register.

If SCLKEN is changed from 1 to 0, A4 will have the direction selected by the A port direction register.

In any of these modes, reading from A4 will return the value present at the pin. SCLKEN and CLK are both set to 0 by $\overline{\text{RESET}}$. The A4 direction register is also set to 0 (input) by $\overline{\text{RESET}}$.

timer 3

Timer 3 can be used as a general-purpose timer or as the clock generator for the serial port. Timer 3 is accessed through T3DATA and SCTL1. The Timer 3 clock source is an internal signal with the frequency equal to $f_{osc}/4$. Timer 3 consists of a 2-bit rescaler and an 8-bit counter. These are automatically reloaded from a 2-bit and an 8-bit reload register, respectively, whenever a register decrements through zero.

Timer 3 is continuously reloaded with the prescaler and decrementer reload register values while the T3RUN bit is 0. Timer 3 differs from Timer 1 and Timer 2 in that Timer 3 cannot be held at the value it contained when the T3RUN bit goes to 0. The timer begins decrementing when the T3RUN bit is changed from 0 to 1. The T3RUN bit is initialized to 0 by reset.

Each time the timer decrements through zero, the Timer 3 flag is set to 1 and the INT4 FLAG is set to 1 if T3ENB (SCTL1 bit 2) is 1. Timer 3 and its flags are not affected by the serial port software reset (UR). Therefore, Timer 3 can be used independent of the serial port.

When using Timer 3 as the serial port clock source, the reload pulse (timer decremented through zero) output of Timer 3 goes to the serial port via a divide-by-two circuit, producing an equal mark-space ratio internal SCLK (see Timer 3 block diagram). The baud rate generated by Timer 3 is user-programmable and is determined by the value of the 2-bit prescaler and the 8-bit timer reload registers.

The equations for determining the output baud rates for both the asynchronous and isosynchronous modes are:

$$\text{Asynchronous Baud Rate} = \frac{f_{osc}}{64(PRR+1)(TRR+1)} = \frac{SCLK}{8}$$

$$\text{Isosynchronous Baud Rate} = \frac{f_{osc}}{8(PRR+1)(TRR+1)} = SCLK$$

where: f_{osc} = frequency of the crystal or external system clock
TRR = Timer 3 decrementer reload register (P23)
PRR = Timer 3 prescale reload register (P24)
SCLK = Serial clock either input or output from the SCLK pin

The baud rate for the serial I/O mode is determined with the same equation used to determine the isosynchronous baud rate.

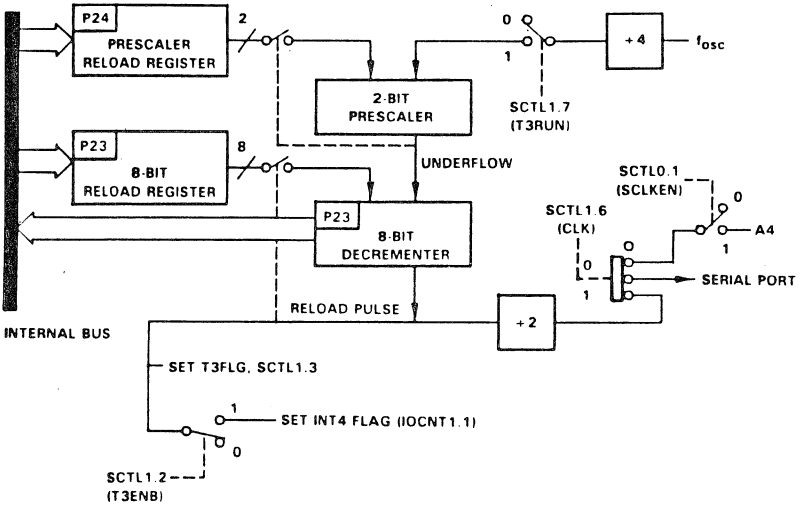


FIGURE 18. TIMER 3 SCHEMATIC DIAGRAM

serial port initialization and reset

After a system reset, the UR bit in SCTL0 will be set to 1 and the serial port will be held in its reset condition. The serial port registers are then set in the order shown below. A serial port reset can be performed by writing a 1 to the UR bit (SCTL0 bit 6) or by writing data to the SMODE register at any time. Whenever a write to the SMODE register is performed, the UR bit is set to 1 and the serial port will be reset. The data written to SMODE will become the new SMODE register contents.

SERIAL PORT INITIALIZATION

SET B3 DATA = 1
WRITE TO SMODE
WRITE TO SCTL0 (SET BIT 6 TO 0)
WRITE TO SCTL1

The BPORT pin 3 must be set to 1 to transmit.

UART reset by software

Setting the UR bit (SCTL0 bit 6) to 1 affects the following:

SCTL0	Bits 7, 2, 1, 0 set to 0 Bit 6 set to 1 Bits 5, 4, 3 not affected
SSTAT	Bits 6, 1 set to 0 Bits 2, 0 set to 1 Bits 7, 5, 4, 3 not affected
SCTL1	Bits 6, 5, 4 set to 0 Bits 7, 3, 2, 1 0 not affected
Pin 37 (B3/TXD)	Outputs the value of bit 3 of P6
Pin 10 (A4/SCLK)	Configures to the corresponding values stored in P4 and P5
Pin 16 (A5/RXD)	Configures to the corresponding values stored in P4 and P5

You cannot write to the affected bits of SCTL0 and SCTL1 while the UR bit is set to 1. The configuration of all bits in SCTL0 can be written to with a single instruction as long as the value of bit 6 (UR) within that instruction is 0.

software example

The following software example initializes the TMS77C82.

```
*************************************************************************
*       R E S E T    A N D    I N I T I A L I Z A T I O N
*************************************************************************
*
*                              E Q U A T E   T A B L E
*
APORT   EQU   P4              Port A Data Register
BPORT   EQU   P6              Port B Data Register
CPORT   EQU   P8              Port C Data Register
DPORT   EQU   P10             Port D Data Register
ADDR    EQU   P5              Port A Data Direction Register
CDDR    EQU   P9              Port C Data Direction Register
DDDR    EQU   P11             Port D Data Direction Register
*
*                              I/O CONTROL REGISTERS
*
IOCNT0 EQU    P0              Interrupts 1,2,3 and E pansion mode
IOCNT1 EQU    P2              Interrupts 4,5
IOCNT2 EQU    P1              Polarity and level control for external ints.
*
*                              TIMER 1 REGISTERS
*
T1CTL0 EQU    P15             Timer 1 control register 0
T1CTL1 EQU    P14             Timer 1 control register 1
T1LSDA EQU    P13             Timer 1 LSB reload Register
T1MSDA EQU    P12             Timer 1 MSB reload Register
PRESC1 EQU    >00             Timer 1 Prescale value for 10 ms
T10MSL EQU    >97             Timer 1 MSB value for 10 ms
T10MSM EQU    >3A             Timer 1 LSB value for 10 ms
```

```
*                                   TIMER 2 REGISTERS
*
T2CTL0  EQU    P19                  Timer 2 control register 0
T2CTL1  EQU    P18                  Timer 2 control register 1
T2LSDA  EQU    P17                  Timer 2 LSB reload Register
T2MSDA  EQU    P16                  Timer 2 MSB reload Register
PRESC2  EQU     0                   Timer 2 Prescale value for 10 KHz
T10KHL  EQU    74                   Timer 2 MSB value for 10 KHz
T10KHM  EQU     0                   Timer 2 LSB value for 10 KHz
*
*                                   UART REGISTERS
*
T3DATA  EQU    P23                  Timer 3 reload register
SMODE   EQU    P20                  Serial port Mode Control Register
SCTL0   EQU    P21                  Serial port Control Register 0
SSTAT   EQU    P22                  Serial port Status Register
SCTL1   EQU    P24                  Serial port Control Register 1
RXBUF   EQU    P25                  Receiver buffer
TXBUF   EQU    P26                  Transmitter buffer
B300    EQU    103                  Count for 300 baud at 6 MHz
*-------------------------------------------------------------------------
*
*       T H I S   I S   T H E   R E S E T   E N T R Y   P O I N T
*
START   DINT                        Disable interrupts
        MOVP   %?01000000,SCTL1     Halt timer 3, sleep & WU bits off
*
        MOV    %100,B               Set stack pointer to R100
        LDSP
*
*                                   I N T E R R U P T S
*
        MOVP   %?00101110,IOCNT0    Set Mode, enable int 2, clear all flags
        MOVP   %?00001111,IOCNT1    Enable interrupts 4,5 ; clear all flags
        MOVP   %?00000000,IOCNT2    Set INTs sense; type = level
*
*                                   P O R T   D A T A
*
        MOVP   %?00001000,APORT     Initialize Port A
        MOVP   %?10111100,BPORT     Initialize Port B (B3=1 for UART operation)
        MOVP   %?10101011,CPORT     Initialize Port C
        MOVP   %?00001001,DPORT     Initialize Port D
        MOVP   %?10001110,ADDR      Initialize Port A
        MOVP   %?10111111,CDDR      Initialize Port C directions registers
        MOVP   %?01001001,DDDR      Initialize Port D directions registers
*
*                                   T I M E R   1
*
*       Timer 1 is configured as an internal real time clock which
*       interrupts the CPU every 10 ms (at 6 MHz) on interrupt level 2
*
        MOVP   %?00000000,T1CTL1    Timer 1 output pin turned off
        MOVP   %T10MSL,T1LSDA       LSB timer latch value for 10 ms
        MOVP   %T10MSM,T1MSDA       MSB timer latch value for 10 ms
        MOVP   %>80+PRESC1,T1CTL0   start bit + prescale value set
```

```
*
*                              T I M E R    2
*
*       Timer 2 will output a square wave with a 10 KHz frequency
*       The square wave will output on Port B pin 0.  Once configured,
*       Timer 2 will run continuously, without futher CPU servicing.
*
        MOVP   %?01000000,T2CTL1  Timer 2 output pin turned on
        MOVP   %T10KHL,T2LSDA     LSB timer latch value for 10 KHz
        MOVP   %T10KHM,T2MSDA     MSB timer latch value for 10 KHz
        MOVP   %>80+PRESC2,T2CTL0 Start bit + prescale value set
*
*                              S E R I A L    P O R T
*
        MOVP   %B300,T3DATA       Configure UART at 300 baud
        MOVP   %?01111110,$MODE   data bits =8,Stop=1,Parity=even,Protocol=moter
        MOVP   %?00010100,SCTL0   Reset errors, enable RX,disable sclk and TX
        MOVP   %?11000010,SCTL1   Start UART, CLK=int, prescale=2;sleep & WU=off
*
        EINT                      LET THE INTERRUPTS IN NOW
*
*       E N D    O F    I N I T I A L I Z A T I O N
```

power reduction modes

The TMS77C82 supports the Wake-up and Halt modes for low power consumption. These low power modes are entered via execution of the IDLE instruction. The power reduction mechanisms for each timer and the UART are completely independent of each other and are selected individually via the T1HALT, T2HALT, and SPH bits, located in the T1CTL0, T2CLT0, and SCTL0 registers, respectively.

The Wake-up mode is entered when the IDLE instruction is executed. If the T1HALT, T2HALT, or SPH bits are set, then the corresponding timer or UART will be disabled. The Wake-up mode is exited by assertion of an enabled interrupt to the CPU or $\overline{\text{RESET}}$.

The HALT modes are entered when the IDLE instruction is executed and the T1HALT, T2HALT, and SPH bits are set, causing the corresponding timers and UART to be disabled. The oscillator will be active or inactive depending on which mask option is chosen. The HALT OSC-ON mode is exited by assertion of an enabled external interrupt or $\overline{\text{RESET}}$, while the HALT OSC-OFF mode is exited only by $\overline{\text{RESET}}$.

MODE	FUNCTIONAL BLOCK STATUS			ENTER MODE	EXIT† MODE
	CPU	T1, T2, T3-UART	OSC	VIA	VIA
Wake-up	Halted	T1, T2, and T3-UART individually programmed as fully active or halted	Active	IDLE	$\overline{\text{RESET}}$ $\overline{\text{INT}}1$ INT2‡ $\overline{\text{INT}}3$ INT4‡ INT5‡
HALT OSC-ON§	Halted	T1,T2, and T3-UART are all halted	Active	IDLE	$\overline{\text{RESET}}$ $\overline{\text{INT}}1$ $\overline{\text{INT}}3$
HALT OSC-OFF§	Halted	T1, T2, and T3-UART are all halted	Inactive	IDLE	$\overline{\text{RESET}}$

†Interrupts must be enabled to exit.
‡May be used to exit wake-up mode if not programmed to be inactive during IDLE.
§Mask Options. Not programmable.

I/O port operation during power reduction

The following table indicates the state of the memory expansion ports (B, C, and D) in the low power mode, for each of the expansion modes. All of the other I/O pins will maintain their current direction (input or output) and will continue to output the same data if in the output mode. No outputs become three-state when Wake-up mode is entered.

I/O PIN	SINGLE-CHIP	PERIPHERAL EXPANSION	FULL EXPANSION AND MICROPROCESSOR
CLOCKOUT (B7)	B7 data register value	0	0
$\overline{\text{ENABLE}}$ (B6)	B6 data register value	1	1
R/$\overline{\text{W}}$ (B5)	B5 data register value	1	1
ALATCH (B4)	B4 data register value	0	0
ADDR/DATA (C7-C0)	Individual I/O	X	X
HIGH ADDR (D7-D0)	Individual I/O	X	X

X = Indeterminate

capture latch operation during power reduction

In Wake-up mode, Timer 1's capture latch will not be loaded when the $\overline{\text{INT3}}$ pin is taken to its active level if T1HALT is 1. If T1HALT is 0 then the capture latch will be loaded every time the $\overline{\text{INT3}}$ pin is taken to its active level, regardless of the value of the INT3EN flag. Similarly, Timer 2's capture latch in Wake-up mode will be loaded when the $\overline{\text{INT1}}$ pin is taken to its active level if T2HALT is 1. If T2HALT is 0, then the capture latch will be loaded every time the $\overline{\text{INT1}}$ pin is taken to its active level, regardless of the value of the INT1EN flag.

Once the TMS77C82 has been brought out of Wake-up mode, the capture latch will always be loaded when the appropriate interrupt pin is taken to its active level.

TMS77C82 SYSTEM CLOCK OPTIONS

The internal state cycle period, called Tc(c), is derived from the clock system that is applied on the XTAL pins of the circuit. The internal clock then divides the external clock source frequency by two to produce the internal state frequency; for example, a 5 MHz crystal produces an internal frequency of 2.5 MHz, which drives a 400-ns machine cycle . TMS77C82 devices can use a crystal, ceramic resonator, or another approximately 50% duty cycle clock as an external clock source. If the TMS77C82 contains the RC mask option it shall use an R-C circuit or a Ceramic resonator in the range described herebelow .

. System Clock Connections

The TMS77C82 uses the following methods to implement the clock options .

CRYSTALS: Parallel resonant crystals are connected between pins XTAL1 and XTAL2/CLKIN. To optimize the crystal waveform unbalanced capacitors should be connected, 15-pF between XTAL1 and Ground, and 33pF between XTAL2 and Ground. The crystal and components should be mounted as close as possible to the input pin to minimize output distortion and start-up stabilisation time. This connection is illustrated in Figure 19 a.

Crystals can be used with XTAL mask option only .

CERAMIC RESONATORS: Ceramic resonators are connected between pins XTAL1 and XTAL2/-CLKIN. A resistor and two capacitors, with values determined by the selected ceramic resonator, must be connected as shown in Figure 19 b. Values vary by manufacturer and type. Ceramic resonators can be used with both XTAL and RC mask options. The following values are recommended by the Murata manufacturer :

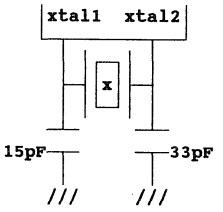
Mask Option	Ceramic Resonator	Frequency (Hz)	RF (MOhm)	RD (KOhm)	CAP (pF)	VCC Range (V)
RC	CSB600P	600 K	1	2.7	150	2.0 -- 5.5
	CSA8.00MT	8 M	1	0	22	4.0 -- 5.5
	CSB1100JT	1100 K	1	2.7	100	2.0 -- 5.5
	CSA3.58MG	3.58 M	1	0	30	3.5 -- 5.5
XTAL	CSB500E	500 K	1	5.6	100	2.0 -- 5.5
	CSA5.00MG	5 M	1	0	30	4.0 -- 5.5
	CSA8.00MT	8 M	1	0	30	4.0 -- 5.5

EXTERNAL CLOCK SOURCE: As shown in Figure 19c, external clock sources are connected to XTAL2/CLKIN, and XTAL1 is not connected. External clock sources can be used with both XTAL and RC mask options.

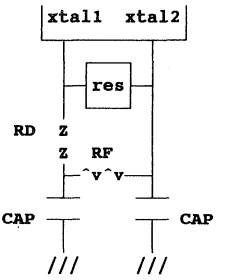
R-C CIRCUITS: R-C circuits provide a simple, low-cost oscillator for applications in which frequency toleration is not a concern. R-C circuits also provide immediate start-up oscillation upon exiting the Halt-Off mode operation.

R-C circuits are connected as shown in Figure 19 d. The recommend ed value for the capacitor C is 47 pF. The value of the resistor R required for the desired frequency must be selected with respect to VCC, ambient temperature, and the tolerance of the R-C components. Recommended values for the resistor in the R-C network fall in the range of 1KΩ-100KΩ.

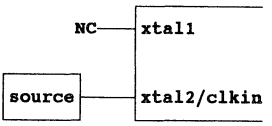
R-C circuit can be used with R-C mask option only .



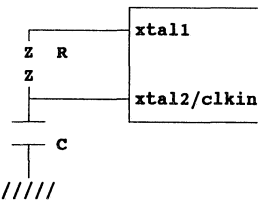
a. Crystal



b. Ceramic Resonator



c. External Source

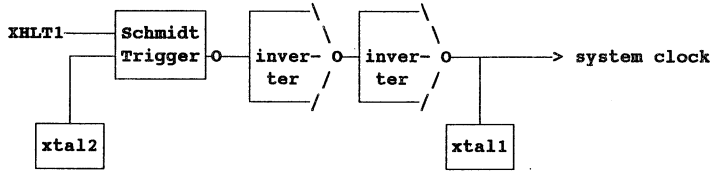


d. R-C circuit

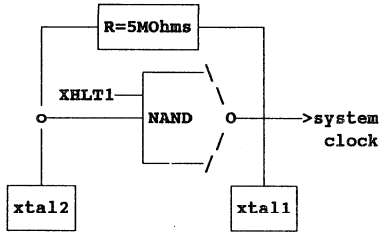
Figure 19 SYSTEM CLOCK CONNECTIONS

Like all devices of the 7000 CMOS family, TMS77C82 devices may use oscillator mask options which provide different levels of functionality and power consumption during the Halt low-power mode . These oscillator options are called RC/OSC-OFF and XTAL/OSC-ON. For these 2 options, the internal Clock circuit block diagrams are defined as follows :

. LOW-POWER MASK OPTION



a. R-C internal clock circuit



b. XTAL internal clock circuit

The OSC-On options (XTAL) keeps the on-chip oscillator active during the Halt mode. When the device is brought out of Halt mode, there is no delay in restoring the full operation since the oscillator is already running. This option is useful in applications where no delay in res toring full operation after Halt mode is more important than the lower power consumption .

The OSC-Off option (RC) is useful in applications where very low power consumption is required in Halt mode. This option causes the oscillator to cease oscillation when Halt mode is entered. This offers the lowest power consumption around 7μA. The RC option (called TMS77C82 NRC or FNRC) supports an R-C circuit as well as a ceramic resonator or other external 50% duty cycle CLKIN signal. If an R-C network is used, it will restart full oscillation immediately upon exiting Halt mode. If a ceramic resonator is used, there will be a period before the oscillations stabi lize, causing a delay in the response to RESET of about 10 milliseconds.

Note:
RC, XTAL are 'mask options', which means the option is placed on a manufacturing template, or mask, that copies the actual circuit onto the silicon device. This means the oscillator option is finalized at the start of manufacture and cannot be changed by software or hardware.

Instruction set

The TMS7000 family instruction set consists of 57 instructions that control input, output, data manipulation, data comparisons, and program flow. The instruction set is supported with eight addressing modes to provide the flexibility to optimize programs to the user's applications. For example, the MOV instruction has ten operand combinations supported by its addressing modes.

ADDRESSING MODE	EXAMPLE		OPERATION
Single Register	DEC	R24	(R24) – 1 → (R24)
Dual Register	ADD	R32,R17	(R32) + (R17) → (R32)
Peripheral File	XORP	A,P17	(P17) .XOR. (A) → (P17)
Immediate	AND	%>C5,R35	(R35) .AND. >C5 → (R35)
Program Counter Relative	JMP	LABEL	(PC) + off8 → (PC)
Direct Memory	LDA	@>F3D4	(F3D4) → A
Register File Indirect	STA	*R22	(A) → (R22)
Indexed	BR	@>1AAA(B)	(>1AAA) + (B) → (PC)

The CPU controls instruction execution by executing microinstructions from a dedicated control memory. The number of system clock cycles required to execute one assembly language instruction varies depending on the instruction complexity and operand addressing mode. Instruction execution times are stated in terms of the number of system clock cycles per instruction. This decouples the bus transaction protocol/timing from specific memory performance requirements. Instruction execution times vary from 5 to 49 internal system clock cycles, with most instructions requiring less than ten cycles to complete.

Similarly, the number of bytes of program memory required to store an instruction will vary with instruction complexity and addressing mode. TMS7000 instructions require from one to four bytes of program memory space, with most instructions occupying one or two bytes.

The TMS7000 FAMILY INSTRUCTION SET SUMMARY, beginning on page 30, shows the instruction set, the addressing modes, the program memory byte length, and the execution cycle count for each instruction. The Addressing Mode entries are in the format of BYTE LENGTH/CYCLE COUNT. The following symbols and abbreviations are used:

SYMBOL	DEFINITION	SYMBOL	DEFINITION
s	Source Operand	d	Destination Operand
A	Register A or R0 in Register File	B	Register B or R1 in Register File
RF	Source or Destination Register in Register File	label	16-bit Label
Pn	Source or Destination Register in Peripheral File	iop16	16-bit Immediate Operand
Rp	Source or Destination Register Pair (Rn, Rn – 1)	PCN	16-bit Address of Next Instruction
iop	8-bit Immediate Operand	ST	Status Register
off8	8-bit Signed Offset (label – PC)	@	Extended Addressing Operand (Direct, Indirect, Indexed)
PC	Program Counter	→	Is Assigned To
SP	Stack Pointer	Stack	Present Address of Stack Pointer
%	Immediate Operand	()	Contents of
C	Status Register Carry Bit		

TMS7000 FAMILY INSTRUCTION SET SUMMARY

OPERATION		ADDRESSING MODES							OTHER	DESCRIPTION
		DIRECT				EXTENDED				
		A	B	RF	Pn	@label	*RF	@label(B)		
ADC	B,_	1/5								Add with Carry
	RF,_	2/8	2/8	3/10						(s) + (d) + (C) — (d)
	%iop,_	2/7	2/7	3/9						
ADD	B,_	1/5								Add
	RF,_	2/8	2/8	3/10						(s) + (d) — (d)
	%iop,_	2/7	2/7	3/9						
AND	B,_	1/5								And
	RF,_	2/8	2/8	3/10						(s) .AND. (d) — (d)
	%iop,_	2/7	2/7	3/9						
ANDP	A,_				2/10					And Peripheral
	B,_				2/9					(s) .AND. (Pn) — (Pn)
	%iop,_				3/11					
BR	_					3/10	2/9	3/12		Branch
										(d) — (PC)
BTJO†	B,_,off8	2/7								Bit Test and Jump If One
	RF,_,off8	3/10	3/10	4/12						If (s) .AND. (d) ≠ 0, then
	%iop,_,off8	3/9	3/9	4/11						(PC) + off8 — (PC)
BTJOP†	A,_,off8				3/11					Bit Test and Jump If One Peripheral
	B,_,off8				3/10					If (s) .AND. (d) ≠ 0, then
	%iop,_,off8				4/12					(PC) + off8 — (PC)
BTJZ†	B,_,off8	2/7								Bit Test and Jump If Zero
	RF,_,off8	3/10	3/10	4/12						If (s) .AND. NOT (d) ≠ 0, then
	%iop,_,off8	3/9	3/9	4/11						(PC) + off8 — (PC)
BTJZP†	A,_,off8				3/11					Bit Test and Jump If Zero Peripheral
	B,_,off8				3/10					If (s) .AND. NOT (d) ≠ 0, then
	%iop,_,off8				4/12					(PC) + off8 — (PC)
CALL	_					3/14	2/13	3/16		Call
										(SP) + 1 — (SP)
										(PC MSB) — (Stack)
										(SP) + 1 — (SP)
										(PC LSB) — (Stack)
										Operand Address — (PC)
CLR	_	1/5	1/5	2/7						Clear
										0 — (d)
CLRC									1/6	Clear Carry Bit
										0 — (C)
CMP	B,_	1/5								Compare
	RF,_	2/8	2/8	3/10						(d) – (s) computed; sets flags on result
	%iop,_	2/7	2/7	3/9						
CMPA	_					3/12	2/11	3/14		Compare A
										(A) – (s) computed; sets flags on result
DAC	B,_	1/7								Decimal Add with Carry
	RF,_	2/10	2/10	3/12						(s) + (d) + (C) — (d) (BCD)
	%iop,_	2/9	2/9	3/11						

† Add 2 to cycle count if jump is taken.

TMS7000 FAMILY INSTRUCTION SET SUMMARY (CONTINUED)

OPERATION		ADDRESSING MODES								DESCRIPTION
		DIRECT				EXTENDED			OTHER	
		A	B	RF	Pa	@lab	*RF	@lab(B)		
DEC	_	1/5	1/5	2/7						Decrement (d) - 1 → (d)
DECD	_	1/9	1/9	2/11						Decrement Double (Rp) - 1 → (Rp)
DINT									1/5	Disable Interrupts 0 → (Global Interrupt Enable Bit)
DJNZ†	A,off8	2/7								Decrement and Jump If Not 0 (d) - 1 → (d); if (d) ≠ 0, then (PC) + off8 → (PC)
	B,off8		2/7							
	RF,off8			3/9						
DSB	B,_	1/7								Decimal Subtract with Borrow (d) - (s) - 1 + (C) → (d) (BCD)
	RF,_	2/10	2/10	3/12						
	%iop,_	2/9	2/9	3/11						
EINT									1/5	Enable Interrupts 1 → (Global Interrupt Enable Bit)
IDLE									1/6+	Idle (PC) → (PC) until interrupt (PC) + 1 → (PC) after return from interrupt
INC	_	1/5	1/5	2/7						Increment (d) + 1 → (d)
INV	_	1/5	1/5	2/7						Invert NOT(d) → (d)
JMP	off8								2/7	Jump (PC) + off8 → (PC)
Jcnd†	off8									Jump on Condition (PC) + off8 → (PC)
JC									2/5	Jump if Carry
JEQ									2/5	Jump if Equal
JGE									2/5	Jump if Greater Than or Equal
JGT									2/5	Jump if Greater Than
JHS									2/5	Jump if Higher or Same value
JL									2/5	Jump if Lower
JNC									2/5	Jump if No Carry
JNE									2/5	Jump if Not Equal
JNZ									2/5	Jump if Not Zero
JP									2/5	Jump if Positive
JPZ									2/5	Jump if Positive or Zero
JZ									2/5	Jump if Zero
LDA	_					3/11	2/10	3/13		Load Accumulator (s) → (A)
LDSP									1/5	Load Stack Pointer (B) → (SP)
MOV	A,_		1/8	2/8						Move (s) → (d)
	B,_	1/5		2/7						
	RF,_	2/8	2/8	2/10						
	%iop,_	2/7	2/7	3/9						

† Add 2 to cycle count if jump is taken.

TMS7000 FAMILY INSTRUCTION SET SUMMARY (CONTINUED)

OPERATION		ADDRESSING MODES								DESCRIPTION
		DIRECT				EXTENDED			OTHER	
		A	B	RF	Pn	@tab	*RF	@tab(B)		
MOVD	%iop16,_			4/15						Move Double
	%iop16(B),_			4/17						(Rp) ← (Rp), or
	Rp,_			3/14						iop16 ← (Rp)
MOVP	A,_				2/10					Move Peripheral
	B,_				2/9					(s) ← (Pn)
	%iop,_				3/11					
	Pn,_	2/9	2/8							
MPY	B,_	1/44								Multiply
	Rn,_	2/47	2/47	3/49						(s) × (d) ← (A,B)
	%iop,_	2/46	2/46	3/48						A = MSB, B = LSB
NOP									1/4	No Operation
										(PC) + 1 ← (PC)
OR	B,_	1/5								OR
	RF,_	2/8	2/8	3/10						(s) .OR. (d) ← (d)
	%iop,_	2/7	2/7	3/9						
ORP	A,_				2/10					OR Peripheral
	B,_				2/9					(s) .OR. (Pn) ← (Pn)
	%iop,_				3/11					
POP	_	1/6	1/6	2/8						Pop
										(Stack) ← (d); (SP) - 1 ← (SP)
POP	ST								1/6	Pop Status
										(Stack) ← (ST); (SP) - 1 ← (SP)
PUSH	_	1/6	1/6	2/8						Push
										(s) ← (Stack)
										(SP) + 1 ← (SP)
PUSH	ST								1/6	Push Status
										Status Register ← (Stack);
										(SP) + 1 ← (SP)
RETI									1/9	Return from Interrupt
										(Stack) ← (PC) LSB
										(SP) - 1 ← (SP)
										(Stack) ← (PC) MSB
										(SP) - 1 ← (SP)
										(Stack) ← (Status Register)
										(SP) - 1 ← (SP)
RETS									1/7	Return from Subroutine
										(Stack) ← (PC) LSB
										(SP) - 1 ← (SP)
										(Stack) ← (PC) MSB
										(SP) - 1 ← (SP)
RL	_	1/5	1/5	2/7						Rotate Left
										Bit(n) ← Bit(n + 1)
										Bit(7) ← Bit(0) + (C)

TMS7000 FAMILY INSTRUCTION SET SUMMARY (CONCLUDED)

OPERATION		ADDRESSING MODES								DESCRIPTION
		DIRECT				EXTENDED			OTHER	
		A	B	RF	Pn	@lab	*RF	@lab(B)		
RLC	_	1/5	1/5	2/7						Rotate Left Through Carry Bit(n) ← Bit(n + 1) (C) ← Bit(0) Bit(7) ← (C)
RR	_	1/5	1/5	2/7						Rotate Right Bit(n) ← Bit(n + 1) Bit(0) ← Bit(7) + (C)
RRC	_	1/5	1/5	2/7						Rotate Right Through Carry Bit(n) ← Bit(n + 1) (C) ← Bit(n) Bit(0) ← (C)
SBB	B,_	1/5								Subtract with Borrow (s) − (d) − 1 + (C) → (d)
	RF,_	2/8	2/8	3/10						
	%iop,_	2/7	2/7	3/9						
SETC									1/5	Set Carry 1 → (C)
STA	_					3/11	2/10	3/13		Store Accumulator: (A) → (d)
STSP									1/6	Store Stack Pointer (SP) → (B)
SUB	B,_	1/5								Subtract (d) − (s) → (d)
	RF,_	2/8	2/8	3/10						
	%iop,_	2/7	2/7	3/9						
SWAP	_	1/8	1/8	2/10						Swap Nibbles d(Hn,Ln) → d(Ln,Hn)
TRAP	0-23								1/14	Trap (SP) + 1 → (SP) (PC MSB) → (Stack) (SP) + 1 → (SP) (PC LSB) → (Stack) (Entry Vector) → (PC)
TSTA									1/6	Test A C ← 0 N, Z set on (A)
TSTB									1/5	Test B C ← 0 N, Z set on (B)
XCHB	_	1/8		2/8						Exchange B (B) ↔ (d)
XOR	B,_	1/5								Exclusive OR (s) .XOR. (d) → (d)
	RF,_	2/8	2/8	3/10						
	%iop,_	2/7	2/7	3/9						
XORP	A,_				2/10					Exclusive OR Peripheral (s) .XOR. (Pn) → (Pn)
	B,_				2/9					
	%iop,_				3/11					

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V_{CC} (see Note 6) . . . . . −0.3 to 7 V
Supply voltge, V_{PP} (MC pin) . . . . . −0.3 V to 14 V
Input voltage range . . . . . −0.3 V to V_{CC} +0.3 V
Output voltage range . . . . . −0.3 V to V_{CC} +0.3 V
Maximum I/O buffer current (per pin) . . . . . ±10 mA
Maximum supply current, I_{CC} . . . . . 60 mA
Maximum supply current, I_{SS} . . . . . −60 mA
Storage temperature range . . . . . −55°C to 150°C
Operating free air temperature . . . . . −40°C to 85°C

†Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
NOTE 6: All voltage values in this data sheet are with respect to V_{SS}.

recommended operating conditions

		MIN	TYP	MAX	UNIT
Supply voltage		3		6	V
Programming supply voltage (MC pin)		12	12.5	13	V
INT1, INT3, RESET, X'TAL PIN					
High level input voltage	$5.5V \leq V_{CC} \leq 6V$	$V_{CC} - 1.0$		V_{CC}	V
	$4.5V \leq V_{CC} \leq 5.5V$	$V_{CC} - 0.7$		V_{CC}	V
	$3.5V \leq V_{CC} \leq 4.5V$	$V_{CC} - 0.5$		V_{CC}	V
	$2.5V \leq V_{CC} \leq 3.5V$	$V_{CC} - 0.35$		V_{CC}	V
Low level input voltage	$5.5V \leq V_{CC} \leq 6V$	0		1.00	V
	$4.5V \leq V_{CC} \leq 5.5V$	0		0.70	V
	$3.5V \leq V_{CC} \leq 4.5V$	0		0.50	V
	$2.5V \leq V_{CC} \leq 3.5V$	0		0.35	V
MC PIN					
High level input voltage, VIH	$5V \leq V_{CC} \leq 6V$	$V_{CC} - 0.5$		V_{CC}	V
	$4V \leq V_{CC} \leq 5V$	$V_{CC} - 0.4$		V_{CC}	V
	$3V \leq V_{CC} \leq 4V$	$V_{CC} - 0.3$		V_{CC}	V
	$2.5V \leq V_{CC} \leq 3V$	$V_{CC} - 0.2$		V_{CC}	V
Low level input voltage, VIL	$5V \leq V_{CC} \leq 6V$	0		0.5	V
	$4V \leq V_{CC} \leq 5V$	0		0.4	V
	$3V \leq V_{CC} \leq 4V$	0		0.3	V
	$2.5V \leq V_{CC} \leq 3V$	0		0.2	V
PORT (EXCEPT INT1, INT3, RESET, X'TAL, MC					
High level input voltage, VIH	$5V \leq V_{CC} \leq 6V$	$V_{CC} - 1.3$		V_{CC}	V
	$4V \leq V_{CC} \leq 5V$	$V_{CC} - 1.0$		V_{CC}	V
	$3V \leq V_{CC} \leq 4V$	$V_{CC} - 0.7$		V_{CC}	V
	$2.5V \leq V_{CC} \leq 3V$	$V_{CC} - 0.4$		V_{CC}	V
Low level input voltage, VIL	$5V \leq V_{CC} \leq 6V$	0		1.5	V
	$4V \leq V_{CC} \leq 5V$	0		1.1	V
	$3V \leq V_{CC} \leq 4V$	0		0.7	V
	$2.5V \leq V_{CC} \leq 3V$	0		0.3	V

electrical characteristics over full range of operating conditions (V_{CC} = 4.5 V to 5.5 V)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
ICC Supply current	Operating mode	V_{CC} = 5.0 V, f = 6.0 Mhz		23.5	29	mA
		V_{CC} = 5.0 V, f = 4.0 Mhz		20.5	26	mA
		V_{CC} = 5.0 V, f = 1.0 Mhz		19	21	mA
	Wake-up modes 1 & 5 (see Note 1)	<Freq = F Mhz> <Freq = 6.0 Mhz>		4	2.0*F + 0.5 12	mA mA
	Wake up mode 2 (see Note 1)	<Freq = F Mhz> <Freq = 6.0 Mhz>		2	1.44*F + 0.5 9	mA mA
	Wake-up modes 3 & 4 (see Note 1)	<Freq = F MHz> <Freq = 6 MHz>		1.7	0.8*F + 0.5 5.5	mA mA
Xtal Option	Halt-off mode	V_{CC} = 5 V, f= 6.0 Mhz		10	20	µA
RC Option	Halt-off mode	V_{CC} = 5 V, f= 6.0 Mhz		7	15	µA
V_{OH} High-level output voltage		V_{CC} = 5.0 V, I_{OH} = −1.0 mA	2.50	4.5		mA
		V_{CC} = 5.0 V, I_{OH} = 0.3 mA	4.50	4.8		V
V_{OL} Low-level output voltage		V_{CC} = 5.0, I_{OL} = 1.7 mA		0.3	0.4	V
I_{OH} Output source current ($V_{HO} = V_{CC} - 0.5$ V)		V_{CC} = 3.0 V	−100	−700		µA
		V_{CC} = 4.0 V	−0.2	1.0		mA
		V_{CC} = 5.0 V	−0.3	−1.2		mA
	V_{OH} = 2.5 V	V_{CC} = 5.0 V	−1.0	−5.0		mA
I_{OL} Output sink current (V_{OL} = 0.4 V)		V_{CC} = 3.0 V	0.7	2.0		mA
		V_{CC} = 4.0 V	1.02	0.4		mA
		V_{CC} = 5.0 V	1.7	2.0		mA
I_I Input leakage current	MC	$V_I = V_{SS}$ or V_{CC}		±0.1	±5	µA
	All others	$V_I = V_{SS}$ to V_{CC}		±0.1	±5	µA
C_I Input capacitance				5		pF

All inputs = V_{CC} or V_{SS} (except XTAL2). All I/O and output pins are open circuit.

Note 1:

MODE	TIMER-1	TIMER-2	UART
WAKE-UP1	USE	USE	USE
WAKE-UP2	USE	OFF	USE
WAKE-UP3	OFF	OFF	USE
WAKE-UP4	USE	OFF	OFF
WAKE-UP5	USE	USE	OFF
HALT-OFF	OFF	OFF	OFF

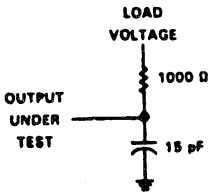
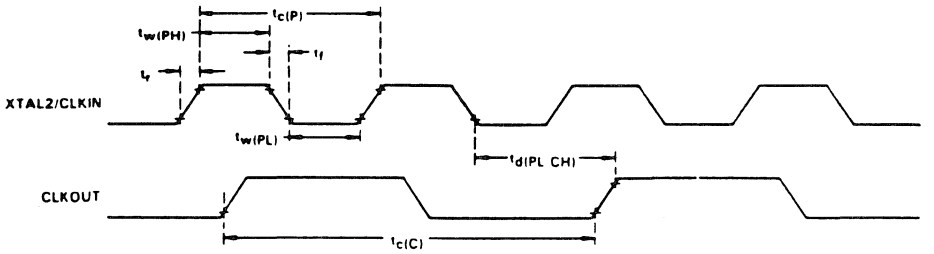


FIGURE 26. OUTPUT LOAD CIRCUIT USED FOR ALL TIMING MEASUREMENTS

recommended crystal operating conditions over full operating range

PARAMETER			MIN	TYP	MAX	UNIT
f_{osc}	Crystal frequency	V_{CC} = 3.0 V	0.5		1.0	MHz
		V_{CC} = 4.0 V	0.5		3.0	MHz
		V_{CC} = 5.0 V ± 10%	0.5		6.0	MHz
		V_{CC} = 6.0 V	0.5		6.0	MHz
	CLKIN duty cycle		45		55	%
$t_{c}(P)$	Crystal cycle time	V_{CC} = 2.5 V	1000		2000	ns
		V_{CC} = 4.0 V	333		2000	ns
		V_{CC} = 5.0 V	167		2000	ns
		V_{CC} = 6.0 V	167		2000	ns
$t_{c}(C)$	Internal state cycle time	V_{CC} = 2.5 V	2000		4000	ns
		V_{CC} = 4.0 V	666		4000	ns
		V_{CC} = 5.0 V	333		4000	ns
		V_{CC} = 6.0 V	333		4000	ns
$t_{w}(PH)$	CLKIN pulse duration high		50			ns
$t_{w}(PL)$	CLKIN pulse duration low		50			ns
t_{r}	CLKIN rise time				30	ns
t_{f}	CLKIN fall time				30	ns
$t_{d}(PL\text{-}CH)$	CLKIN fall to CLKOUT rise delay time			140	250	ns

clock timing



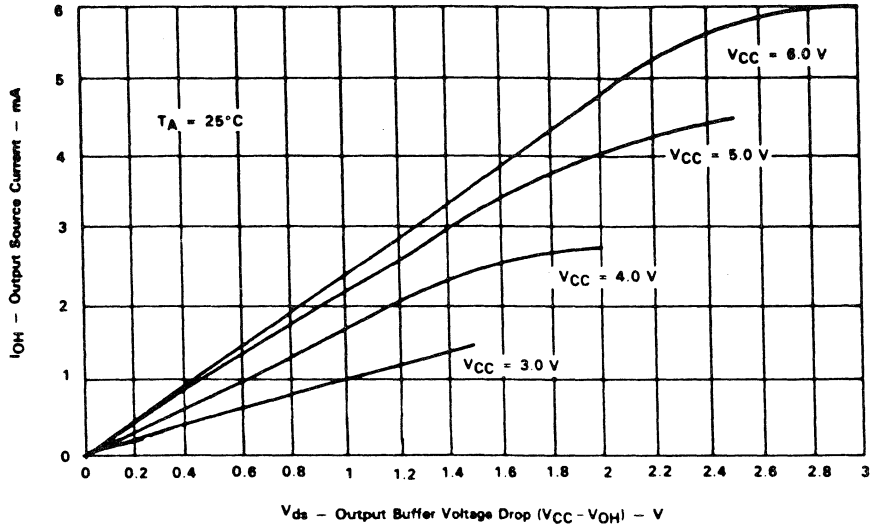


FIGURE 24. TYPICAL OUTPUT SOURCE CHARACTERISTICS

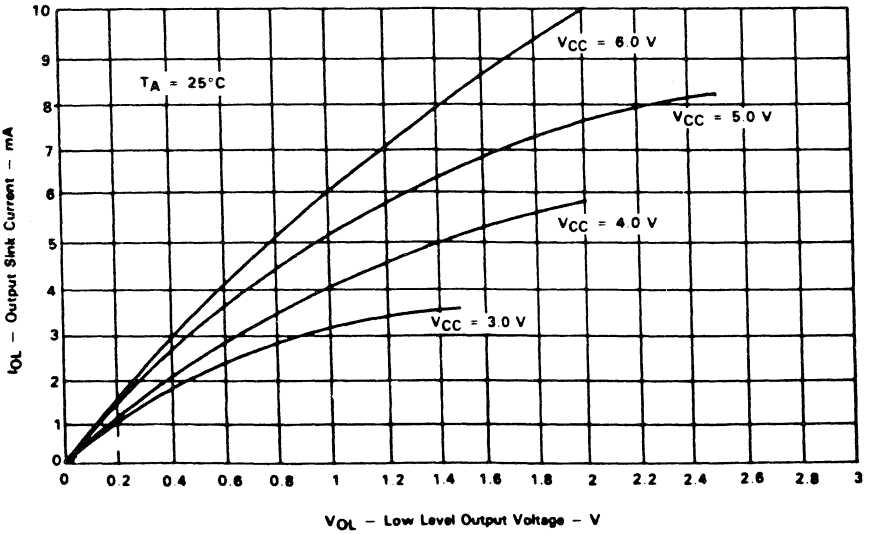


FIGURE 25. TYPICAL OUTPUT SINK CHARACTERISTICS

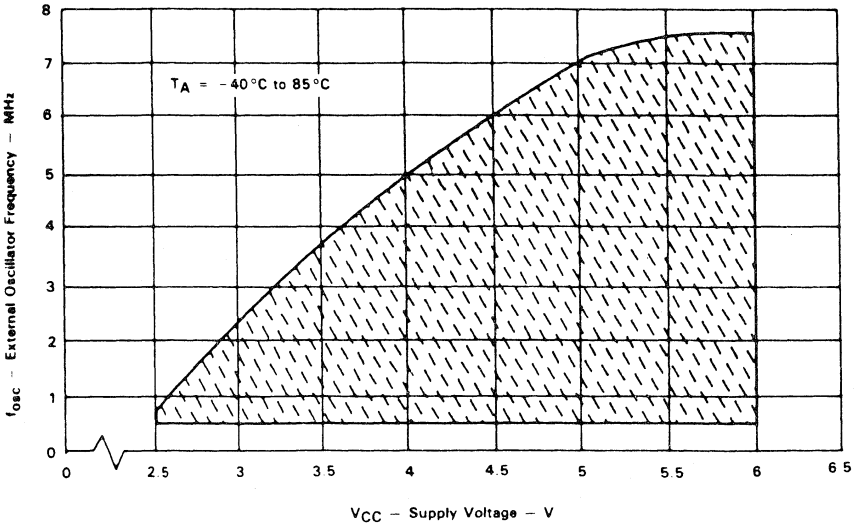


FIGURE 20. OPERATING FREQUENCY RANGE

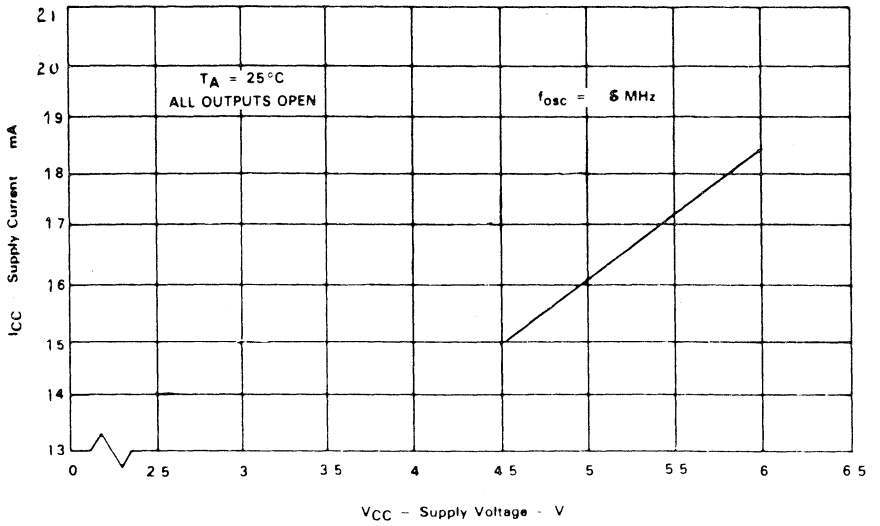


FIGURE 22. TYPICAL OPERATING CURRENT VS. SUPPLY VOLTAGE

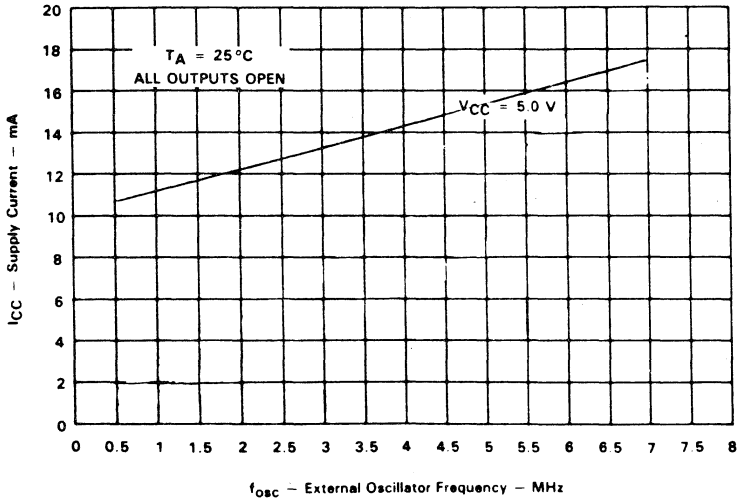
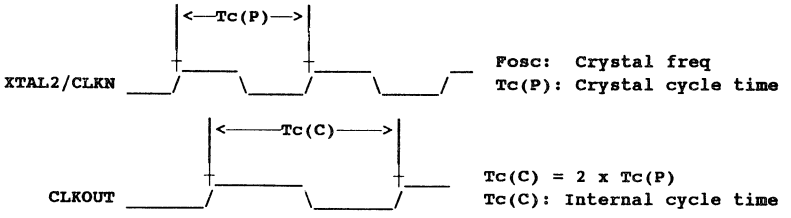


FIGURE 23. TYPICAL OPERATING I_{CC} VS. OSCILLATOR FREQUENCY

memory interface timing as a function of frequency

	Parameter	MIN	TYP	MAX	
Tc(C)	CLCKOUT cycle time.....		Tc		NS
Tw(CH)	CLCKOUT high pulse duration.....	0.5Tc-100	0.5Tc	0.5Tc+100	
Tw(CL)	CLCKOUT low pulse duration.....	0.5Tc-100	0.5Tc	0.5Tc+100	
Tc (R)	Read cycle time.....		2Tc		NS
Tc (W)	Write cycle time.....		2Tc		NS
Td(CH-JL)	Clockout high to ALATCH low.....	0.5Tc-50	0.5Tc		NS
Td(CH-JH)	Clockout high to ALATCH high.....	0.25Tc-40	0.25Tc		NS
Td(CH-HA)	Clockout high to high address valid...	0.25Tc-40	0.25Tc		NS
Td(CH-EL)	Clockout high to ENABLE low.....	-10	30		NS
Ta(EL-D)	ENABLE active to data in.....	0.75Tc-160	0.75Tc		NS
Th(EH-D)	Data in hold after ENABLE inactive....	0			NS
Ta(A-D)	Data in from address valid.....	1.5Tc-200	1.5Tc-100		
Tw(JH)	ALATCH active.....	0.25Tc-50	0.25Tc		NS
Tsu(HA-JL)	High address to ALATCH fall.....	0.25Tc-50	0.25Tc		NS
Tsu(LA-JL)	Low address to ALATCH fall.....	0.25Tc-55	0.25Tc		NS
Th(JL-LA)R	Low address hold from ALATCH fall (RD)	0.25Tc-50	0.25Tc		NS
Th(EH-RW)	Enable inactive to R/W.....	0.5Tc-100	0.5Tc		NS
Td(EH-JH)	ENABLE inactive to ALATCH high.....	0.5Tc-60	0.5Tc		NS
Td(EH-A)	ENABLE inactive to low address drive..	0.5Tc-100	0.5Tc		NS
Th(EH-HA)	High address hold from ENABLE inactive	0.5Tc-100	0.5Tc		NS
Tsu(Q-EH)	Data out to ENABLE inactive.....	0.5Tc-50	0.5Tc		NS
Th(EH-Q)	Data out hold from ENABLE inactive....	0.5Tc-60	0.5Tc		NS
Td(LA-EL)	Low address high-Z to ENABLE active...	0	0.25Tc		NS
Td(A-EH)	Low address to ENABLE high.....	1.5Tc-100	1.5Tc		NS
Th(JL-LA)w	Low address hold from ALATCH fall (WR)	0.75Tc-100	0.75Tc		NS
Tsu(RW-JL)	R/W valid before ALATCH fall.....	0.25Tc-60	0.25Tc		NS
Tw(E)	ENABLE pulse width.....	0.75Tc-80	0.75Tc		NS

+ Vcc = 5 V +/-10%, tc = 2/freq



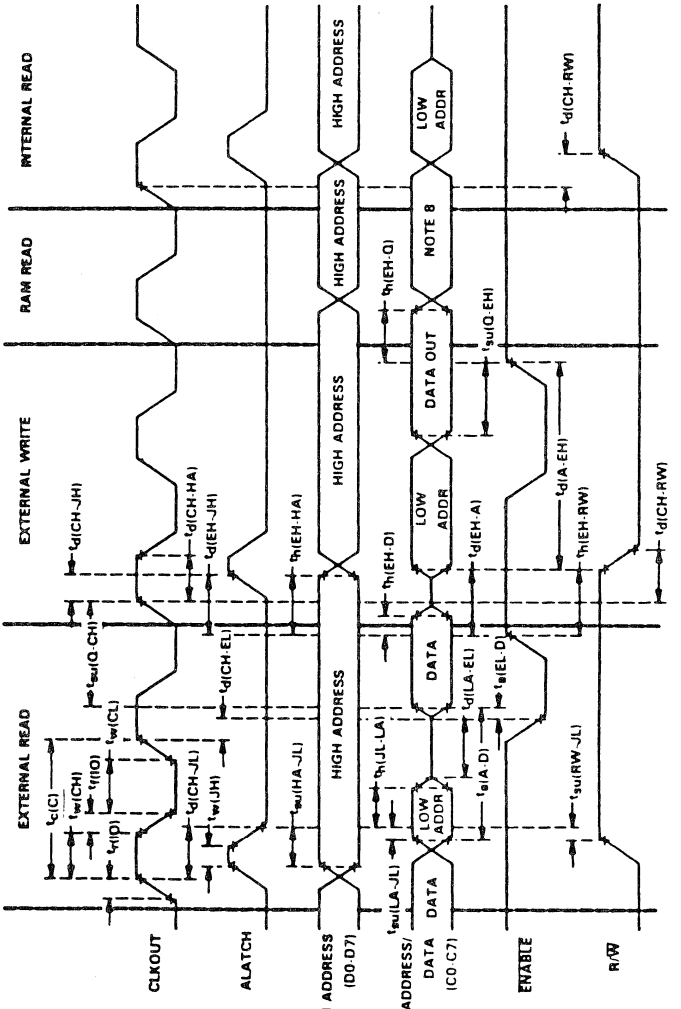
NOTE: Period of internal clock Tc(C) = 2 x Tc(P) = 2 / F osc.
Timings are given in Tc(C).

memory interface timing at 5 Mhz

Parameter		MIN	TYP	MAX	
Tc(C)	CLCKOUT cycle time.....		400		NS
Tw(CH)	CLCKOUT high pulse duration.....	100	200	300	NS
Tw(CL)	CLCKOUT low pulse duration.....	100	200	300	
Tc (R)	Read cycle time.....		800		NS
Tc (W)	Write cycle time.....		800		NS
Td(CH-JL)	Clockout high to ALATCH low.....	150	200		NS
Td(CH-JH)	Clockout high to ALATCH high.....	60	100		NS
Td(CH-HA)	Clockout high to high address valid...	60	100		NS
Td(CH-EL)	Clockout high to ENABLE low.....	-10	30		NS
Ta(EL-D)	ENABLE active to data in.....	140	300		NS
Th(EH-D)	Data in hold after ENABLE inactive....	0			NS
Ta(A-D)	Data in from address valid.....	400	500		NS
Tw(JH)	ALATCH active.....	50	100		NS
Tsu(HA-JL)	High address to ALATCH fall.....	50	100		NS
Tsu(LA-JL)	Low address to ALATCH fall.....	45	100		NS
Th(JL-LA)R	Low address hold from ALATCH fall (RD)	50	100		NS
Th(EH-RW)	Enable inactive to R/W.....	100	200		NS
Td(EH-JH)	ENABLE inactive to ALATCH high.....	140	200		NS
Td(EH-A)	ENABLE inactive to low address drive..	100	200		NS
Th(EH-HA)	High address hold from ENABLE inactive	100	200		NS
Tsu(Q-EH)	Data out to ENABLE inactive.....	150	200		NS
Th(EH-Q)	Data out hold from ENABLE inactive....	140	200		NS
Td(LA-EL)	Low address high-Z to ENABLE active...	0	100		NS
Td(A-EH)	Low address to ENABLE high.....	500	600		NS
Th(JL-LA)w	Low address hold from ALATCH fall (WR)	200	300		NS
Tsu(RW-JL)	R/W valid before ALATCH fall.....	40	100		NS
Tw(E)	ENABLE pulse width.....	220	300		NS

† Vcc = 5 V +/-10%, tc = 2/freq

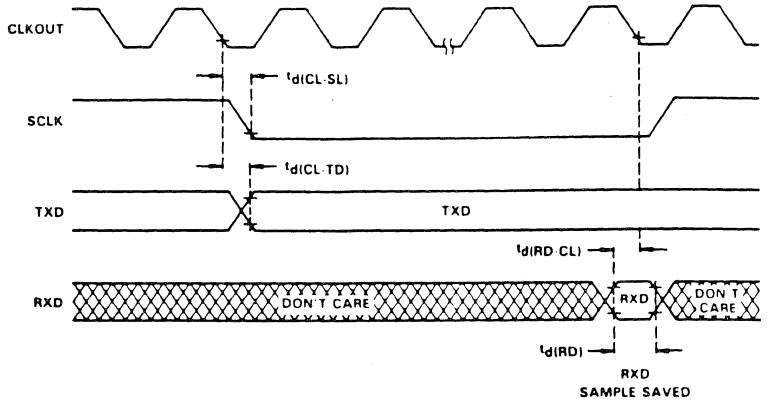
read and write cycle timing



NOTE 8: During an internal RAM access, the CPORT outputs are stable but the data is a "don't care".

serial port timing

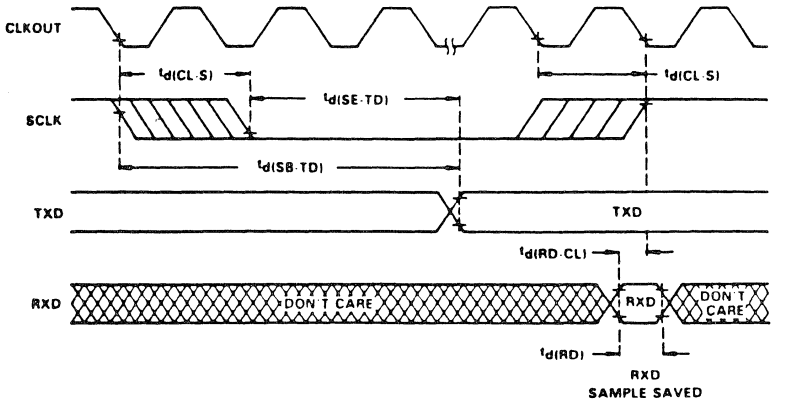
Internal serial clock



NOTES: 9. The CLKOUT signal is not available in Single-Chip mode.
10. CLKOUT = $t_{c(C)}$.

	PARAMETER	TYP	UNIT
$t_{d(CL-SL)}$	CLKOUT low to SCLK low	1/4 $t_{c(C)}$	ns
$t_{d(CL-TD)}$	CLKOUT low to new TXD data	1/4 $t_{c(C)}$	ns
$t_{d(RD-CL)}$	RXD data valid before CLKOUT low	1/4 $t_{c(C)}$	ns
$t_{d(RD)}$	RXD data valid time	1/2 $t_{c(C)}$	ns

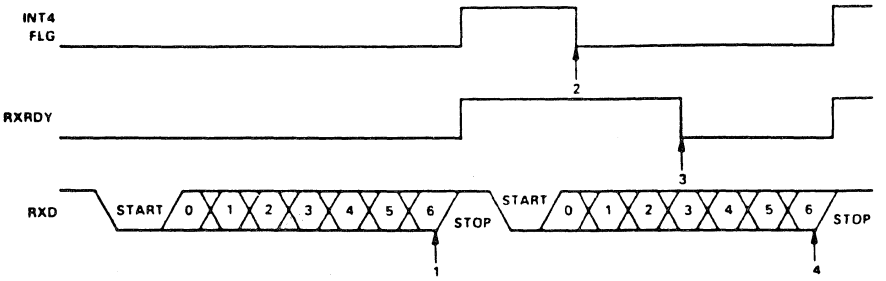
external serial clock



NOTES: 10. CLKOUT = $t_{c(C)}$.
11. The CLKOUT signal is not available in Single-Chip mode.
12. SCLK sampled; if SCLK = 1 then 0, fall transition found.
13. SCLK sampled; if SCLK = 0 then 1, rise transition found.

PARAMETER		TYP	UNIT
$t_{d(RD-CL)}$	RXD data valid before CLKOUT low	1/4 $t_{c(C)}$	ns
$t_{d(RD)}$	RXD data valid time	1/2 $t_{c(C)}$	ns
$t_{d(SB-TD)}$	Start of SCLK sample to new TXD data	3 1/4 $t_{c(C)}$	ns
$t_{d(SE-TD)}$	End of SCLK sample to new TXD data	2 1/4 $t_{c(C)}$	ns
$t_{d(CL-S)}$	CLKOUT low to SCLK transition	$t_{c(C)}$	ns

RX signals in communication modes

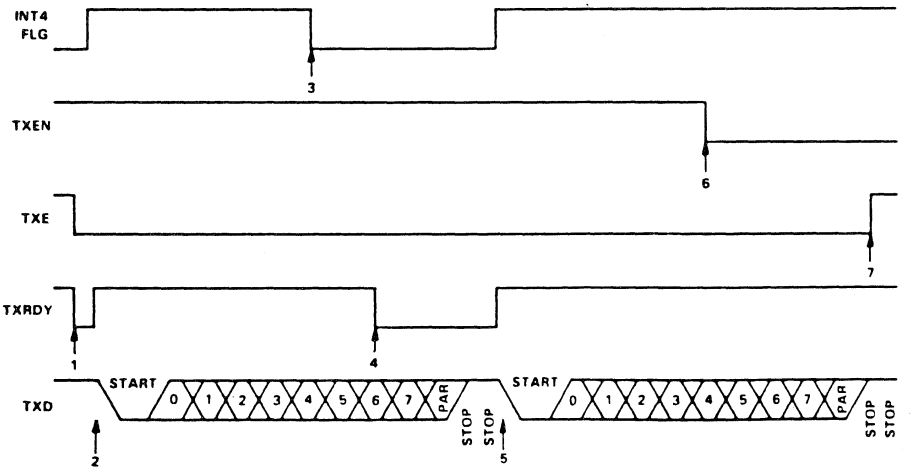


NOTES: 14. Format shown is start bit + seven data bits + stop bit.
15. SCLK is continuous, external or internal.
16. If RXEN = 0, RXSHF still receives data from RXD. However, the data is not transferred to RXBUF and RXRDY and INT4FLG are not set.

Sequence of Events:

1. RXSHF data is transferred to RXBUF. Error status bits are set if an error is detected.
2. Software writes to INT4CLR to clear INT4FLG; or, CPU clears INT4FLG on entry to level 4 interrupt routine.
3. Software reads RXBUF.
4. Repeat step 1.

TX signals in communication modes

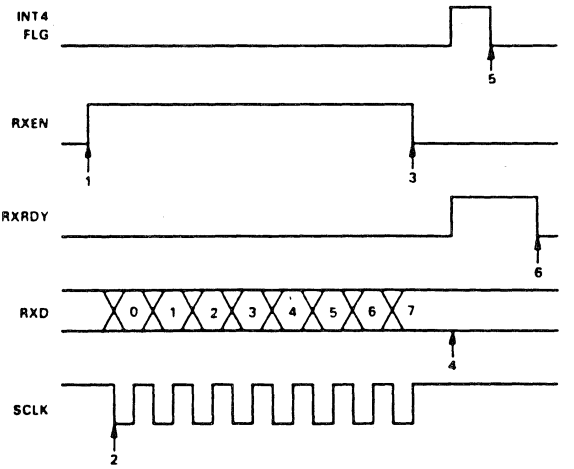


NOTES: 17. Format shown is start bit + eight data bits + parity bit + two stop bits.
18. SCLK is continuous whether internal or external.

Sequence of Events:

1. Software writes to TXBUF.
2. TXBUF and WU data are transferred to TXSHF and WUT (Wake-up temporary flag). INT4FLG and TXRDY are set.
3. Software writes to INT4CLR to clear INT4FLG or CPU clears INT4FLG on entry to level 4 interrupt routine.
4. Software writes to TXBUF.
5. TXBUF and WU data are transferred to TXSHF and WUT (Wake-up temporary flag). INT4FLG and TXRDY are set.
6. Software resets TXEN; current frame will finish and transmission will stop whether TXBUF is full or empty.
7. TXE is set if TXBUF and TXSHF are empty.

RX signals in serial I/O modes

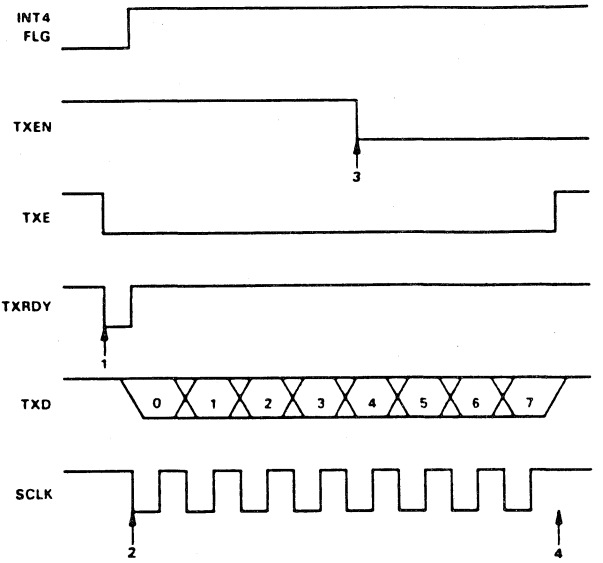


NOTES: 19. RXEN has no effect on INT4FLG or RXRDY in Serial I/O mode.
20. RXD is sampled on SCLK rise; external shift registers should be clocked on SCLK fall.

Sequence of Events

1. Software starts receiving by setting RXEN.
2. Gated SCLK starts and data is received.
3. RXEN is automatically cleared in last data bit.
4. RXSHF data is transferred to RXBUF, and RXRDY and INT4 are set.
5. Software writes to INT4CLR to clear INT4FLG; or, CPU clears INT4FLG on entry to level 4 interrupt routine.
6. Software reads RXBUF.

TX signals in serial I/O modes



NOTE 21: Format shown is eight data bits.

Sequence of Events

1. Software writes to TXBUF.
2. TXBUF data is transferred to TXSFT; INT4FLG and TXRDY are set, and SCLK starts.
3. Software resets TXEN, current frame will finish and transmission will halt whether TXBUF is full or empty.
4. Frame ends and SCLK stops because TXEN = 0.

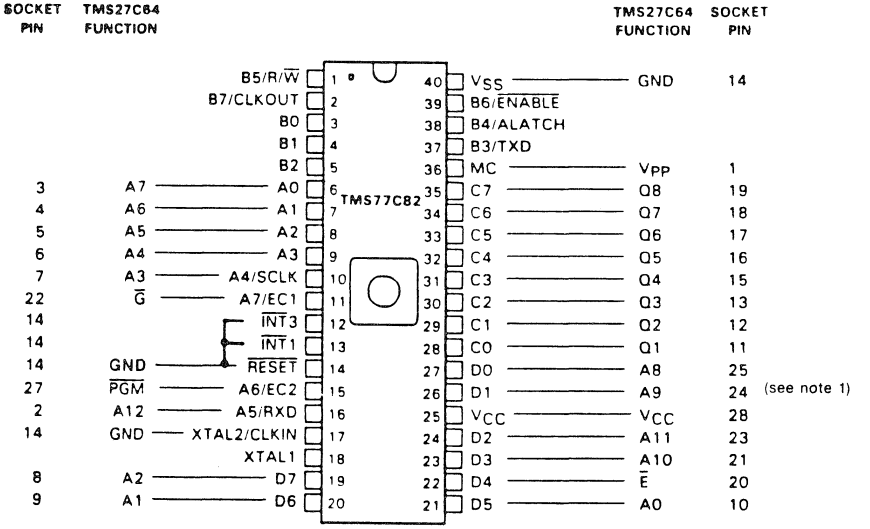
programming the TMS77C82 using a PROM programmer

The TMS77C82 can be programmed like any Texas Instruments TMS27C64 on a wide variety of PROM programmers. Programming the TMS77C82 requires a 40-to-28 pin adapter socket with the $\overline{\text{RESET}}$ and XTAL2 pins grounded. Contact your PROM programmer manufacturer or local TI field sales office for programming support.

adapter sockets

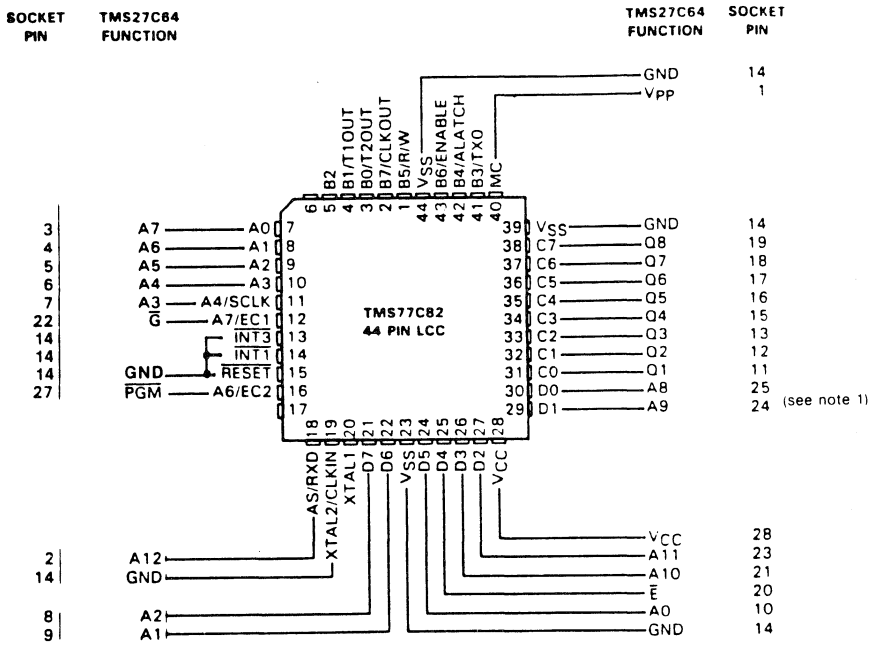
The following diagram shows the connections needed to be made on the 40-to-28 pin socket.

40-pin DIP to 28-pin socket



Note 1: For signature mode, insert a 3.9 K.ohms resistor between the pin # 24 of the socket and the pin # 26 of the TMS77C82 NL (Dil Package)

44-pin PLCC to 44-pin socket



Note 1: For signature mode, insert a 3.9 K.ohms resistor between the pin # 24 of the socket and the pin # 29 of the TMS77C82 FNL (Plcc Package)

EPROM Integrity protection using the R bit

Once the TMS77C82 has been programmed with the desired code, the contents of the EPROM may be protected with the use of the R bit integrity feature. The function of the R bit is to disable all external accesses to the on-chip EPROM while in the EPROM mode, which will prevent a protected code from being modified or read externally. The only way to "unprotect" the TMS77C82 after the R bit has been programmed is by erasing the EPROM, thereby destroying the protected code.

The following table and procedure demonstrates how to program and verify the R bit.

TMS77C82		R BIT	
FUNCTION	PIN #	PROGRAM	VERIFY
XTAL/CLKIN	17	V_{SS}	V_{SS}
$\overline{\text{RESET}}$	14	V_{SS}	V_{SS}
MC	36	12.5 V	V_{CC}
$\overline{\text{INT3}}$	12	12.5 V	12.5 V
D4	22	V_{CC}	V_{SS}
A7/EC1	11	V_{CC}	V_{SS}
A6/EC2	15	V_{CC}	V_{CC}
A3	9	V_{CC}	X
A1	7	X	V_{SS}
C7	35	$V_{IH}/V_{IL}/V_{IH}$	Refer to Step 3 in R Bit Verify Procedure

Note: X = Don't Care.

R bit Programming Procedure:

1. Configure all referenced pins for the R bit Program mode.
2. Power up the device.
3. Toggle C7 from a logical high (1), to a logical low (0), and back to a logical high (1).
4. Power down the device.

R bit Verify Procedure:

1. Configure all referenced pins for the R bit Verify mode.
2. Power up the device.
3. Read C7. Zero (0) is programmed, one (1) is not programmed.
4. Power down the device.

erasure

The TMS77C82 is erased by exposing the chip to shortwave ultraviolet light that has a wavelength of 253.7 nanometers (2537 angstroms). The recommended minimum exposure dose (UV intensity × exposure time) is 15 watt-seconds-per-square-centimeter. The lamp should be located about 2.5 centimeters (1 inch) above the chip during erasure. After erasure, all bits are at a high level. It should be noted that normal ambient light contains the correct wavelength for erasure. Therefore, when using the TMS77C82 the window should be covered with an opaque label.

programming characteristics over recommended supply voltage range and operating free-air temperature range T_A = 25°C, V_{CC} = 6 V, V_{PP} = 12.5 V

	PARAMETER	TEST CONDITIONS†	MIN	MAX	UNIT
$t_{a(A)}$	Access time from address	C_L = 100 pF, 1 Series 74 TTL load, $t_r \leq 20$ ns, $t_f \leq 20$ ns		450	ns
$t_{a(E)}$	Access time from chip enable			450	ns
$t_{en(G)}$	Output enable time from $\overline{G}$			150	ns
t_{dis}‡	Output disable time from $\overline{G}$ or $\overline{E}$, whichever occurs first		0	130	ns
$t_{v(A)}$	Output data valid time after change of address, $\overline{E}$, or $\overline{G}$ whichever occurs first		0		ns

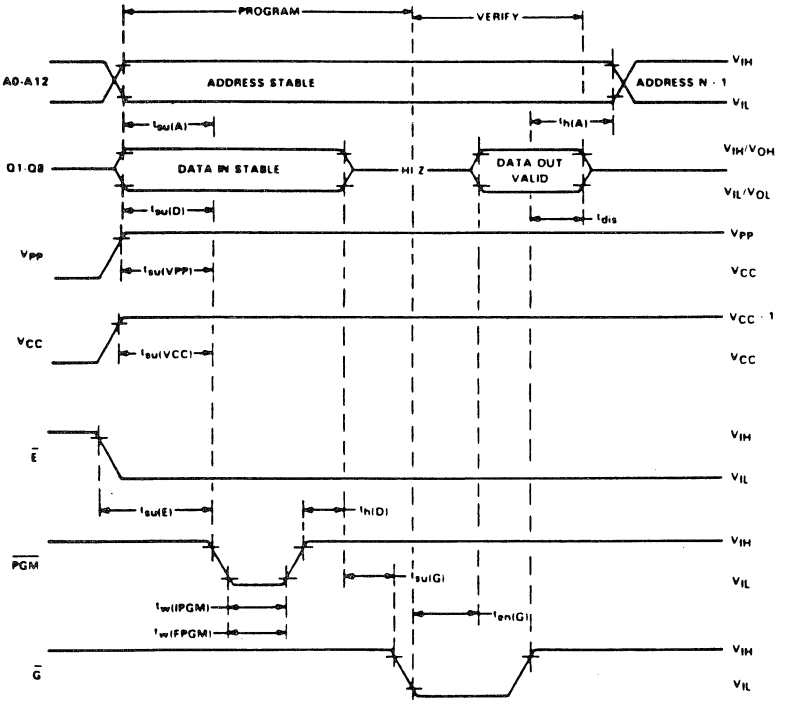
†For all switching characteristics and timing measurements, input pulse levels are 0.40 V to 2.4 V and Vpp = 12.5 ± 0.5 V during programming.
‡Value calculated from 0.5 V delta to measured output level. This parameter is only sampled and not 100% tested.

recommended conditions for programming, T_A = 25°C§

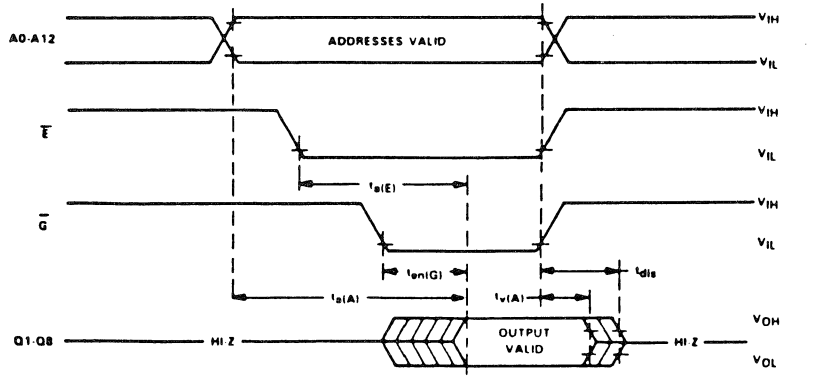
		MIN	NOM	MAX	UNIT
$t_{w(IPGM)}$	Initial program pulse duration	0.95	1	1.05	ms
$t_{w(FPGM)}$	Final pulse duration	2.85		78.75	ms
$t_{su(A)}$	Address setup time	2			µs
$t_{su(E)}$	$\overline{E}$ setup time	2			µs
$t_{su(G)}$	$\overline{G}$ setup time	2			µs
$t_{dis(G)}$	Output disable time from $\overline{G}$	0		130	ns
$t_{en(G)}$	Output enable time from $\overline{G}$			150	ns
$t_{su(D)}$	Data setup time	2			µs
$t_{su(VPP)}$	Vpp setup time	2			µs
$t_{su(VCC)}$	V_{CC} setup time	2			µs
$t_{h(A)}$	Address hold time	0			µs
$t_{h(D)}$	Data hold time	2			µs

§Common test conditions apply for $t_{dis(G)}$ except during programming.

program cycle timing



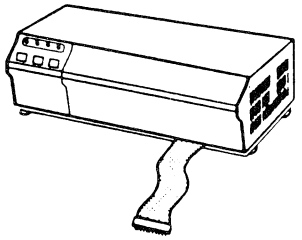
read cycle timing



NOTE: The timing measurement reference levels for inputs and outputs are 0.8 V and 2 V.

TMS77C82 DEVELOPMENT SUPPORT

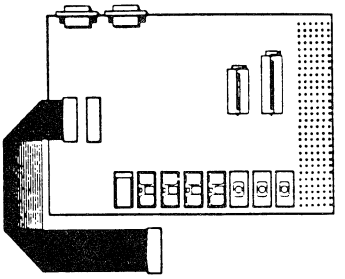
TMS7000 CMOS XDS† — extended development system



- Full TMS7000 Family Development System
- Host Independent/RS-232-C Interface
- Full Speed In-Circuit Emulation
- Extensive Breakpoint and Trace Functions
  - Detailed Timing Analysis
  - 2K-Byte Trace Samples
  - Breakpoint Sequencing Ability
- Command/Default Storage
- Removable Target Connector
- External Probe for Breakpoint/ Trace Qualifiers
- On-Board Assembler and reverse Assembler
- Multiprocessing Capabilities

CMOS EVM — evaluation module RTC/EVM7000C-1

- TMS7000 Family Low Cost Development System
- Single-Chip Mode Emulation Only
- On-Board Assembler/Line Text Editor
- Multiple Breakpoints
- Trace Display Function
- EPROM Programmer Utilities



assembler/linker packages

Crossware† assembler/linker packages are available through Texas Instruments for the following operating systems:

Operating System	TI Part Number
MS†-DOS and PC-DOS	TMDS7040810-02
DEC VAX† VMS	TMDS7040210-08

†XDS and Crossware are registered trademarks of Texas Instruments Incorporated. MS is a trademark of Microsoft, Inc. VAX is a trademark of Digital Equipment Corporation.

EPROM programming support

The following third-party companies support programming the TMS77C82 EPROM microcontroller directly or with an adapter socket (see note 24).

- Data I/O Corporation

  10525 Willows Road N.E.
  P.O. Box 97046
  Redmond, WA 98073-9746
  (206) 881-6444

- PROMAC

  Adams MacDonald Enterprises, Inc.
  2999 Monterey/Salinas Highway
  Monterey, CA 93940
  (408) 373-3607

- Advanced Microcomputer System Inc.

  2780 S.W. 14th Street
  Pompano Beach, Fl 33069
  (305) 975-9515

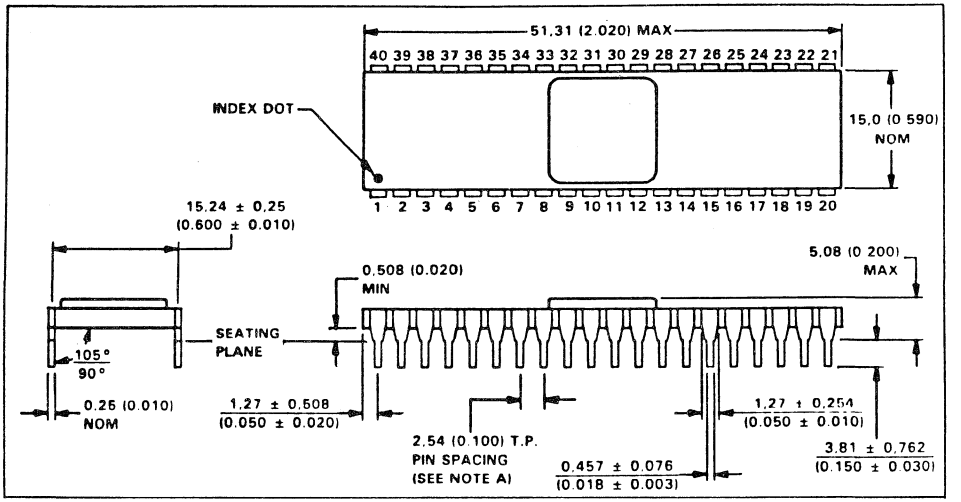
- Logical Devices, Inc.

  1321-E N.W. 65th Place
  Fort Lauderdale, FL 33309

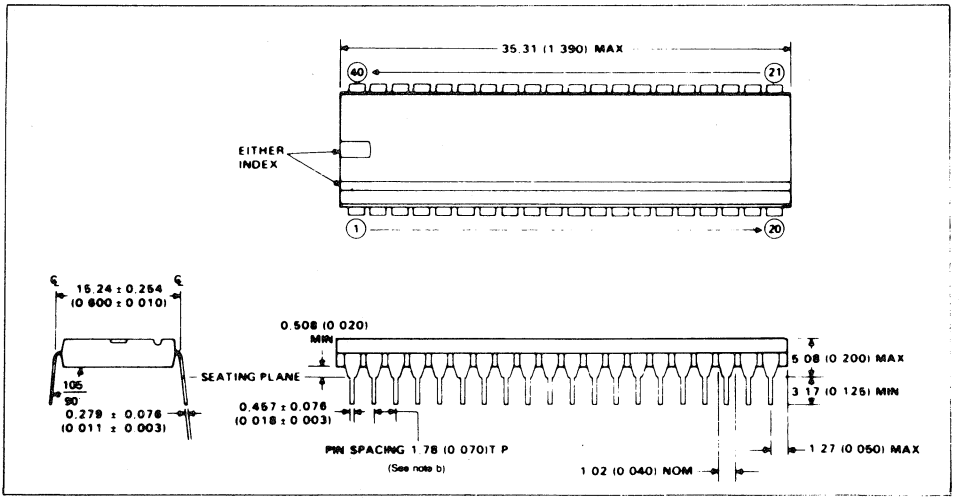
NOTE: Contact your local Texas Instruments field sales office for availability of the 40-to-28 pin programmer adapter socket (part no. RTC/PGMC82A-06).

MECHANICAL DATA

40-pin JD ceramic sidebraze package



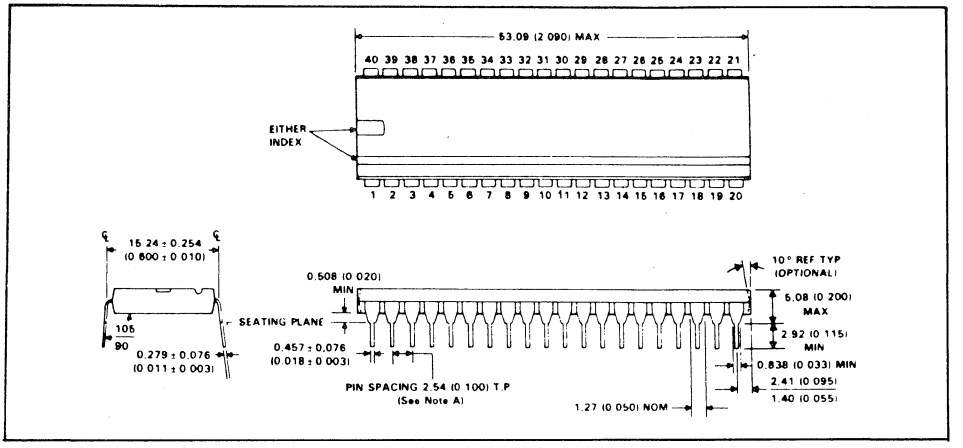
40-pin plastic package (70 mil spacing)



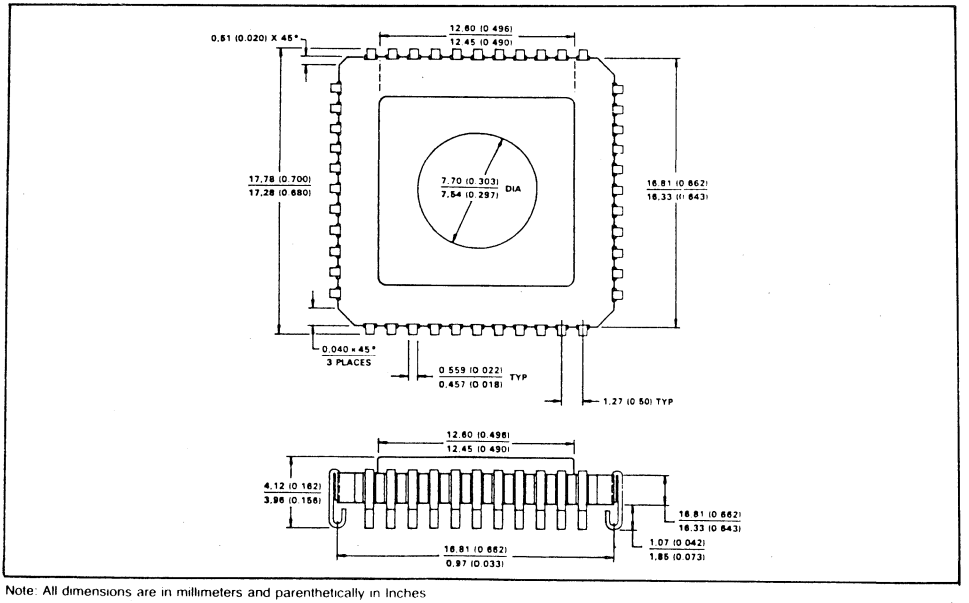
Note: All dimensions are in millimeters and parenthetically in inches

MECHANICAL DATA

40-pin N dual-in-line package



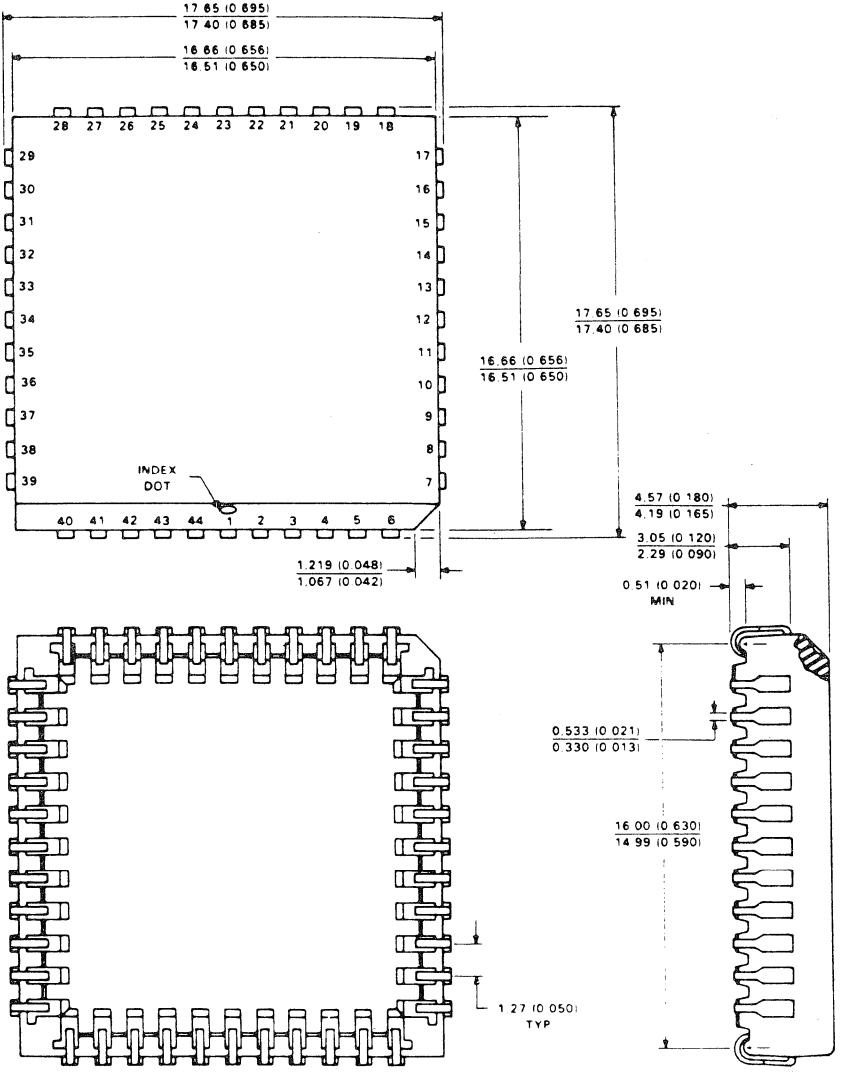
44-pin leaded chip carrier with window



Note: All dimensions are in millimeters and parenthetically in inches

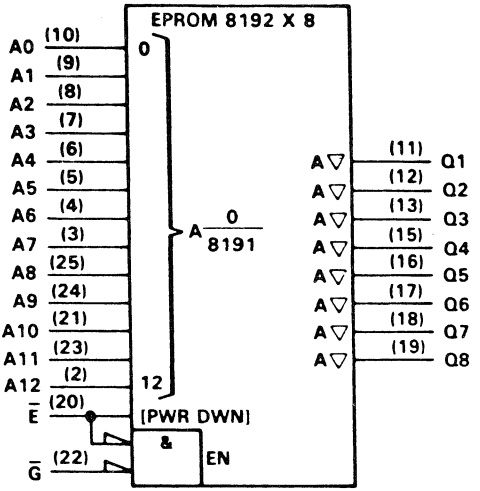
MECHANICAL DATA

44-lead plastic chip carrier package



ALL DIMENSIONS ARE IN MILLIMETERS AND PARENTHETICALLY IN INCHES

logic symbol†



†This symbol is in accordance with ANSI/IEEE Std 91-1984 and IEC Publication 617-12.

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage range, V_{CC} (see Note 1)	−0.6 V to 7 V
Supply voltage range, V_{PP} (see Note 1)	−0.6 V to 14 V
Input voltage range (see Note 1): All inputs except A9	−0.6 V to 6.5 V
A9	−0.6 V to 13.5 V
Output voltage range (see Note 1)	−0.6 V to V_{CC} + 1 V
Operating free-air temperature range	0°C to 70°C
Storage temperature range	−65°C to 150°C

†Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the "Recommended Operating Conditions" section of this specification is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: Under absolute maximum ratings, voltage values are with respect to GND.

recommended operating conditions

			TMS27C64-1 TMS27C64-2 TMS27C64 TMS27C64-3 TMS27C64-4			TMS27C64-15 TMS27C64-20 TMS27C64-25 TMS27C64-30 TMS27C64-45			UNIT
			MIN	NOM	MAX	MIN	NOM	MAX	
V_{CC}	Supply voltage (see Note 2)		4.75	5	5.25	4.5	5	5.5	V
V_{PP}	Supply voltage (see Note 3)			V_{CC}			V_{CC}		V
V_{IH}	High-level input voltage	TTL	2		$V_{CC}+1$	2		$V_{CC}+1$	V
		CMOS	$V_{CC}-0.2$		$V_{CC}+0.2$	$V_{CC}-0.2$		$V_{CC}+0.2$	V
V_{IL}	Low-level input voltage	TTL	−0.5		0.8	−0.5		0.8	V
		CMOS	GND−0.2		GND+0.2	GND−0.2		GND+0.2	V
T_A	Operating free-air temperature		0		70	0		70	°C

NOTES: 2. V_{CC} must be applied before or at the same time as V_{PP} and removed after or at the same time as V_{PP}. The device must not be inserted into or removed from the board when V_{PP} or V_{CC} is applied.

3. V_{PP} can be connected to V_{CC} directly (except in the program mode). V_{CC} supply current in this case would be $I_{CC}+I_{PP}$. During programming, V_{PP} must be maintained at 12.5 V (±0.5 V).

electrical characteristics over full ranges of recommended operating conditions

PARAMETER			TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage		I_{OH} = −400 µA	2.4			V
V_{OL}	Low-level output voltage		I_{OL} = 2.1 mA			0.4	V
I_I	Input current (leakage)		V_I = 0 V to 5.5 V			±10	µA
I_O	Output current (leakage)		V_O = 0 V to V_{CC}			±10	µA
I_{PP1}	V_{PP} supply current		V_{PP} = V_{CC} = 5.5 V			100	µA
I_{PP2}	V_{PP} supply current (during program pulse)		V_{PP} = 13 V		30	50	mA
I_{CC1}	V_{CC} supply current (standby)	TTL-input level	V_{CC} = 5.5 V, $\overline{E}$ = V_{IH}			500	µA
		CMOS-input level	V_{CC} = 5.5 V, $\overline{E}$ = V_{CC}			250	µA
I_{CC2}	V_{CC} supply current (active)		V_{CC} = 5.5 V, $\overline{E}$ = V_{IL}, t_{cycle} = minimum cycle time, outputs open		30	40	mA

†Typical values are at T_A = 25°C and nominal voltages.

capacitance over recommended supply voltage range and operating free-air temperature range, f = 1 MHz†

PARAMETER		TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
C_i	Input capacitance	V_I = 0 V, f = 1 MHz		6	9	pF
C_o	Output capacitance	V_O = 0 V, f = 1 MHz		8	12	pF

†Capacitance measurements are made on sample basis only.
‡Typical values are at T_A = 25°C and nominal voltages.

switching characteristics over full ranges of recommended operating conditions (see Note 4)

PARAMETER		TEST CONDITIONS (SEE NOTES 4 AND 5)	'27C64-1 '27C64-15		'27C64-2 '27C64-20		'27C64 '27C64-25		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
$t_{a(A)}$	Access time from address	C_L = 100 pF, 1 Series 74 TTL Load, Input $t_r \leq 20$ ns, Input $t_f \leq 20$ ns		150		200		250	ns
$t_{a(E)}$	Access time from chip enable			150		200		250	ns
$t_{en(G)}$	Output enable time from $\overline{G}$			75		75		100	ns
t_{dis}	Output disable time from $\overline{G}$ or $\overline{E}$, whichever occurs first†		0	60	0	60	0	60	ns
$t_{v(A)}$	Output data valid time after change of address, $\overline{E}$, or $\overline{G}$, whichever occurs first†		0		0		0		ns

PARAMETER		TEST CONDITIONS (SEE NOTES 4 AND 5)	'27C64-3 '27C64-30		'27C64-4 '27C64-45		UNIT
			MIN	MAX	MIN	MAX	
$t_{a(A)}$	Access time from address	C_L = 100 pF, 1 Series 74 TTL Load, Input $t_r \leq 20$ ns, Input $t_f \leq 20$ ns		300		450	ns
$t_{a(E)}$	Access time from chip enable			300		450	ns
$t_{en(G)}$	Output enable time from $\overline{G}$			120		150	ns
t_{dis}	Output disable time from $\overline{G}$ or $\overline{E}$, whichever occurs first†		0	105	0	130	ns
$t_{v(A)}$	Output data valid time after change of address, $\overline{E}$, or $\overline{G}$, whichever occurs first†		0		0		ns

†Value calculated from 0.5 V delta to measured output level. This parameter is only sampled and not 100% tested.

recommended timing requirements for programming, T_A = 25°C, V_{CC} = 6 V, V_{PP} = 12.5 V (see Note 4)

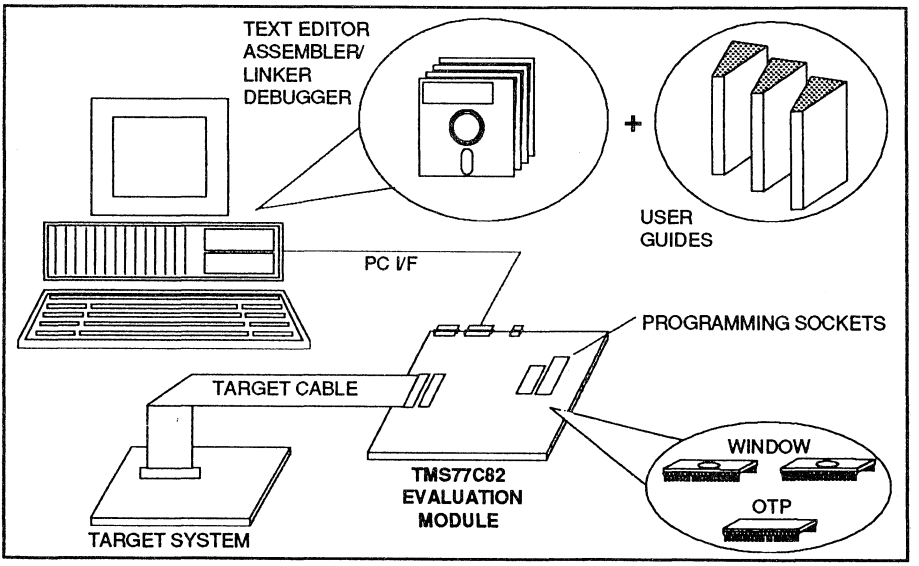
		MIN	NOM	MAX	UNIT
$t_{w(IPGM)}$	Initial program pulse duration	0.95	1	1.05	ms
$t_{w(FPGM)}$	Final pulse duration	2.85		78.75	ms
$t_{su(A)}$	Address setup time	2			µs
$t_{su(E)}$	$\overline{E}$ setup time	2			µs
$t_{su(G)}$	$\overline{G}$ setup time	2			µs
$t_{dis(G)}$	Output disable time from $\overline{G}$	0		130	ns
$t_{en(G)}$	Output enable time from $\overline{G}$			150	ns
$t_{su(D)}$	Data setup time	2			µs
$t_{su(VPP)}$	V_{PP} setup time	2			µs
$t_{su(VCC)}$	V_{CC} setup time	2			µs
$t_{h(A)}$	Address hold time	0			µs
$t_{h(D)}$	Data hold time	2			µs

NOTES: 4. For all switching characteristics and timing measurements, input pulse levels are 0.40 V to 2.4 V and V_{PP} = 12.5 V ± 0.5 V during programming.

5. Common test conditions apply for $t_{dis(G)}$ except during programming.

77C82 KIT EVM
TMS77C82 Evaluation Kit

- Complete Development Solution in One Package.
- Unpack and Begin - Everything is Provided.
- Interactive, Windowed Debugger.
- Optimized User Interface Enhances Productivity.
- Real-Time Emulation.
- When Only Real-Time Will Do.
- Integrated EPROM Programmer.
- Quick and Easy Way to Generate Test EPROM
- Shorter Development Cycle.
- Faster to Market.
- Stand-Alone or PC Modes for EVM Card.
- Flexibility and Independ ence.
- PC Mode Functions with any IBM XT/AT or Compatible.
- Standard RS232 Serial Link to Host Computer.



TEXAS INSTRUMENTS

About the 77C82 KIT EVM

The 77C82 KIT EVM offers a low cost but highly efficient route to TMS7000 single-chip microcontroller development. Features such as the new interactive windowed debugger, real-time emulation and integrated EPROM programmer, all contribute to enhanced user productivity and consequently a shorter design cycle.

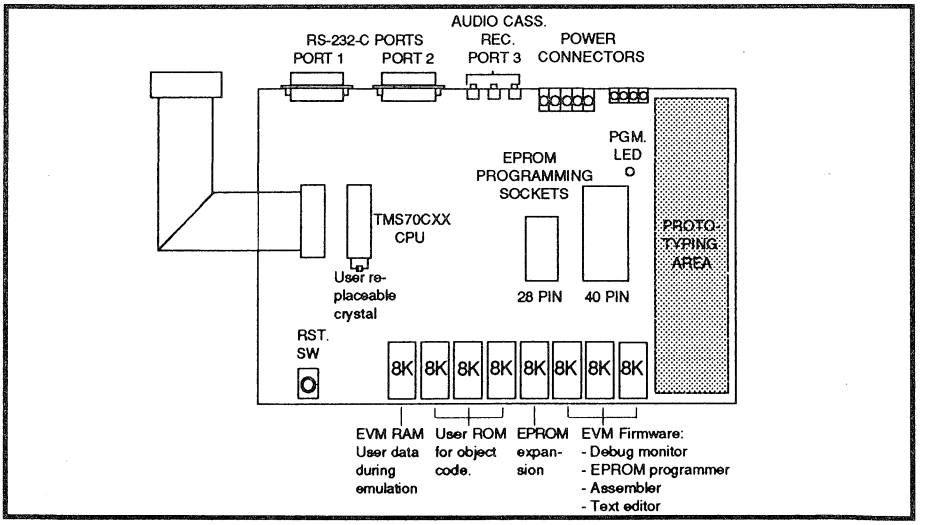
The TMS77C82 Evaluation Module Kit (Part Number - CMF 179) contains:

- EVM77C82 Board.
- Assembler/ Linker for IBM XT/AT or compatible.
- New, windowed debugger for IBM XT/AT or compatible.
- Complete supporting documentation.
- 3 TMS77C82 devices - (1 OTP + 2 Windowed).
- Target Cable.

Once the 77C82 KIT EVM is unpacked, application development can start immediately. Everything required to emulate the single-chip mode of the TMS7000 CMOS family is provided. The EVM77C82 evaluation module needs only to be connected to an external power supply and a terminal.

I/O Ports

The Evaluation Module firmware supports three ports for the purpose of loading and dumping data (text and object code). Ports 1 & 2 conform to the EIA RS-232-C standard and Port 3 provides an audio cassette recorder interface. Port 1 is for a user terminal connection and Port 2 handles either the host CPU link or a printer. The UART function for Ports 1 & 2 is implemented by the evaluation module firmware and supports the following Baud rates: 110; 150; 300; 600; 1200; 2400; 4800; 9600. The Baud rate of port 1 is determined at



EVM77C82 - Major Functional Areas

power-up automatically, by hitting Carrage Return on the terminal keyboard after pressing the **RESET** key on the evaluation module. This feature is known as Autobaud and eliminates the need to select Baud rates. The Baud rate of Port 2 defaults to 9600 at RESET and the Baud rates of both ports may be changed by means of a simple monitor command.

EPROM Programmer

The Evaluation Module contains resident firmware providing EPROM programming facilities. A group of eight simple commands perform PROGRAM, COMPARE READ and VERIFY operations. Each command has an associated destination parameter that specifies the device that is to be programmed. This parameter enables an out-of-range address check of the EPROM start and stop locations. On-card 28 pin and 40 pin programming sockets accomodate most popular types of EPROM devices. Of course, the EVM module will also program the TMS77C82. An LED adjacent to the programming sockets illuminates when the programming voltage is being applied.

Prototyping Area

A wire-wrap pin prototyping area is provided at one side of the Evaluation Module to facilitate development of user applications. All address, data and control lines as well as power supplies, are brought out to clearly marked pins situated at the edge of the prototyping area.

Crystal Frequency

Since the EVM is intended to be a development tool, using the emulation cable, the crystal frequency of the EVM can be altered to suit the needs of the target system.

Power Supply

Two sizes of screw terminal power supply connectors are provided for + 12V, - 12V, + 5V and Ground.

Operational Modes

For maximum flexibiity, the EVM77C82 board has two modes of operation:

- Peripheral (PC) Mode.
- Standalone Mode.

The EVM77C82 can function as a peripheral to a host computer, or in standalone mode as a competely independant development system using an audio cassette as a mass storage device.

Peripheral (PC) Mode

Operating in Peripheral Mode takes advantage of the sophisticated mass storage facilities offered by a host computer. Text files may be downloaded to the Evaluation Module via the EIA port and then assembled by the powerful resident two pass assembler. After assembly, the resultant object file can be subsequently uploaded to the host.

Stand-alone Mode

Stand-alone mode requires no external computer support for the EVM to function as a development system. The file structured cassette tape interface (with file search) is the mass-storage medium. The

resident Text Editor provides for creation and modification of large text files that can be passed directly to the asembler. A unique set of monitor commands allow breakpoints to be set, based on Text Editor line numbers (such numbers are easy to remember and are quickly displayed from the Text Editor as well as the monitor).

Operating System

The EVM operating system firmware resides in 24K bytes of EPROM and comprises three major parts:

- Interactive Debug monitor and EPROM programmer.
- Text editor.
- Assembler/Linker.

All the software is designed to interact with the user to provide a complete, powerful, and easy to use development tool.

The menu driven nature of the software enables the user to exploit the features of the Evaluation Module at maximum efficiency. Optimization of the man-machine interface is achieved through an enhanced level of communication, resulting in better user productivity.

Main features of the Operating System are:

- Windows used for general information.
- Menu-driven instructions with macros
- Cell and editor commands.
- Symbolic disassembler.
- Symbolic access to all instructions.
- Wide choice of display format.
- Windowed editing.
- Step-by-step execution.
- 10 breakpoints with diskette backup.
- Terminal emulator mode for driving the Evaluation Module Directly.
- Temporary return to DOS.

When performing assembly and debug operations, the Evaluation Module RAM can be configured to emulate all TMS7000 family members. The Evaluation Module software even allows assembly of text files from RAM, leaving the text intact for immediate editing after execution. After assembly of the text editor output, breakpoints can be set based on either addresses or line numbers.

During execution, several modes of fixed display are available - a hex display of all register and peripheral files for example, or a binary display of the peripheral ports. While using a fixed display, subsequent execution to a breakpoint or execution of a single instruction step will overwritethe old data on the screen with new data. A programmable line of up to six registers or peripheral locations is provided for display with breakpoints and instruction steps.

The text editor is cursor and line number orientated with auto-increment line numbers, resequence line numbers, change line number, duplicate line number and find string commands. The cursor orientated edit capablility simulates a screen editor by allowing editing of the previous or next line simply by moving the cursor up or down.

TMS7000 assembly language is processed by a two-pass macro assembler. During the first pass the assembler:

- Updates the location counter.
- Builds a symbol table.

- Produces a copy of the source code.

Then in the second pass the assembler:

- Reads the source code copy.
- Assembles the object code using the opcodes and symbol table produced in the first pass.

The assembler/Linker supports a number of assembler directives that perform functions such as modifying the location counter; initializing constants; linking program modules; etc...

The TMS7000 Interactive Debugger

Once an executable object module has been produced by the TMS7000 assembler and linker, the debugger is used to load it into emulator memory and execute it

Debugging a system may require intervenion in a number of different areas: the code being executed; the registers of the target machine; the variables in the progam; etc. The TMS7000 Debugger displays this data in a set of windows on the screen. Each window contains information pertinent to one aspect of the debugging process. The user can move from window to window to perform specific operations such as moving to the code window to examine code or moving to the CPU register window to clear a register.

In addition to the various windows, the top line of the entry level debugging screen shows a menu of active command options.

```
 Display    Modify    Execute    Reset    Save    Load    Terminal    Quit
Command : Execute step
================ code ==================||============ cpu registers ============
   FCS0 E0AB       JMP  IREC            ||     PC   FC88 SP   78    ST   cnzi
 TRT2   A44006     ORP  %>40,P006       ||     A    FF   B    01    40   0100
   FC85 A41006     ORP  %>10,P006       ||====== r file =======||====== stack ====
   FC88 2207       MOV  %>07,A          || R00  FFh  R01  01h  ||  SP   (78)00
   FC8A 8EFDO6     CALL @WW             || R02  00h  R00  FFh  ||  - 1  (77) 20
   FC8D E09E       JMP  IREC            || R04  F3h  R05  00h  ||  - 2  (76) 01
 TRTJ   A3BF06     ANDP%>BF,P006        || R06  00h  R07  00h  ||  - 3  (75) 00
   FC92 A3EF06     ANDP%>EF,P006        || R06  00h  R09  00h  ||  - 4  (74) 00
   FC95 2201       MOV  %>01,A          || R0A  00h  R0B  00h  ||  - 5  (73) 00
   FC97 8EFDO6     CALL @WW             || R0C  00h  R0   F5h  ||  - 6  (72) 00
=============== display ================|| R0E  00h  R0F  00h  ||  - 7  (71) 00
0000      FF 01 00 FF FJ 00 00 00 ........ || R10  00h  R11  00h  ||  - 8  (70) 00
0008      00 00 00 00 00 F5 00 00 ........ || R12  00h  R13  00H  ||  - 9  (6F) 00
0010      00 00 00 00 00 00 00 00 ........ ||========== eprom programmer ===========
0018      00 00 00 00 00 00 F5 00 ........ || selected device  :  TMS77c82
0020      00 00 00 00 00 00 00 00 ........ ||
0028      00 00 00 00 00 00 00 00 ........ ||
0030      00 00 00 00 00 00 00 00 ........ ||
0038      00 00 00 00 00 00 00 00 ........ ||
0040      00 00 00 00 00 00 00 00 ........ ||
0048      00 00 00 00 00 00 00 00 ........ ||

F1Window+  F2Window-  F3Edit  F4SetPc  F5Update  F6SetBrkp  F9Cmd +/-  F10Help
```

Interactive Debugger - Top Level Screen

TEXAS INSTRUMENTS

In the second line, the chosen option is combined with a suitable parameter to form a command. The bottom line shows the active function key assignements.

The windows are designated:

Code	Display	CPU Registers
R file	Stack	EPROM Programmer

The windows are based on virtual buffers in the debugger. The debugger therefore keeps track of more information than is actually displayed in a given window. This allows the user to scroll the window up or down very quickly because the debugger does not need to request the required data from the Evaluation Module. The windows are automatically updated whenever the microcontroller stops running, or by the modify command. When the values change in a window containing register or memory information, the new values are highlighted so that they are easily identified on the screen.

Debugger Command Overview

Display Memory	Display memory dump
Display Rfile	Display register file
Display Pfile	Display peripheral file
Display Stack	Display stack
Display code	Display disassembly code
Display Labels	Display symbols
Display Breakpoint	Display breakpoint addresses
Display File	Display DOS file
Display Trace	Display instruction trace buffer
Modify Memory	Modify memory location(s)
Modify Blockmem	Modify memory block
Modify Register	Modify register
Modify Preg	Modify peripheral register
Modify Breakpoint	Modify breakpoint setting
Execute	Execute code
Execute From	Execute code from address
Execute Until	Execute code until address
Execute Step	Single step
Execute Cmdfile	Execute Command file
Execute Promprog	Execute EPROM programmer utility
Reset	Set PC to reset vector
Reset Device	Set PC to reset vector
Reset breaKpoint	Reset breakpoint
Reset Allbrkpt	Reset all breakpoints
Save breaKpoints	Save current breakpoint settings
Save Commands	Save executed debugger commands
Save Scrcolor	Save current screen color settings
Save Memory	Save EVM memory
Load Memory	Load EVM memory
Load Labels	Load symbol table
Load breaKpoints	Load breakpoint settings
Load Scrcolor	Load screen color settings
Quit Definitively	Exit debugger
Quit Temporary	Escape debugger and load DOS

PART # CMF 179

TI Sales Offices

BELGIQUE/BELGIË
S.A. Texas Instruments Belgium N.V.
11, Avenue Jules Bordetlaan 11,
1140 Bruxelles/Brussel
Tel : (02) 242 30 80
Telex : 61161 TEXBEL

DANMARK
Texas Instruments A/S
Borupvang 2D,
DK-2750 Ballerup
Tel : (44) 68 74 00
Telefax : (44) 68 64 00

DEUTSCHLAND
Texas Instruments
Deutschland GmbH.
Haggertystraße 1
8050 Freising
Tel : (08161) 80-0 od. Nbst
Telex : 5 26 529 texin d
Btx : *28050#

Kurfürstendamm 195-196
1000 Berlin 15
Tel : (030) 8 82 73 65
Telex : 5 26 529 texin d

Düsseldorfer Straße 40
6236 Eschborn 1
Tel : (06196) 80 70
Telex : 5 26 529 texin d

Gildehofcenter
Hollestrasse 3
4300 Essen 1
Tel : (0201) 24 25-0
Fax : (0201) 236640
Telex : 5 26 529 texin d

Kirchhorster Straße 2
3000 Hannover 51
Tel : (0511) 64 68-0
Telex : 5 26 529 texin d

Maybachstraße II
7302 Ostfildern 2 (Nellingen)
Tel : (0711) 34 03-0
Telex : 5 26 529 texin d

ESPAÑA
Texas Instruments España S.A.
c/ Gobelas 43.
Ctra de La Coruña km. 14
La Florida
28023 Madrid
Tel : (1) 372 8051
Telex : 23634

C/Diputacion, 279-3-5
08007 Barcelona
Tel : (3) 317 91 80
Telex : 50436
Fax : (3) 301 84 61

FRANCE, Middle East & Africa
Texas Instruments France
8-10 Avenue Morane Saulnier - B.P. 67
78141 Vélizy Villacoublay cedex
Tel : Standard : (1) 30 70 10 03
Service Technique : (1) 30 70 11 33
Telex : 698707 F

HOLLAND
Texas Instruments Holland B.V.
Hogehilweg 19
Postbus 12995
1100 AZ Amsterdam-Zuidoost
Tel : (020) 5602911
Telex : 12196

HUNGARY
Texas Instruments International
Budaörsi u.42, H-1112 Budapest
Tel : (1) 1 66 66 17
Fax : (1) 1 66 61 61
Telex : 2 27 676

ITALIA
Texas Instruments Italia S.p.A.
Centro Direzionale Colleoni
Palazzo Perseo - Via Paracelso, 12
20041 Agrate Brianza (Mi)
Tel : (039) 63221 - Fax : (039) 632299

Via Castello della Magliana, 38
00148 Roma
Tel : (6) 657 2651
Telex : 610587 ROTEX I
Fax : (6) 657 0447

Via Amendola, 17 - 40100 Bologna
Tel : (051) 554004

NORGE
Texas Instruments Norge A/S
PB 106 - Refstad (Sinsenveien 53)
0513 Oslo 5
Tel : (02) 155090

PORTUGAL
Texas Instruments Equipamento
Electronico (Portugal) LDA.
Ing. Frederico Ulricho, 2650
Moreira Da Maia - 4470 Maia
Tel : (2) 948 1003
Telex : 22485

REPUBLIC OF IRELAND
Texas Instruments Ireland Ltd
7/8 Harcourt Street - Dublin 2
Tel : (01) 755233
Telex : 32626

SCHWEIZ/SUISSE
Texas Instruments Switzerland AG
Riedstraße 6
CH-8953 Dietikon
Tel : (01) 74 42 811
Telex : 825 260 TEXIN
Fax : (01) 74 13 357

SUOMI FINLAND
Texas Instruments OY
P.O. Box 86, 02321 Espoo
Tel : (0) 802 6517
Fax : (0) 802 6519
Telex : 121457

SVERIGE
Texas Instruments
International Trade Corporation
(Sverigefilialen)
Box 30, S-164 93 Kista
Visit address : Isafjordsgatan 7, Kista
Tel : (08) 752 58 00
Telefax : (08) 751 97 15
Telex : 10377 SVENTEX S

UNITED KINGDOM
Texas Instruments Ltd.
Manton Lane, Bedford,
England, MK41 7PA
Tel : (0234) 270 111
Telex : 82178

Technical Enquiry Service
Tel : (0234) 223000

Your distributor

TI Regional Technology Centres

DEUTSCHLAND
Texas Instruments
Deutschland GmbH.
Haggertystraße 1
8050 Freising
Tel : (08161) 80 40 43

Frankfurt/Main
Düsseldorfer Straße 40
6236 Eschborn
Tel : (0 61 96) 80 74 18

Kirchhorster Straße 2
3000 Hannover 51
Tel : (0511) 64 80 21

Maybachstraße 11
7302 Ostfildern 2 (Nellingen)
Stuttgart
Tel : (0711) 34 03-0

FRANCE
Centre de Technologie
Texas Instruments France
8-10 Avenue Morane Saulnier, B.P. 67
78141 Vélizy Villacoublay cedex
Tel : Standard : (1) 30 70 10 03
Service Technique : (1) 30 70 11 33
Telex : 698707 F

Centre Européen de Développement
et Siège Social
Texas Instruments France
B. P. 5
06271 Villeneuve-Loubet cedex
Tel : 93 22 20 01
Telex : 470127 F

HOLLAND
Texas Instruments Holland B.V.
Hogehilweg 19
Postbus 12995
1100 AZ Amsterdam-Zuidoost
Tel : (020) 5602911
Telex : 12196

ITALIA
Texas Instruments Italia S.p.A.
Centro Direzionale Colleoni
Palazzo Perseo - Via Paracelso, 12
20041 Agrate Brianza (Mi)
Tel : 039-63221
Fax : (039) 632299

SVERIGE
Texas Instruments
International Trade Corporation
(Sverigefilialen)
Box 30
S-164 93 Kista
Isafjordsgatan 7
Tel : (08) 752 58 00
Telefax : (08) 751 97 15
Telex : 10377 SVENTEX

UNITED KINGDOM
Texas Instruments Ltd.
Regional Technology Centre
Manton Lane,
Bedford,
England, MK41 7PA
Tel : (0234) 270 111
Telex : 82178

Technical Enquiry Service
Tel : (0234) 223000

TEXAS INSTRUMENTS