- High-Performance Static CMOS Technology
- Includes the TMS320C2xx Core CPU
  - Object-Compatible With the TMS320C2xx
  - Source-Code-Compatible With TMS320C25
  - Upwardly Compatible With TMS320C5x™ DSP Generation
  - 50-ns Instruction Cycle Time
- Pin-Compatible With TMS320F241 (64-Pin/68-Pin)
- Code-Compatible With TMS320F243 and TMS320F241
- Commercial and Industrial Temperature Available
- Memory
  - 544 Words x 16 Bits of On-Chip Data/Program Dual-Access RAM (DARAM)
  - 4K Words x 16 Bits of On-chip Program ROM
- Event-Manager Module
  - Eight Compare/Pulse-Width Modulation (PWM) Channels
  - Two 16-Bit General-Purpose Timers With Four Modes, Including Continuous Upand Up/Down Counting
  - Three 16-Bit Full Compare Units With Deadband
  - Three Capture Units (Two With Quadrature Encoder-Pulse Interface Capability)
- Single 10-Bit Analog-to-Digital Converter (ADC) Module With 8 Multiplexed Input Channels
- 26 Individually Programmable, Multiplexed General-Purpose I/O (GPIO) Pins
- Phase-Locked-Loop (PLL)-Based Clock
- Watchdog (WD) Timer Module
- Serial Communications Interface (SCI)
- Five External Interrupts (Power Drive Protection, Reset, NMI, and Two Maskable Interrupts)
- Three Power-Down Modes for Low-Power Operation
- Scan-Based Emulation
- Development Tools Available:
  - Texas Instruments (TI) ANSI C Compiler, Assembler/Linker, and C-Source Debugger
  - Full Range of Emulation Products
    - Self-Emulation (XDS510™)
  - Third-Party Digital Motor Control and Fuzzy-Logic Development Support
- 68-Pin PLCC FN Package
- 64-Pin QFP PG Package

description

The TMS320C242 device is a member of the TMS320™ DSP family of digital signal processor (DSP) controllers based on the TMS320C2xx generation of 16-bit fixed-point DSPs. This new generation is optimized for digital motor/motion control applications. The DSP controllers combine the enhanced TMS320™ DSP family architectural design of the C2xx core CPU for low-cost, high-performance processing capabilities and several advanced peripherals optimized for motor/motion control applications. These peripherals include the event manager module, which provides general-purpose timers and PWM registers to generate PWM outputs, and a single,10-bit analog-to-digital converter (ADC), which can perform conversion within 1 µs.

Table of Contents

REVISION HISTORY

REVISION	DATE	PRODUCT STATUS	HIGHLIGHTS
D	September 2000	Production Data	The internal pullup/pulldown information has been corrected for the pins as appropriate. This information can be found in the “Terminal Functions” section of the data sheet. Errors in the memory map have been rectified with respect to reserved and illegal addresses. SPI and CAN register spaces have been marked as “Illegal”. Changes have been made to improve the clarity of the memory maps. The functional diagram depicting the operation of the GPIO pins has been modified to enhance the understanding of pin behavior. The bit positions for the PIRQRn and PIACKRn registers have been corrected in Table 5 to reflect the silicon. The input currents (I_I) have been specified separately for pins with pullup and pulldown. The I_{DD} specification now includes a “Typical” number. The LPM2 current number has been changed from 75 µA to 5 mA. The figure corresponding to “IDLE2 Entry and Exit Timing” for LPM1 mode has been removed. Note that the parameters are identical to LPM0. The “Reset Timing” figure now includes the state of the GPIO pins for “Power-on” reset. Table 14, “Differences Between the TMS320F241 and the TMS320C242 Devices”, has been added to aid migration from Flash to ROM designs.

device features

Table 1 and Table 2 provide a comparison of the features of the C242 to the F241. See the functional block diagram for the C242 peripherals and memory.

Table 1. Hardware Features of the TMS320x24x DSP Controllers

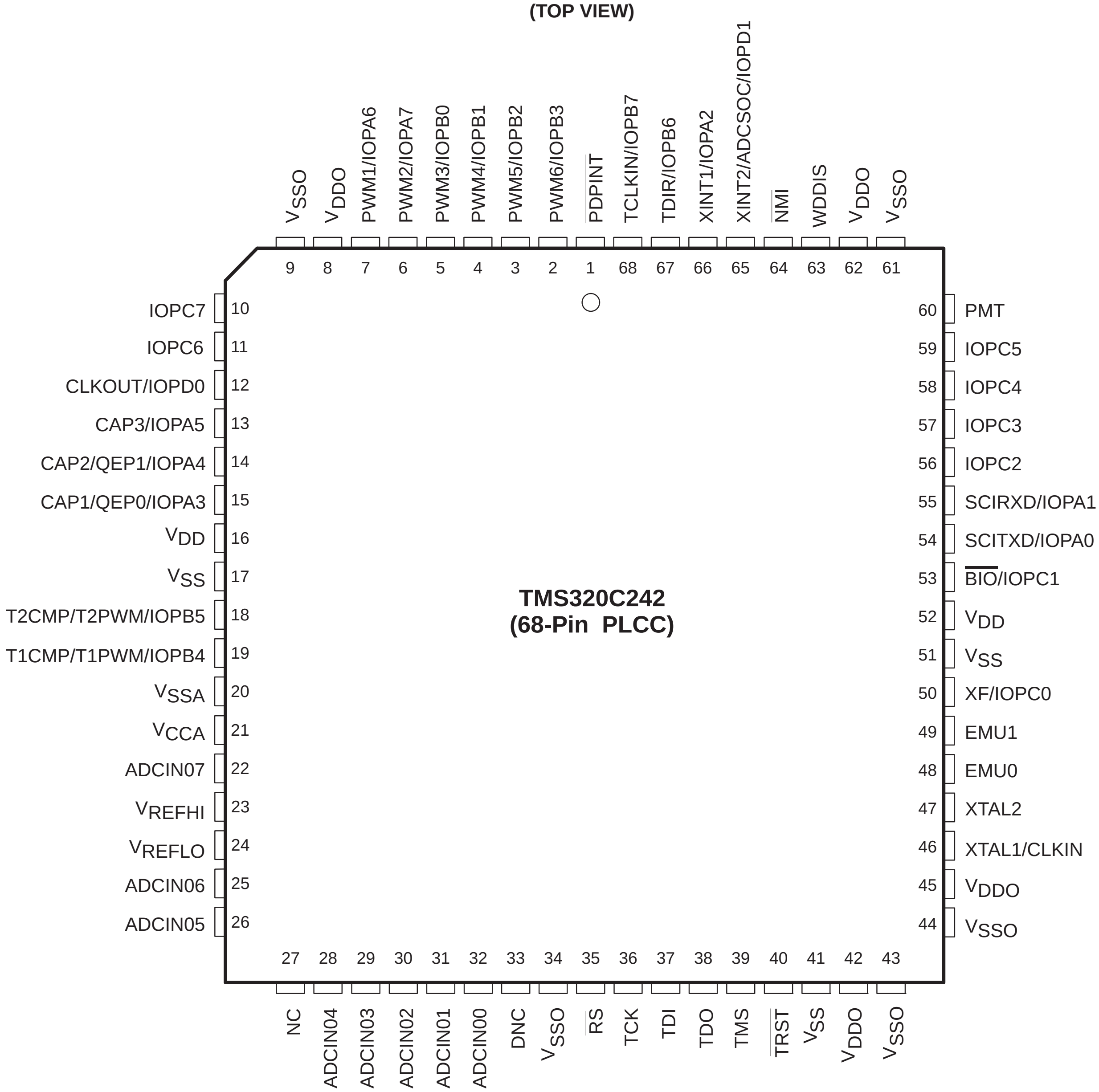
TMS320x24x DEVICES	ON-CHIP MEMORY (WORDS) RAM DATA SPACE (B1 RAM - 256 WORDS) (B2 RAM - 32 WORDS)	ON-CHIP MEMORY (WORDS) RAM CONFIGURABLE DATA / PROG SPACE (B0 RAM)	EXTERNAL MEMORY INTERFACE	POWER SUPPLY (V)	CYCLE TIME (ns)
TMS320C242	288	256	−	5	50
TMS320F241					

Table 2. Device Specifications of the TMS320x24x DSP Controllers

TMS320x24x DEVICES	ON-CHIP MEMORY (WORDS) ROM PROG	ON-CHIP MEMORY (WORDS) FLASH EEPROM PROG	ADC CHANNELS	PERIPHERALS CAN	PERIPHERALS SPI	GPIO	PACKAGE TYPE PIN COUNT
TMS320C242	4K	−	8	−	−	26	FN 68-PLCC PG 64-PQFP
TMS320F241	−	8K		√	√		

pinouts

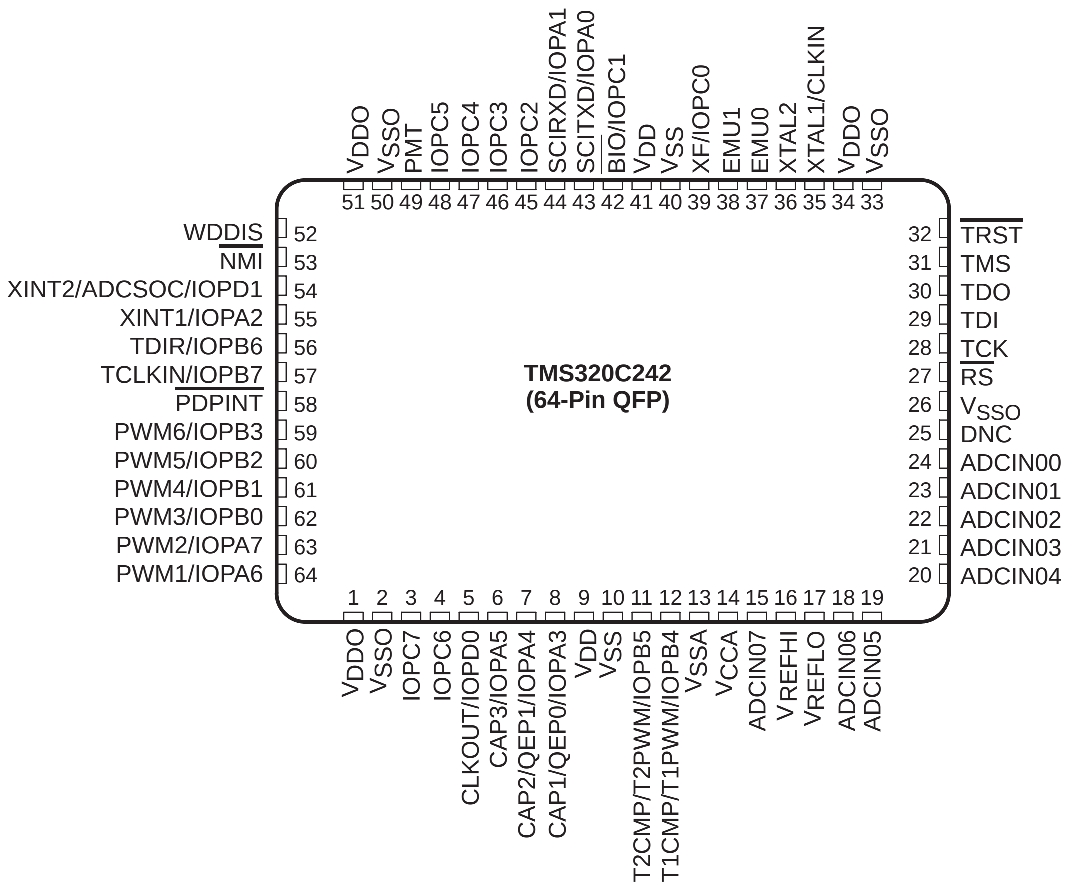
FN PACKAGE†
(TOP VIEW)



† NC = No connection, DNC = Do not connect

pinouts (continued)

PG PACKAGE†
(TOP VIEW)



† NC = No connection, DNC = Do not connect

Terminal Functions - C242 PG and FN Packages

NAME	64-PIN QFP NO.	68-PIN PLCC NO.	TYPE†	RESET STATE‡	DESCRIPTION
INTERFACE CONTROL SIGNALS					
WDDIS	52	63	I	I	Watchdog disable. Note that on ROM devices, only the WDDIS function is valid. If the input is low, the watchdog timer cannot be disabled in the software. If the input is high, the watchdog timer can be disabled in the software through the WDDIS bit in the WDCR register.
ANALOG-TO-DIGITAL CONVERTER (ADC) INPUTS					
ADCIN00	24	32	I	I	Analog inputs to the ADC
ADCIN01	23	31	I	I	Analog inputs to the ADC
ADCIN02	22	30	I	I	Analog inputs to the ADC
ADCIN03	21	29	I	I	Analog inputs to the ADC
ADCIN04	20	28	I	I	Analog inputs to the ADC
ADCIN05	19	26	I	I	Analog inputs to the ADC
ADCIN06	18	25	I	I	Analog inputs to the ADC
ADCIN07	15	22	I	I	Analog inputs to the ADC
V_{CCA}	14	21	−	−	Analog supply voltage for ADC (5 V). It is highly recommended to isolate V_{CCA} from the digital supply voltage (and V_{SSA} from digital ground) to maintain the specified accuracy and improve the noise immunity of the ADC.
V_{SSA}	13	20	−	−	Analog ground reference for ADC
V_{REFHI}	16	23	−	−	ADC analog high-voltage reference input
V_{REFLO}	17	24	−	−	ADC analog low-voltage reference input
EVENT MANAGER					
T1CMP/T1PWM/***IOPB4***§	12	19	I/O/Z	I	Timer 1 compare output/general-purpose bidirectional digital I/O (GPIO). (↓)
T2CMP/T2PWM/***IOPB5***	11	18	I/O/Z	I	Timer 2 compare output/GPIO
TDIR/***IOPB6***§	56	67	I/O	I	Counting direction for GP timer/GPIO. If TDIR=1, upward counting is selected. If TDIR=0, downward counting is selected. (↓)
TCLKIN/***IOPB7***	57	68	I/O	I	External clock input for GP timer/GPIO. Note that timer can also use the internal device clock.
CAP1/QEP0/***IOPA3***§	8	15	I/O	I	Capture input #1/quadrature encoder pulse input #0/GPIO (↓)
CAP2/QEP1/***IOPA4***§	7	14	I/O	I	Capture input #2/quadrature encoder pulse input #1/GPIO (↓)
CAP3/***IOPA5***§	6	13	I/O	I	Capture input #3/GPIO (↓)
PWM1/***IOPA6***	64	7	I/O/Z	I	Compare/PWM output pin #1 or GPIO
PWM2/***IOPA7***	63	6	I/O/Z	I	Compare/PWM output pin #2 or GPIO
PWM3/***IOPB0***	62	5	I/O/Z	I	Compare/PWM output pin #3 or GPIO
PWM4/***IOPB1***	61	4	I/O/Z	I	Compare/PWM output pin #4 or GPIO
PWM5/***IOPB2***	60	3	I/O/Z	I	Compare/PWM output pin #5 or GPIO
PWM6/**IOPB3**	59	2	I/O/Z	I	Compare/PWM output pin #6 or GPIO

† I = input, O = output, Z = high impedance
‡ The reset state indicates the state of the pin at reset. If the pin is an input, indicated by an I, its state is determined by user design. If the pin is an output, its level at reset is indicated.
§ These pins are internally pulled low. However, these pins are not internally pulled low in F243/F241.
¶ These pins are internally pulled high. However, these pins are not internally pulled high in F243/F241.
# The PMT pin should be connected to GND on F243/F241. On the C242, this pin can be left open or connected to GND.
NOTE: ***Bold, italicized pin names*** indicate pin function after reset.
LEGEND: ↑ − Internal pullup ↓ − Internal pulldown (Typical active pullup/pulldown value is 150 µA.)

Terminal Functions - C242 PG and FN Packages (Continued)

NAME	64-PIN QFP NO.	68-PIN PLCC NO.	TYPE†	RESET STATE‡	DESCRIPTION
EVENT MANAGER (CONTINUED)					
PDPINT	58	1	I	I	Power drive protection interrupt input. This interrupt, when activated, puts the PWM output pins in the high-impedance state, should motor drive/power converter abnormalities, such as overvoltage or overcurrent, etc., arise. PDPINT is a falling-edge-sensitive interrupt. After the falling edge, this pin must be held low for two clock cycles for the core to recognize the interrupt.
BIT I/O PINS					
IOPC2§	45	56	I/O	I	GPIO (↓)
IOPC3§	46	57	I/O		GPIO (↓)
IOPC4§	47	58	I/O		GPIO (↓)
IOPC5§	48	59	I/O		GPIO (↓)
IOPC6§	4	11	I/O		GPIO (↓)
IOPC7§	3	10	I/O		GPIO (↓)
SERIAL COMMUNICATIONS INTERFACE (SCI) AND BIT I/O PINS					
SCITXD/***IOPA0***	43	54	I/O	I	SCI asynchronous serial port transmit data or GPIO
SCIRXD/***IOPA1***	44	55	I/O		SCI asynchronous serial port receive data or GPIO
INTERRUPT, EXTERNAL ACCESS, AND MISCELLANEOUS SIGNALS					
RS	27	35	I/O	I	Device reset. RS causes the C242 to terminate execution and sets PC=0. After RS is brought to a high level, execution begins at location zero of program memory. RS affects (sets to zero) various registers and status bits. When the watchdog timer overflows, it initiates a system reset pulse that is reflected on the RS pin. This pulse is eight clock cycles wide.
NMI¶	53	64	I	I	Nonmaskable interrupt. When NMI is activated, the device is interrupted regardless of the state of the INTM bit of the status register. NMI is (falling) edge- and low-level-sensitive. To be recognized by the core, this pin must be kept low for at least one clock cycle after the falling edge. (↑)
XINT1/***IOPA2***§	55	66	I/O	I	External user interrupt 1 or GPIO. Both XINT1 and XINT2 are edge-sensitive. To be recognized by the core, these pins must be kept low/high for at least one clock cycle after the edge. The edge polarity is programmable. (↓)
XINT2/ADCSOC/***IOPD1***	54	65	I/O	I	External user interrupt 2. External “start-of-conversion” input for ADC/GPIO. Both XINT1 and XINT2 are edge-sensitive. To be recognized by the core, these pins must be kept low/high for at least one clock cycle after the edge. The edge polarity is programmable.

† I = input, O = output, Z = high impedance
‡ The reset state indicates the state of the pin at reset. If the pin is an input, indicated by an I, its state is determined by user design. If the pin is an output, its level at reset is indicated.
§ These pins are internally pulled low. However, these pins are not internally pulled low in F243/F241.
¶ These pins are internally pulled high. However, these pins are not internally pulled high in F243/F241.
# The PMT pin should be connected to GND on F243/F241. On the C242, this pin can be left open or connected to GND.
NOTE: ***Bold, italicized pin names*** indicate pin function after reset.
LEGEND: ↑ − Internal pullup ↓ − Internal pulldown (Typical active pullup/pulldown value is 150 μA.)

Terminal Functions - C242 PG and FN Packages (Continued)

NAME	64-PIN QFP NO.	68-PIN PLCC NO.	TYPE†	RESET STATE‡	DESCRIPTION
INTERRUPT, EXTERNAL ACCESS, AND MISCELLANEOUS SIGNALS (CONTINUED)					
XF/IOPC0¶	39	50	I/O	O − 1	External flag output (latched software-programmable signal). XF is a general-purpose output pin. It is set/reset by the `SETC XF/CLRC XF` instruction. This pin is configured as an external flag output by all device resets. It can be used as a GPIO, if not used as XF. (↑)
$\overline{BIO}$/IOPC1¶	42	53	I/O	I	Branch control input. $\overline{\text{BIO}}$ is polled by the `BCND pma,BIO` instruction. If $\overline{\text{BIO}}$ is low, a branch is executed. If $\overline{\text{BIO}}$ is not used, it should be pulled high. This pin is configured as a branch control input by all device resets. It can be used as a GPIO, if not used as a branch control input. (↑)
PMT	49	60	I	I	Do not connect. This pin is not internally connected.# (↓)
CLOCK SIGNALS					
XTAL1/CLKIN	35	46	I	I	PLL oscillator input pin. Crystal input to PLL/clock source input to PLL. XTAL1/CLKIN is tied to one side of a reference crystal.
XTAL2	36	47	O	O	Crystal output. PLL oscillator output pin. XTAL2 is tied to one side of a reference crystal. This pin goes in the high-impedance state when EMU1/$\overline{\text{OFF}}$ is active low.
CLKOUT/IOPD0	5	12	I/O	O	Clock output. This pin outputs the CPU clock (CLKOUT) only. This pin can be used as a GPIO, if it is not used as a clock output pin.
TEST SIGNALS					
TCK	28	36	I	I	JTAG test clock with internal pullup (↑)
TDI	29	37	I	I	JTAG test data input (TDI) with internal pullup. TDI is clocked into the selected register (instruction or data) on a rising edge of TCK. (↑)
TDO	30	38	I/O	O	JTAG scan out, test data output (TDO). The contents of the selected register (instruction or data) is shifted out of TDO on the falling edge of TCK. (↓)
TMS	31	39	I	I	JTAG test-mode select (TMS) with internal pullup. This serial control input is clocked into the TAP controller on the rising edge of TCK. (↑)
$\overline{\text{TRST}}$	32	40	I	I	JTAG test reset with internal pulldown. $\overline{\text{TRST}}$, when driven high, gives the scan system control of the operations of the device. If this signal is not connected or driven low, the device operates in its functional mode, and the test reset signals are ignored. (↓)
EMU0	37	48	I/O	I	Emulator I/O pin 0 with internal pullup. When $\overline{\text{TRST}}$ is driven high, this pin is used as an interrupt to or from the emulator system and is defined as input/output through the JTAG scan. (↑)
EMU1	38	49	I/O	I	Emulator I/O pin 1 with internal pullup. When $\overline{\text{TRST}}$ is driven high, this pin is used as an interrupt to or from the emulator system and is defined as input/output through JTAG scan. (↑)

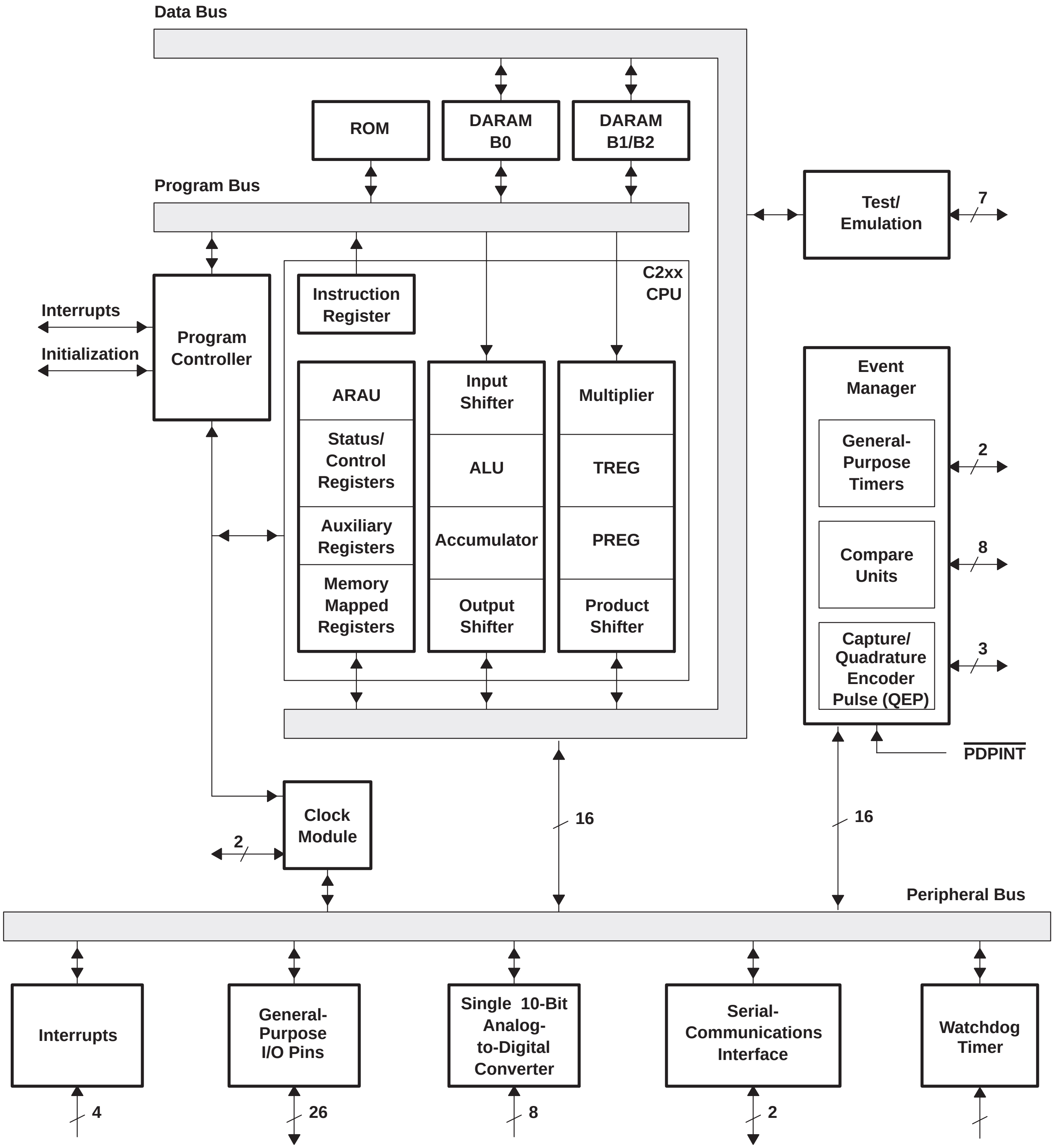
† I = input, O = output, Z = high impedance
‡ The reset state indicates the state of the pin at reset. If the pin is an input, indicated by an I, its state is determined by user design. If the pin is an output, its level at reset is indicated.
§ These pins are internally pulled low. However, these pins are not internally pulled low in F243/F241.
¶ These pins are internally pulled high. However, these pins are not internally pulled high in F243/F241.
# The PMT pin should be connected to GND on F243/F241. On the C242, this pin can be left open or connected to GND.
NOTE: ***Bold, italicized pin names*** indicate pin function after reset.
LEGEND: ↑ − Internal pullup ↓ − Internal pulldown (Typical active pullup/pulldown value is 150 µA.)

Terminal Functions - C242 PG and FN Packages (Continued)

NAME	64-PIN QFP NO.	68-PIN PLCC NO.	TYPE†	RESET STATE‡	DESCRIPTION
SUPPLY SIGNALS					
V_{DD}	9	16	−	−	Digital logic supply voltage (5 V)
	41	52	−	−	
V_{DDO}	−	42	−	−	Digital logic and buffer supply voltage (5 V)
	1	8	−	−	
	34	45	−	−	
	51	62	−	−	
V_{SS}	−	41	−	−	Digital logic ground reference
	10	17	−	−	
	40	51	−	−	
V_{SSO}	−	43	−	−	Digital logic and buffer ground reference
	2	9	−	−	Digital logic and buffer ground reference
	26	34	−	−	
	33	44	−	−	
	50	61	−	−	
NO CONNECT					
NC	−	27			No internal connection made to this pin
DNC	25	33	−	−	Do not connect. Reserved for test.

† I = input, O = output, Z = high impedance
‡ The reset state indicates the state of the pin at reset. If the pin is an input, indicated by an I, its state is determined by user design. If the pin is an output, its level at reset is indicated.
§ These pins are internally pulled low. However, these pins are not internally pulled low in F243/F241.
¶ These pins are internally pulled high. However, these pins are not internally pulled high in F243/F241.
# The PMT pin should be connected to GND on F243/F241. On the C242, this pin can be left open or connected to GND.
NOTE: ***Bold, italicized pin names*** indicate pin function after reset.
LEGEND: ↑ − Internal pullup ↓ − Internal pulldown (Typical active pullup/pulldown value is 150 μA.)

functional block diagram of the 24x DSP controller



architectural overview

The functional block diagram provides a high-level description of each component in the C242 DSP controllers. The TMS320x24x devices are composed of three main functional units: a C2xx DSP core, internal memory, and peripherals. In addition to these three functional units, there are several system-level features of the C242 that are distributed. These system features include the memory map, device reset, interrupts, digital input/output (I/O), clock generation, and low-power operation.

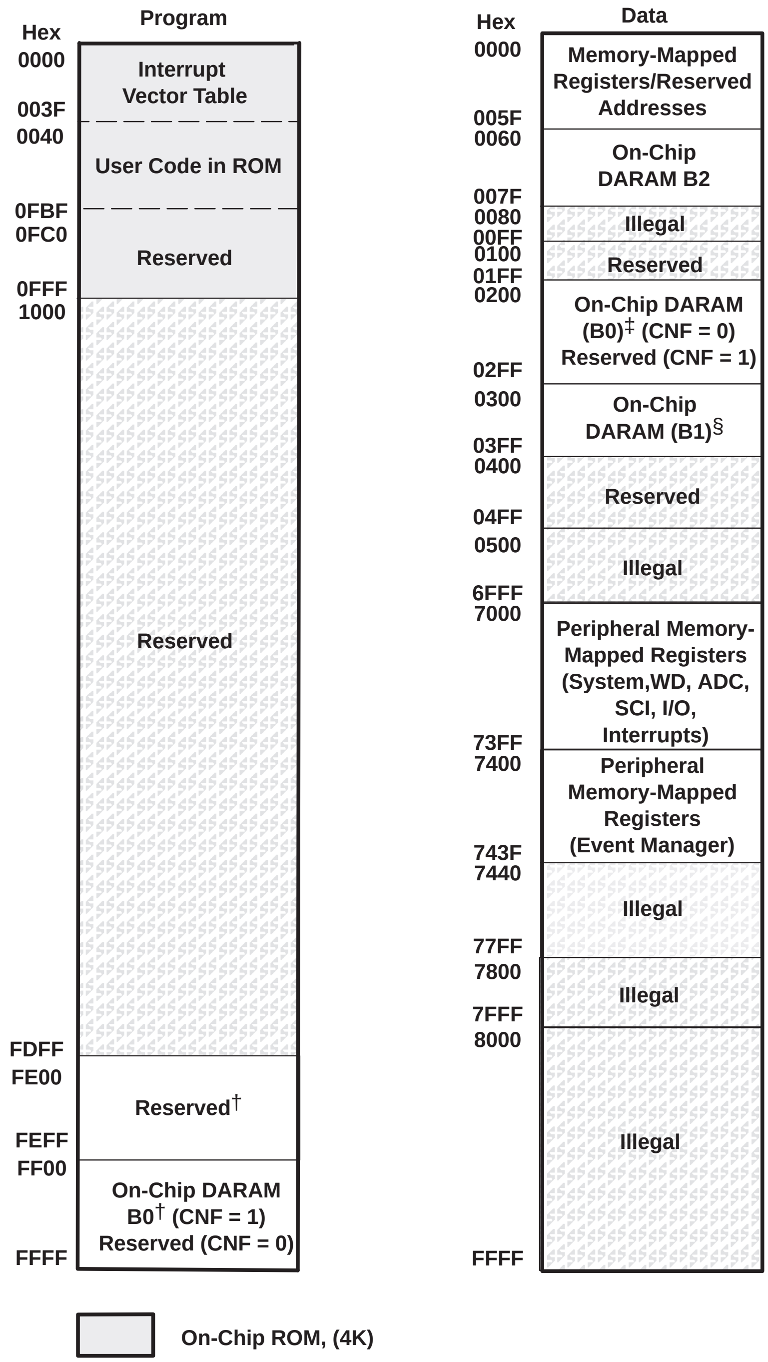
system-level functions

device memory map

The C242 device implements separate address spaces for program memory and data memory. On the C242, the first 96 (0−5Fh) data memory locations are either allocated for memory-mapped registers or reserved. This memory-mapped register space contains various control and status registers, including those for the CPU.

All the on-chip peripherals of the C242 devices are mapped into data memory space. Access to these registers is made by the CPU instructions addressing their data memory locations. Figure 1 shows the C242 memory map.

memory map



† When CNF = 1, addresses FE00h−FEFFh and FF00h−FFFFh are mapped to the same physical block (B0) in program-memory space. For example, a write to FE00h will have the same effect as a write to FF00h. For simplicity, addresses FE00h−FEFFh are referred to as reserved when CNF = 1.

‡ When CNF = 0, addresses 0100h−01FFh and 0200h−02FFh are mapped to the same physical block (B0) in data-memory space. For example, a write to 0100h will have the same effect as a write to 0200h. For simplicity, addresses 0100h−01FFh are referred to as reserved.

§ Addresses 0300h−03FFh and 0400h−04FFh are mapped to the same physical block (B1) in data-memory space. For example, a write to 0400h has the same effect as a write to 0300h. For simplicity, addresses 0400h−04FFh are referred to as reserved.

NOTE A: There is no external memory space for program, data, global data, or I/O in the C242. The GREG register is reserved in the C242.

Figure 1. TMS320C242 Memory Map

peripheral memory map

The system and peripheral control register frame contains all the data, status, and control bits to operate the system and peripheral modules on the device (excluding the event manager). The register frame is mapped in the data memory space.

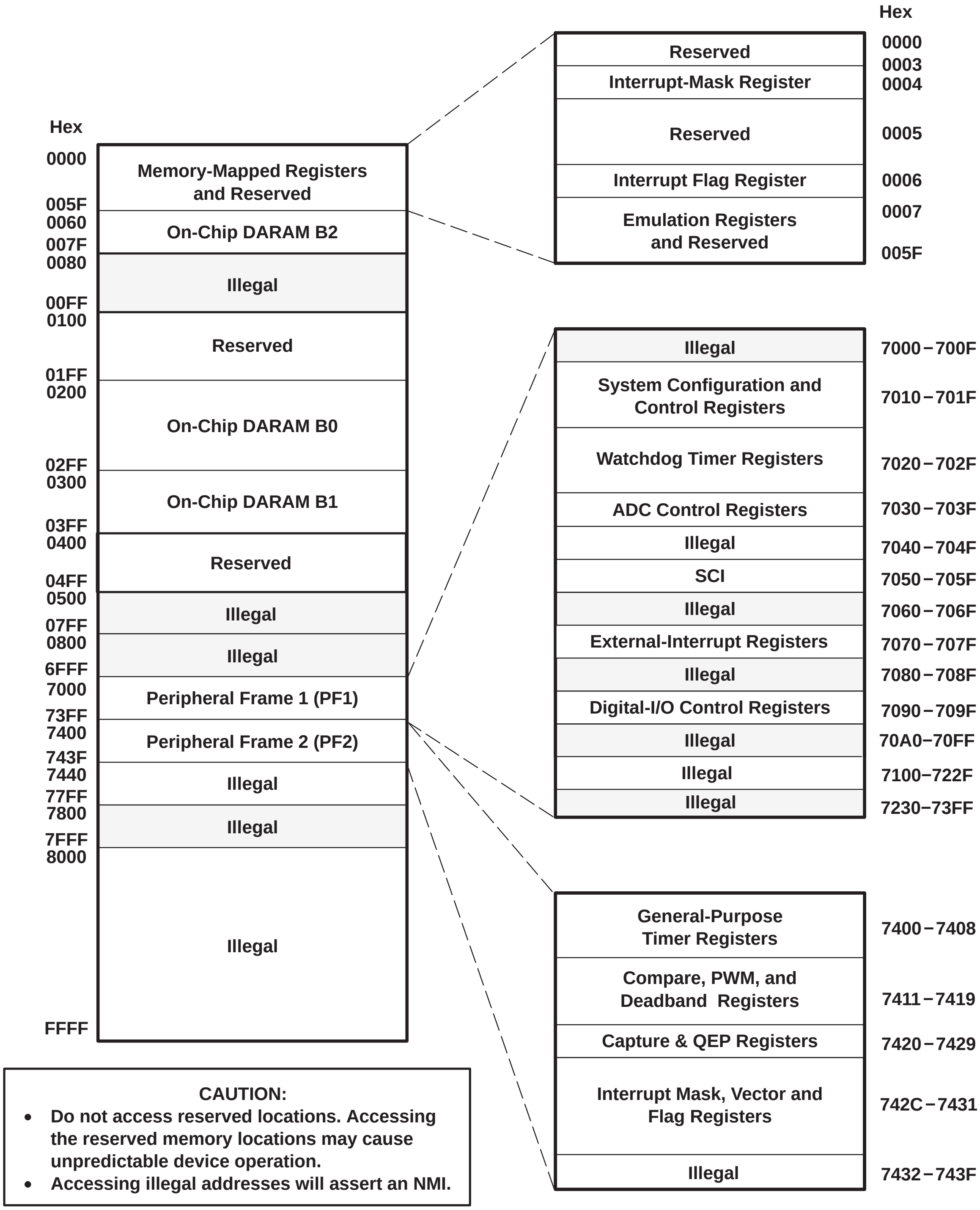


Figure 2. Peripheral Memory Map for C242

digital I/O and shared pin functions

The C242 has a total of 26 general-purpose, bidirectional, digital I/O (GPIO) pins — most of which are shared between primary functions and I/O. Twenty (20) I/O pins of the C242 are shared with other functions. The digital I/O ports module provides a flexible method for controlling both dedicated I/O and shared pin functions. All I/O and shared pin functions are controlled using eight 16-bit registers. These registers are divided into two types:

- Output Control Registers — used to control the multiplexer selection that chooses between the primary function of a pin or the general-purpose I/O function.
- Data and Control Registers — used to control the data and data direction of bidirectional I/O pins.

description of shared I/O pins

The control structure for shared I/O pins is shown in Figure 3, where each pin has three bits that define its operation:

- Mux control bit — this bit selects between the primary function (1) and I/O function (0) of the pin.
- I/O direction bit — if the I/O function is selected for the pin (mux control bit is set to 0), this bit determines whether the pin is an input (0) or an output (1).
- I/O data bit — if the I/O function is selected for the pin (mux control bit is set to 0) and the direction selected is an input, data is read from this bit; if the direction selected is an output, data is written to this bit.

The mux control bit, I/O direction bit, and I/O data bit are in the I/O control registers.

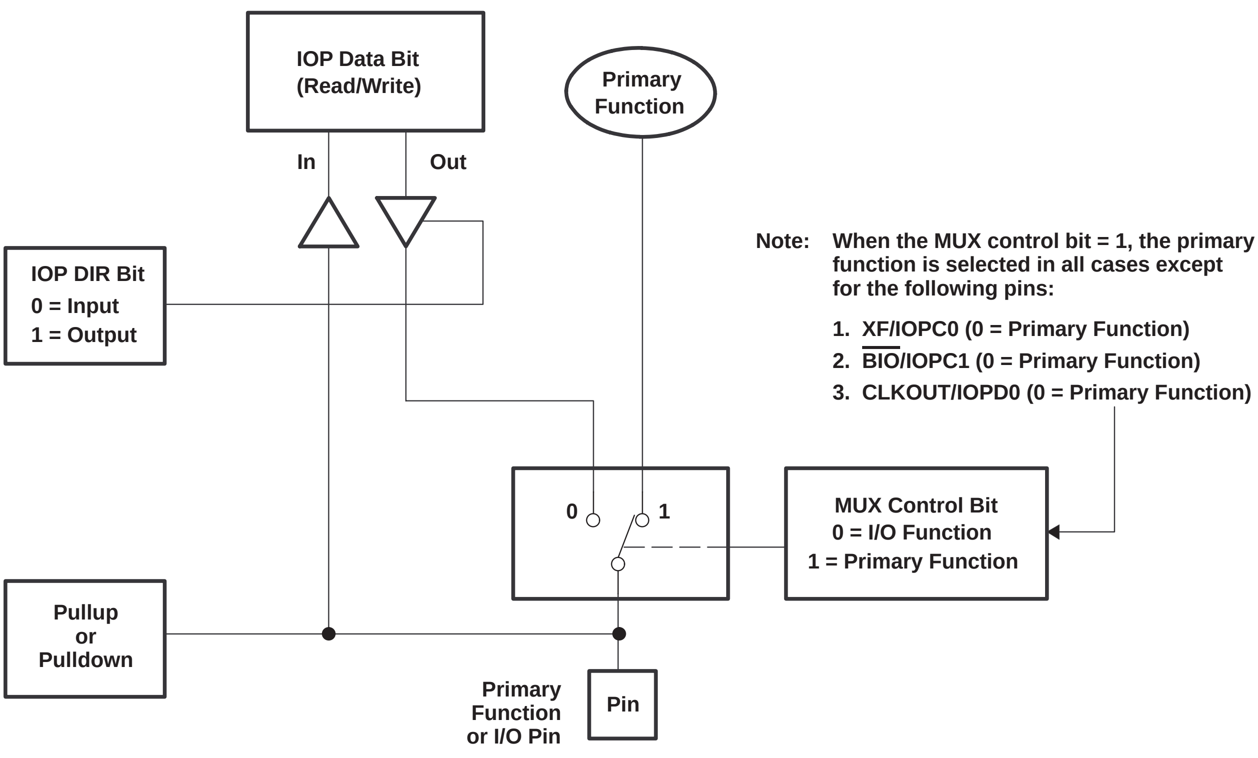


Figure 3. Shared Pin Configuration

A summary of shared pin configurations and associated bits is shown in Table 3.

description of shared I/O pins (continued)

Table 3. Shared Pin Configurations

PIN NO.		MUX CONTROL REGISTER (name.bit #)	PIN FUNCTION SELECTED		I/O PORT DATA AND DIRECTION†		
68-PIN PLCC	64-PIN QFP		(OCRx.n = 1)	(OCRx.n = 0)	REGISTER	DATA BIT NO.‡	DIR BIT NO.§
					PORT A		
54	43	OCRA.0	SCITXD	IOPA0	PADATDIR	0	8
55	44	OCRA.1	SCIRXD	IOPA1	PADATDIR	1	9
66	55	OCRA.2	XINT1	IOPA2	PADATDIR	2	10
15	8	OCRA.3	CAP1/QEP0	IOPA3	PADATDIR	3	11
14	7	OCRA.4	CAP2/QEP1	IOPA4	PADATDIR	4	12
13	6	OCRA.5	CAP3	IOPA5	PADATDIR	5	13
7	64	OCRA.6	PWM1	IOPA6	PADATDIR	6	14
6	63	OCRA.7	PWM2	IOPA7	PADATDIR	7	15
					PORT B		
5	62	OCRA.8	PWM3	IOPB0	PBDATDIR	0	8
4	61	OCRA.9	PWM4	IOPB1	PBDATDIR	1	9
3	60	OCRA.10	PWM5	IOPB2	PBDATDIR	2	10
2	59	OCRA.11	PWM6	IOPB3	PBDATDIR	3	11
19	12	OCRA.12	T1PWM/T1CMP	IOPB4	PBDATDIR	4	12
18	11	OCRA.13	T2PWM/T2CMP	IOPB5	PBDATDIR	5	13
67	56	OCRA.14	TDIR	IOPB6	PBDATDIR	6	14
68	57	OCRA.15	TCLKIN	IOPB7	PBDATDIR	7	15
					PORT C		
50	39	OCRB.0	IOPC0	XF	PCDATDIR	0	8
53	42	OCRB.1	IOPC1	$\overline{\text{BIO}}$	PCDATDIR	1	9
56¶	45¶	OCRB.2	−	IOPC2	PCDATDIR	2	10
57¶	46¶	OCRB.3	−	IOPC3	PCDATDIR	3	11
58¶	47¶	OCRB.4	−	IOPC4	PCDATDIR	4	12
59¶	48¶	OCRB.5	−	IOPC5	PCDATDIR	5	13
11¶	4¶	OCRB.6	−	IOPC6	PCDATDIR	6	14
10¶	3¶	OCRB.7	−	IOPC7	PCDATDIR	7	15
					PORT D		
12	5	OCRB.8	IOPD0	CLKOUT	PDDATDIR	0	8
65	54	OCRB.9	XINT2/ADCSOC	IOPD1	PDDATDIR	1	9

† Valid only if the I/O function is selected on the pin
‡ If the GPIO pin is configured as an output, these bits can be written to. If the pin is configured as an input, these bits are read from.
§ If the DIR bit is 0, the GPIO pin functions as an input. For a value of 1, the pin is configured as an output.
¶ Dedicated I/O pins

digital I/O control registers

Table 4 lists the registers available in the digital I/O module. As with other C242 peripherals, the registers are memory-mapped to the data space.

Table 4. Addresses of Digital I/O Control Registers

ADDRESS	REGISTER	NAME
7090h	OCRA	I/O mux control register A
7092h	OCRB	I/O mux control register B
7098h	PADATDIR	I/O port A data and direction register
709Ah	PBDATDIR	I/O port B data and direction register
709Ch	PCDATDIR	I/O port C data and direction register
709Eh	PDDATDIR	I/O port D data and direction register

device reset and interrupts

The TMS320x24x software-programmable interrupt structure supports flexible on-chip and external interrupt configurations to meet real-time interrupt-driven application requirements. The C242 recognizes three types of interrupt sources:

- **Reset** (hardware- or software-initiated) is unarbitrated by the CPU and takes immediate priority over any other executing functions. All maskable interrupts are disabled until the reset service routine enables them.

  The C242 device has two sources of reset: an external reset pin and a watchdog timer timeout (reset).

- **Hardware-generated interrupts** are requested by external pins or by on-chip peripherals. There are two types:
  - *External interrupts* are generated by one of four external pins corresponding to the interrupts XINT1, XINT2, $\overline{\text{PDPINT}}$, and $\overline{\text{NMI}}$. The first three can be masked both by dedicated enable bits and by the CPU's interrupt mask register (IMR), which can mask each maskable interrupt line at the DSP core. $\overline{\text{NMI}}$, which is not maskable, takes priority over peripheral interrupts and software-generated interrupts. It can be locked out only by an already executing $\overline{\text{NMI}}$ or a reset.
  - *Peripheral interrupts* are initiated internally by these on-chip peripheral modules: the event manager, SCI, WD, and ADC. They can be masked both by enable bits for each event in each peripheral and by the CPU's IMR, which can mask each maskable interrupt line at the DSP core.

- **Software-generated interrupts** for the C242 include:
  - *The INTR instruction.* This instruction allows initialization of any C242 interrupt with software. Its operand indicates the interrupt vector location to which the CPU branches. This instruction globally disables maskable interrupts (sets the INTM bit to 1).
  - *The NMI instruction.* This instruction forces a branch to interrupt vector location 24h, the same location used for the nonmaskable hardware interrupt NMI. NMI can be initiated by driving the $\overline{\text{NMI}}$ pin low or by executing an NMI instruction. This instruction globally disables maskable interrupts.
  - *The TRAP instruction.* This instruction forces the CPU to branch to interrupt vector location 22h. The TRAP instruction does *not* disable maskable interrupts (INTM is not set to 1); therefore, when the CPU branches to the interrupt service routine, that routine can be interrupted by the maskable hardware interrupts.
  - *An emulator trap.* This interrupt can be generated with either an INTR instruction or a TRAP instruction.

reset

The reset operation ensures an orderly startup sequence for the device. There are two possible causes of a reset, as shown in Figure 4.

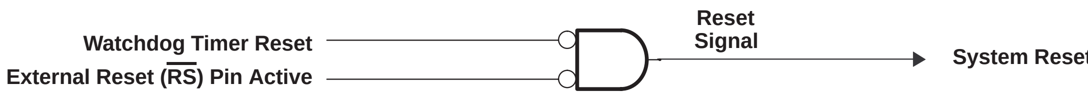


Figure 4. Reset Signals

The two possible reset signals are generated as follows:

- **Watchdog timer reset.** A watchdog-timer-generated reset occurs if the watchdog timer overflows or an improper value is written to either the watchdog key register or the watchdog control register. (Note that when the device is powered on, the watchdog timer is automatically active.) The watchdog timer reset is reflected on the external $\overline{RS}$ pin also.
- **Reset pin active.** To generate an external reset pulse on the $\overline{RS}$ pin, a low-level pulse duration of at least one CPUCLK cycle is necessary to ensure that the device recognizes the reset signal.

Once watchdog reset is activated, the external $\overline{RS}$ pin is driven (active) low for a minimum of eight CPUCLK cycles. This allows the TMS320x24x device to reset external system components.

The occurrence of a reset condition causes the TMS320x24x to terminate program execution and affects various registers and status bits. During a reset, RAM contents remain unchanged, and all control bits that are affected by a reset are initialized to their reset state.

hardware-generated interrupts

The 24x CPU supports one nonmaskable interrupt (NMI) and six maskable prioritized interrupt requests. The 24x devices have many peripherals, and each peripheral is capable of generating one or more interrupts in response to many events. The 24x CPU does not have sufficient interrupt requests to handle all these peripheral interrupt requests; therefore, a centralized interrupt controller is provided to arbitrate the interrupt requests from all the different sources. Throughout this section, refer to Figure 5 .

hardware-generated interrupts (continued)

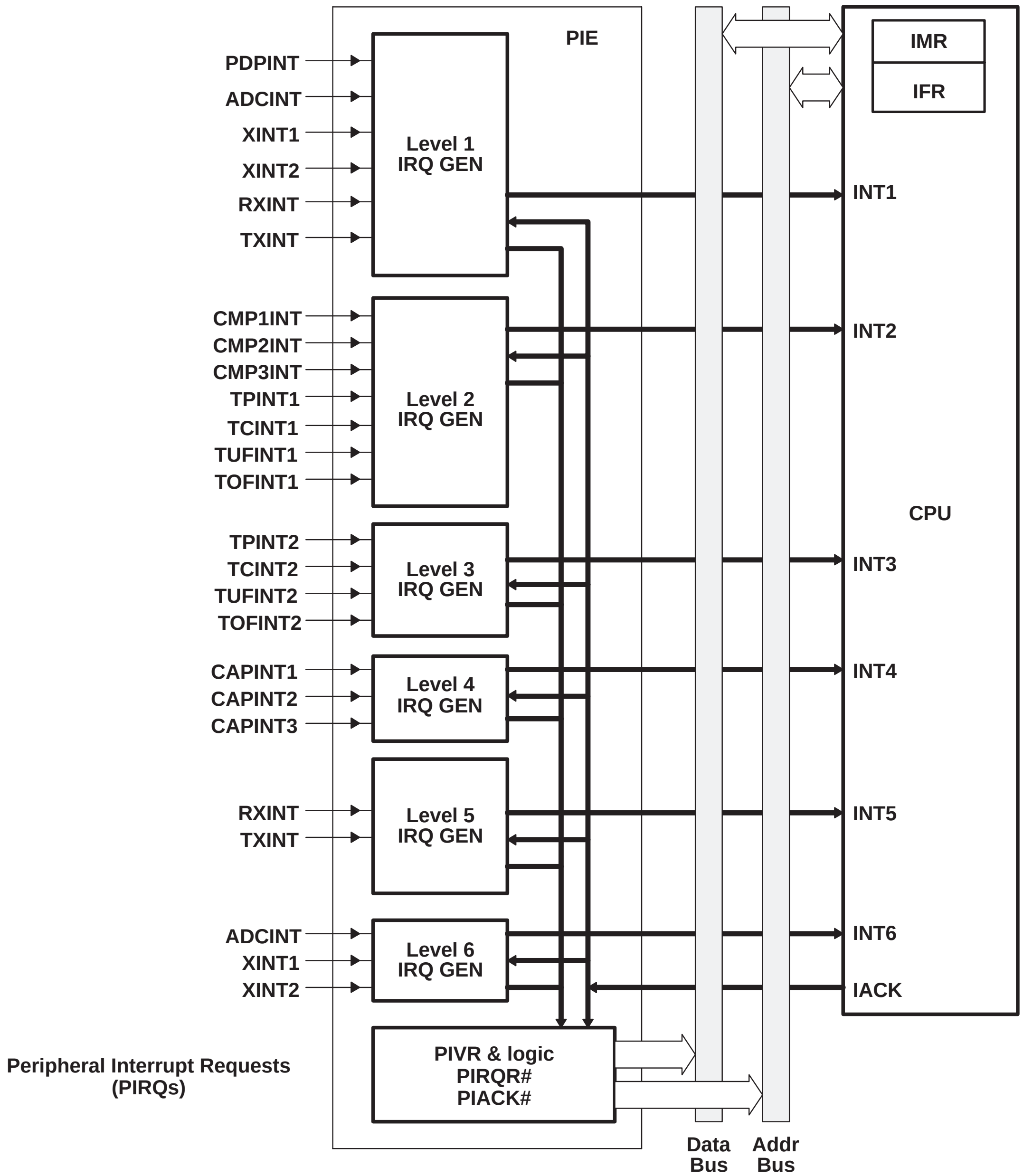


Figure 5. Peripheral Interrupt Expansion Block Diagram

interrupt hierarchy

The number of interrupt requests available is expanded by having two levels of hierarchy in the interrupt request system. There are two levels of hierarchy in both the interrupt request/acknowledge hardware and in the interrupt service routine software.

interrupt request structure

1. At the lower level of the hierarchy, the peripheral interrupt requests (PIRQs) from several peripherals to the interrupt controller are ORed together to generate a request to the CPU. There is an interrupt flag bit and an interrupt enable bit located in the peripheral for each event that can cause a peripheral interrupt request. There is also one PIRQ for each event. If an interrupt-causing event occurs in a peripheral, and the corresponding interrupt enable bit is set, the interrupt request from the peripheral to the interrupt controller is asserted. This interrupt request simply reflects the status of the peripheral's interrupt flag gated with the interrupt enable bit. When the interrupt flag is cleared, the interrupt request is cleared. Some peripherals have the capability to make either a high-priority or a low-priority interrupt request. If a peripheral has this capability, the value of its interrupt priority bit is transmitted to the interrupt controller. The interrupt request continues to be asserted until it is either automatically cleared by an interrupt acknowledge or cleared by software.

2. At the upper level of the hierarchy, the ORed PIRQs generate interrupt (INT) requests to the CPU. The request to the 24x CPU is a low-going pulse of 2 CPU clock cycles. The Peripheral Interrupt Expansion (PIE) controller generates an INT pulse when any of the PIRQs controlling that INT go active. If any of the PIRQs capable of asserting that CPU interrupt request are still active in the cycle following an interrupt acknowledge for that INT, another INT pulse is generated in the PIE. Each INT request is followed by an interrupt acknowledge from the CPU, which helps to clear the interrupt-causing flag in the PIE. The interrupt controller defines which CPU interrupt requests get asserted by which peripheral interrupt requests, and the relative priority of each peripheral interrupt request. Thus, priority is determined by the interrupt controller and is not part of any of the peripherals. Table 5 lists interrupt source priority and vectors.

interrupt request structure (continued)

Table 5. C242 Interrupt Source Priority and Vectors

INTERRUPT NAME	OVERALL PRIORITY	CPU INTERRUPT AND VECTOR ADDRESS	BIT POSITION IN PIRQRx AND PIACKRx†	PERIPHERAL INTERRUPT VECTOR (PIV)	MASK-ABLE?	SOURCE PERIPHERAL MODULE	DESCRIPTION
Reset	1	RSN 0000h		N/A	N	$\overline{RS}$ pin, Watchdog	Reset from pin, watchdog timeout
Reserved	2	− 0026h		N/A	N	CPU	Emulator Trap
NMI	3	NMI 0024h		N/A	N	Nonmaskable Interrupt	Nonmaskable interrupt
PDPINT	4	INT1 0002h	0.0	0020h	Y	EV	Power device protection interrupt pin
ADCINT	5		0.1	0004h	Y	ADC	ADC interrupt in high-priority mode
XINT1	6		0.2	0001h	Y	External Interrupt Logic	External interrupt pins in high priority
XINT2	7		0.3	0011h	Y	External Interrupt Logic	External interrupt pins in high priority
Reserved	8						
RXINT	9		0.4	0006h	Y	SCI	SCI receiver interrupt in high-priority mode
TXINT	10		0.5	0007h	Y	SCI	SCI transmitter interrupt in high-priority mode
Reserved	11						
Reserved	12						
CMP1INT	13	INT2 0004h	0.6	0021h	Y	EV	Compare 1 interrupt
CMP2INT	14		0.7	0022h	Y	EV	Compare 2 interrupt
CMP3INT	15		0.8	0023h	Y	EV	Compare 3 interrupt
TPINT1	16		0.9	0027h	Y	EV	Timer 1 period interrupt
TCINT1	17		0.10	0028h	Y	EV	Timer 1 PWM interrupt
TUFINT1	18		0.11	0029h	Y	EV	Timer 1 underflow interrupt
TOFINT1	19		0.12	002Ah	Y	EV	Timer 1 overflow interrupt
TPINT2	20	INT3 0006h	0.13	002Bh	Y	EV	Timer 2 period interrupt
TCINT2	21		0.14	002Ch	Y	EV	Timer 2 PWM interrupt
TUFINT2	22		0.15	002Dh	Y	EV	Timer 2 underflow interrupt
TOFINT2	23		1.0	002Eh	Y	EV	Timer 2 overflow interrupt
CAPINT1	24	INT4 0008h	1.1	0033h	Y	EV	Capture 1 interrupt
CAPINT2	25		1.2	0034h	Y	EV	Capture 2 interrupt
CAPINT3	26		1.3	0035h	Y	EV	Capture 3 interrupt

† The bit positions of these bits are different compared to F243/F241. These bits should not be used in the user code.
‡ Refer to the *TMS320C24x CPU System and Instruction Set, Volume 1* (SPRU160); and the *TMS320F243,F241,C242 DSP Controllers System and Peripherals User's Guide Volume 2* (SPRU276) for more information.

interrupt request structure (continued)

Table 5. C242 Interrupt Source Priority and Vectors (Continued)

INTERRUPT NAME	OVERALL PRIORITY	CPU INTERRUPT AND VECTOR ADDRESS	BIT POSITION IN PIRQRx AND PIACKRx†	PERIPHERAL INTERRUPT VECTOR (PIV)	MASK-ABLE?	SOURCE PERIPHERAL MODULE	DESCRIPTION
Reserved	27	INT5 000Ah					
RXINT	28		1.4	0006h	Y	SCI	SCI receiver interrupt (low-priority mode)
TXINT	29		1.5	0007h	Y	SCI	SCI transmitter interrupt (low-priority mode)
Reserved	30	INT5 000Ah					
Reserved	31						
ADCINT	32	INT6 000Ch	1.6	0004h	Y	ADC	ADC interrupt (low-priority)
XINT1	33		1.7	0001h	Y	External Interrupt Logic	External interrupt pins (low-priority mode)
XINT2	34		1.8	0011h	Y	External Interrupt Logic	External interrupt pins (low-priority mode)
Reserved		000Eh		N/A	Y	CPU	Analysis interrupt
TRAP	N/A	0022h		N/A	N/A	CPU	TRAP instruction
Phantom Interrupt Vector	N/A	N/A		0000h	N/A	CPU	Phantom interrupt vector
INT8 through INT16	N/A	0010h through 0020h		N/A	N/A	CPU	Software Interrupt Vectors‡
INT20 through INT31	N/A	00028h through 0603Fh		N/A	N/A	CPU	

† The bit positions of these bits are different compared to F243/F241. These bits should not be used in the user code.
‡ Refer to the *TMS320C24x CPU System and Instruction Set, Volume 1* (SPRU160); and the *TMS320F243,F241,C242 DSP Controllers System and Peripherals User's Guide Volume 2* (SPRU276) for more information.

interrupt acknowledge

When the CPU asserts its interrupt acknowledge, it simultaneously puts a value on the memory interface program address bus, which corresponds to the CPU interrupt being acknowledged (it does this because it is fetching the CPU interrupt vector from program memory, each INT has a vector stored in a dedicated program memory address). This value is shown in Table 5, column 3, CPU Interrupt and Vector Address. The PIE controller uses the CPU interrupt acknowledge to generate its internal signals to clear the current interrupt request.

interrupt vectors

When the CPU receives an interrupt request (INT), it does not know which peripheral PIRQ caused the INT request. To enable the CPU to distinguish among the PIRQs, a unique interrupt vector is generated in response to a CPU interrupt acknowledge signal. This vector (PIV) is loaded into the Peripheral Interrupt Vector Register (PIVR) in the PIE controller. The CPU reads this PIV vector value from PIVR and branches to the respective Interrupt Service Routine (SISR). The PIVs are all implemented as hard-coded values on the C242, according to Table 5, column 5.

In effect, there are two vector tables: a CPU vector table and a user-specified peripheral vector table. The CPU's vector table, which starts at 0000h, is used to get to the General Interrupt Service Routine (GISR) in response to a CPU interrupt request (INT). A user-specified peripheral vector table is employed to get to the Event-Specific Interrupt Service Routine (SISR), corresponding to the event which caused the peripheral interrupt request (PIRQ). The code in the GISR should read the Peripheral Interrupt Vector Register (PIVR) after saving any necessary context, and use this value PIV to generate a branch to the SISR. There is one SISR for every interrupt request from a peripheral to the interrupt controller. The SISR performs the actions required in response to the peripheral interrupt request.

phantom interrupt vector

The phantom interrupt vector is an interrupt system integrity feature. If the CPU's interrupt acknowledge is asserted, but there is no associated peripheral interrupt request asserted, the phantom vector is used so that this fault is handled in a controlled manner. One way the phantom interrupt vector could be required is if the CPU executes a software interrupt instruction with an argument corresponding to a peripheral interrupt (usually INT1−INT6). The other way would be if a peripheral made an interrupt request, but its interrupt request flag was cleared by software before the CPU acknowledged the request. In this case, there may be no peripheral interrupt request asserted to the interrupt controller, so the controller would not know which peripheral interrupt vector to load into the PIVR. In these situations, the phantom interrupt vector is loaded into the PIVR in lieu of a peripheral interrupt vector.

nonmaskable interrupts

The PIE controller does not support expansion of nonmaskable interrupts. This is because an ISR must read the peripheral interrupt vector from the PIVR before interrupts are re-enabled. All interrupts (INT1 − INT6) are automatically disabled when the CPU branches to each of the respective vectors. If the PIVR is not read before interrupts are re-enabled (INTM = 0), another interrupt would be acknowledged and a new peripheral interrupt vector would be loaded into the PIVR, causing permanent loss of the original peripheral interrupt vector. Since, by their very nature, nonmaskable interrupts cannot be masked, they cannot be included in the interrupt expansion controller because they could cause the loss of peripheral interrupt vectors.

interrupt operation sequence

1. An interrupt-generating event occurs in a peripheral. The interrupt flag (IF) bit corresponding to that event is set in a register in the peripheral. If the appropriate interrupt enable (IE) bit is set, the peripheral generates an interrupt request to the PIE controller by asserting its PIRQ. If the interrupt is not enabled in the peripheral register, the IF remains set until cleared by software. If the interrupt is enabled at a later time, and the interrupt flag is still set, the PIRQ will immediately be asserted. The interrupt flag (IF) in the peripheral register should be cleared by software only. If the IF bit is not cleared after the respective interrupt service, future interrupts will not be recognized.
2. If no unacknowledged CPU interrupt request of the same priority level has previously been sent, the peripheral interrupt request, PIRQ, causes the PIE controller to generate a CPU interrupt request pulse. This pulse is active low for 2 CPU clock cycles.
3. The interrupt request to the CPU sets the corresponding flag in the CPU's interrupt flag register, IFR. If the CPU interrupt has been enabled (by setting the appropriate bit in the CPU's Interrupt Mask Register, IMR), the CPU stops what it is doing. It then masks all other maskable interrupts by setting the INTM bit, saves some context, clears the respective IFR bit, and starts executing the General Interrupt Service Routine (GISR) for that interrupt priority level. The CPU generates an interrupt acknowledge automatically, which is accompanied by a value on the Program Address Bus (PAB) that corresponds to the interrupt priority level being responded to. These values are shown in Table 5, column 3.
4. The PIE controller decodes the PAB value and generates an internal peripheral interrupt acknowledge to load the PIV into the PIVR. The appropriate peripheral interrupt vector (or the phantom interrupt vector), is referenced from the table stored in the PIE controller.
5. When the GISR has completed any necessary context saves, it reads the PIVR and uses the interrupt vector as a target (or to generate a target) for a branch to the Event-Specific Interrupt Service Routine (SISR) for the interrupt event which occurred in the peripheral. Interrupts *must not* be re-enabled until the PIVR has been read; otherwise, its contents can get overwritten by a subsequent interrupt.

NOTE: If an interrupt occurs during the execution of a CLRC INTM instruction, the device always completes CLRC INTM as well as the next instruction before the pending interrupt is processed. This ensures that a return instruction that directly follows CLRC INTM will be executed before an interrupt is processed. The return instruction will pop the previous return address off the top of the stack before the new return address is pushed onto the stack. To allow the CPU to complete the return, interrupts are also blocked after a RET instruction until at least one instruction at the return address is executed. Interrupts may be blocked for more than one instruction if the instruction at the return address requires additional blocking for pipeline protection. If you want an ISR to occur *within* the current ISR rather than *after* the current ISR, place the CLRC INTM instruction more than one instruction before the return (RET) instruction.

external interrupts

The C242 device has four external interrupts. These interrupts include:

- **XINT1.** The XINT1 control register (at 7070h) provides control and status for this interrupt. XINT1 can be used as a high-priority (Level 1) or low-priority (Level 6) maskable interrupt or as a general-purpose I/O pin. XINT1 can also be programmed to trigger an interrupt on either the rising or the falling edge.
- **XINT2.** The XINT2 control register (at 7071h) provides control and status for this interrupt. XINT2 can be used as a high-priority (Level 1) or low-priority (Level 6) maskable interrupt or a general-purpose I/O pin. XINT2 can also be programmed to trigger an interrupt on either the rising or the falling edge.
- **$\overline{\text{NMI}}$.** This is a nonmaskable external interrupt.
- **$\overline{\text{PDPINT}}$.** This interrupt is provided for safe operation of power converters and motor drives controlled by the C242. This maskable interrupt can put the timers and PWM output pins in high-impedance states and inform the CPU in case of motor drive abnormalities such as overvoltage, overcurrent, and excessive temperature rise. $\overline{\text{PDPINT}}$ is a Level 1 interrupt.

Table 6 is a summary of the external interrupt capability of the C242.

Table 6. External Interrupt Types and Functions

EXTERNAL INTERRUPT	CONTROL REGISTER NAME	CONTROL REGISTER ADDRESS	MASKABLE?
XINT1	XINT1CR	7070h	Yes (Level 1 or 6)
XINT2	XINT2CR	7071h	Yes (Level 1 or 6)
$\overline{\text{NMI}}$	—	—	No
$\overline{\text{PDPINT}}$	EVIMRA	742Ch	Yes (Level 1)

clock generation

The C242 device has an on-chip, (x4) PLL-based clock module. This module provides all the necessary clocking signals for the device, as well as control for low-power mode entry. The only external component necessary for this module is a fundamental crystal. The "times 4" (x4) option for the C242 PLL is fixed and cannot be changed.

The PLL-based clock module provides two modes of operation:

- Crystal-operation
  This mode allows the use of a 5-MHz external reference crystal/resonator to provide the time base to the device.
- External clock source operation
  This mode allows the internal oscillator to be bypassed. The device clocks are generated from an external clock source input on the XTAL1/CLKIN pin. In this case, an external oscillator clock is connected to the XTAL1/CLKIN pin.

The clock module includes two external pins:

1. XTAL1/CLKIN clock source/crystal input
2. XTAL2 output to crystal

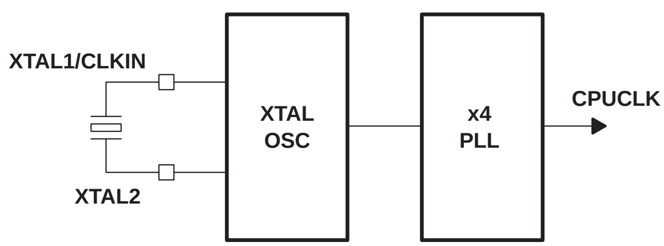


Figure 6. PLL Clock Module Block Diagram

low-power modes

The 24x has an IDLE instruction. When executed, the IDLE instruction stops the clocks to all circuits in the CPU, but the clock output from the CPU continues to run. With this instruction, the CPU clocks can be shut down to save power while the peripherals (clocked with CLKOUT) continue to run. The CPU exits the IDLE state if it is reset, or, if it receives an interrupt request.

clock domains

All 24x-based devices have two clock domains:

1. CPU clock domain − consists of the clock for most of the CPU logic
2. System clock domain − consists of the peripheral clock (which is derived from CLKOUT of the CPU) and the clock for the interrupt logic in the CPU.

When the CPU goes into IDLE mode, the CPU clock domain is stopped while the system clock domain continues to run. This mode is also known as IDLE1 mode. The 24x CPU also contains support for a second IDLE mode, IDLE2. By asserting IDLE2 to the 24x CPU, both the CPU clock domain and the system clock domain are stopped, allowing further power savings. A third low-power mode, HALT mode, the deepest, is possible if the oscillator and WDCLK are also shut down when in IDLE2 mode.

clock domains (continued)

Two control bits, LPM(1) and LPM(0), specify which of the three possible low-power modes is entered when the IDLE instruction is executed (see Table 7). These bits are located in the System Control and Status Register (SCSR) described in the *TMS320F243,F241,C242 DSP Controllers System and Peripherals User's Guide Volume 2* (literature number SPRU276).

Table 7. Low-Power Modes Summary

LOW-POWER MODE	LPMx BITS SCSR[13:12]	CPU CLOCK DOMAIN	SYSTEM CLOCK DOMAIN	WDCLK STATUS	PLL STATUS	OSC STATUS	EXIT CONDITION
CPU running normally	XX	On	On	On	On	On	—
IDLE1 − (LPM0)	00	Off	On	On	On	On	Peripheral Interrupt, External Interrupt, Reset
IDLE2 − (LPM1)	01	Off	Off	On	On	On	Wakeup Interrupts, External Interrupt, Reset
HALT − (LPM2) {PLL/OSC power down}	1X	Off	Off	Off	Off	Off	Reset Only

wakeup from low-power modes

reset

A reset (from any source) causes the device to exit any of the IDLE modes. If the device is halted, the reset will first start the oscillator, and there can be a delay while the oscillator powers up before clocks are generated to initiate the CPU reset sequence.

external interrupts

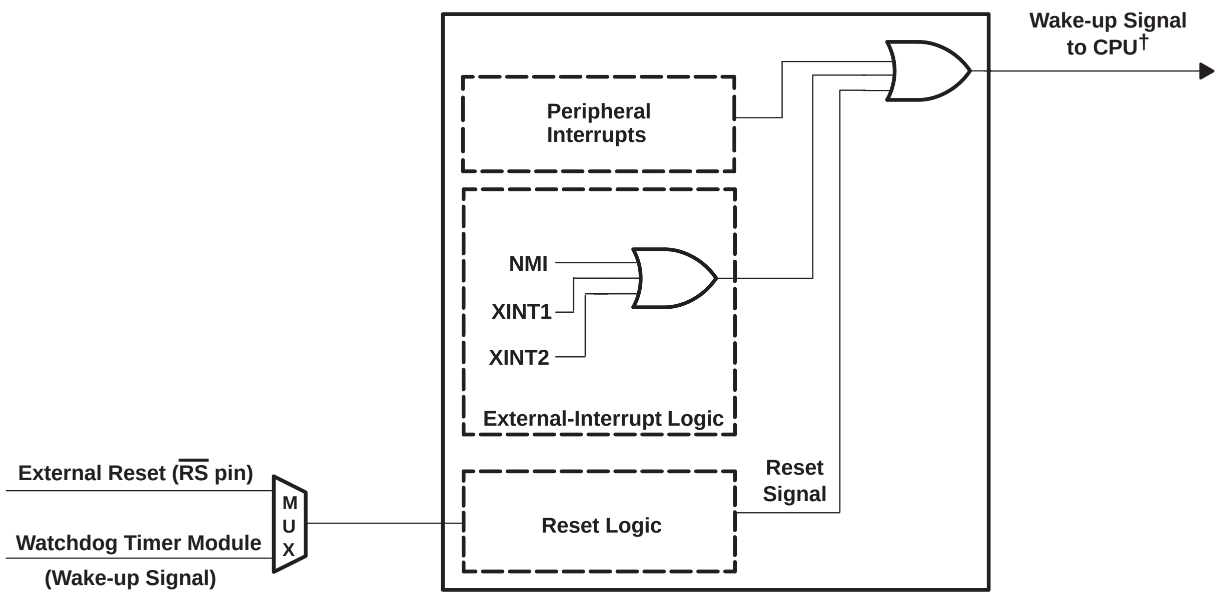
The external interrupts, XINTx, can cause the device to exit any of the low-power modes, except HALT. If the device is in IDLE2 mode, the synchronous logic connected to the external interrupt pins is bypassed with combinatorial logic which recognizes the interrupt on the pin, starts the clocks, and then allows the clocked logic to generate an interrupt request to the PIE controller. Note that in Table 7, external interrupts include $\overline{\text{PDPINT}}$.

wakeup interrupts

Certain peripherals can have the capability to start the device clocks and then generate an interrupt in response to certain external events, for example, activity on a communication line.

peripheral interrupts

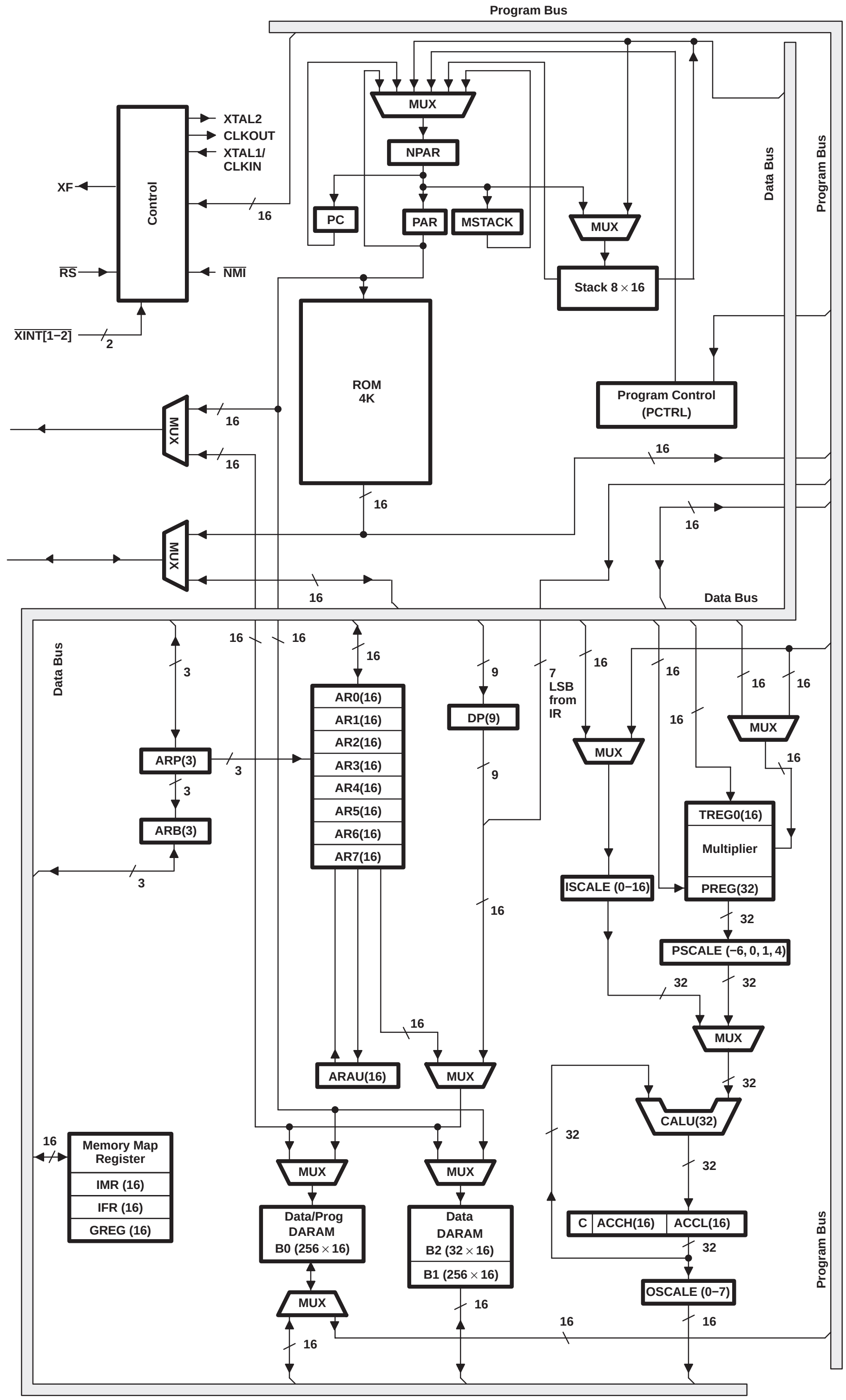
All peripheral interrupts, if enabled locally and globally, can cause the device to exit IDLE1 mode.



† The CPU can exit HALT mode (LPM2) with a RESET only.

Figure 7. Waking Up the Device From Power Down

functional block diagram of the 24x DSP CPU



NOTES: A. Symbol descriptions appear in Table 8 and Table 9.
B. For clarity, the data and program buses are shown as single buses although they include address and data bits.

24x legend for the internal hardware functional block diagram

Table 8. Legend for the 24x Internal Hardware Functional Block Diagram

SYMBOL	NAME	DESCRIPTION
ACC	Accumulator	32-bit register that stores the results and provides input for subsequent CALU operations. Also includes shift and rotate capabilities
ARAU	Auxiliary Register Arithmetic Unit	An unsigned, 16-bit arithmetic unit used to calculate indirect addresses using the auxiliary registers as inputs and outputs
AUX REGS	Auxiliary Registers 0−7	These 16-bit registers are used as pointers to anywhere within the data space address range. They are operated upon by the ARAU and are selected by the auxiliary register pointer (ARP). AR0 can also be used as an index value for AR updates of more than one and as a compare value to AR.
$\overline{BR}$	Bus Request Signal	$\overline{BR}$ is asserted during access of the external global data memory space. READY is asserted to the device when the global data memory is available for the bus transaction. $\overline{BR}$ can be used to extend the data memory address space by up to 32K words.
C	Carry	Register carry output from CALU. C is fed back into the CALU for extended arithmetic operation. The C bit resides in status register 1 (ST1), and can be tested in conditional instructions. C is also used in accumulator shifts and rotates.
CALU	Central Arithmetic Logic Unit	32-bit-wide main arithmetic logic unit for the TMS320C2xx core. The CALU executes 32-bit operations in a single machine cycle. CALU operates on data coming from ISCALE or PSCALE with data from ACC, and provides status results to PCTRL.
DARAM	Dual-Access RAM	If the on-chip RAM configuration control bit (CNF) is set to 0, the reconfigurable data dual-access RAM (DARAM) block B0 is mapped to data space; otherwise, B0 is mapped to program space. Blocks B1 and B2 are mapped to data memory space only, at addresses 0300−03FF and 0060−007F, respectively. Blocks 0 and 1 contain 256 words, while Block 2 contains 32 words.
DP	Data Memory Page Pointer	The 9-bit DP register is concatenated with the seven least significant bits (LSBs) of an instruction word to form a direct memory address of 16 bits. DP can be modified by the LST and LDP instructions.
GREG	Global Memory Allocation Register	GREG specifies the size of the global data memory space. This register is reserved in the C242 as there is no external memory interface on this device.
IMR	Interrupt Mask Register	IMR individually masks or enables the seven interrupts.
IFR	Interrupt Flag Register	The 7-bit IFR indicates that the TMS320C2xx has latched an interrupt from one of the seven maskable interrupts.
INT#	Interrupt Traps	A total of 32 interrupts by way of hardware and/or software are available.
ISCALE	Input Data-Scaling Shifter	16- to 32-bit barrel left-shifter. ISCALE shifts incoming 16-bit data 0 to16 positions left, relative to the 32-bit output within the fetch cycle; therefore, no cycle overhead is required for input scaling operations.
MPY	Multiplier	16×16-bit multiplier to a 32-bit product. MPY executes multiplication in a single cycle. MPY operates either signed or unsigned 2s-complement arithmetic multiply.
MSTACK	Micro Stack	MSTACK provides temporary storage for the address of the next instruction to be fetched when program address-generation logic is used to generate sequential addresses in data space.
MUX	Multiplexer	Multiplexes buses to a common input
NPAR	Next Program Address Register	NPAR holds the program address to be driven out on the PAB on the next cycle.
OSCALE	Output Data-Scaling Shifter	16- to 32-bit barrel left-shifter. OSCALE shifts the 32-bit accumulator output 0 to 7 bits left for quantization management and outputs either the 16-bit high- or low-half of the shifted 32-bit data to the data-write data bus (DWEB).
PAR	Program Address Register	PAR holds the address currently being driven on PAB for as many cycles as it takes to complete all memory operations scheduled for the current bus cycle.
PC	Program Counter	PC increments the value from NPAR to provide sequential addresses for instruction-fetching and sequential data-transfer operations.
PCTRL	Program Controller	PCTRL decodes instruction, manages the pipeline, stores status, and decodes conditional operations.

24x legend for the internal hardware functional block diagram (continued)

Table 8. Legend for the 24x Internal Hardware Functional Block Diagram (Continued)

SYMBOL	NAME	DESCRIPTION
PREG	Product Register	32-bit register holds results of 16 × 16 multiply
PSCALE	Product-Scaling Shifter	0-, 1-, or 4-bit left shift, or 6-bit right shift of multiplier product. The left-shift options are used to manage the additional sign bits resulting from the 2s-complement multiply. The right-shift option is used to scale down the number to manage overflow of product accumulation in the CALU. PSCALE resides in the path from the 32-bit product shifter and from either the CALU or the data-write data bus (DWEB), and requires no cycle overhead.
STACK	Stack	STACK is a block of memory used for storing return addresses for subroutines and interrupt-service routines, or for storing data. The C24x stack is 16-bit wide and eight-level deep.
TREG	Temporary Register	16-bit register holds one of the operands for the multiply operations. TREG holds the dynamic shift count for the LACT, ADDT, and SUBT instructions. TREG holds the dynamic bit position for the BITT instruction.

C242 DSP core CPU

The TMS320x24x devices use an advanced Harvard-type architecture that maximizes processing power by maintaining two separate memory bus structures — program and data — for full-speed execution. This multiple bus structure allows data and instructions to be read simultaneously. Instructions support data transfers between program memory and data memory. This architecture permits coefficients that are stored in program memory to be read in RAM, thereby eliminating the need for a separate coefficient ROM. This, coupled with a four-deep pipeline, allows the C242 to execute most instructions in a single cycle.

Please refer to the TMS320F243/F241 data sheet (SPRS064), specifically the F243/241 DSP core CPU section; the *TMS320C24x CPU System and Instruction Set, Volume 1* (SPRU160); and the *TMS320F243,F241,C242 DSP Controllers System and Peripherals User's Guide Volume 2* (literature number SPRU276) for more information regarding the CPU, input scaling shifter, multiplier, central arithmetic logic unit, accumulator, auxiliary registers, and the auxiliary-register arithmetic unit.

status and control registers

Two status registers, ST0 and ST1, contain the status of various conditions and modes. These registers can be stored into data memory and loaded from data memory, thus allowing the status of the machine to be saved and restored for subroutines.

The load status register (LST) instruction is used to write to ST0 and ST1. The store status register (SST) instruction is used to read from ST0 and ST1 — except for the INTM bit, which is not affected by the LST instruction. The individual bits of these registers can be set or cleared when using the SETC and CLRC instructions. Figure 8 shows the organization of status registers ST0 and ST1, indicating all status bits contained in each. Several bits in the status registers are reserved and are read as logic 1s. Table 9 lists status register field definitions.

	15–13	12	11	10	9	8–0
ST0	ARP	OV	OVM	1	INTM	DP

	15–13	12	11	10	9	8	7	6	5	4	3	2	1–0
ST1	ARB	CNF	TC	SXM	C	1	1	1	1	XF	1	1	PM

Figure 8. Status and Control Register Organization

status and control registers (continued)

Table 9. Status Register Field Definitions

FIELD	FUNCTION
ARB	Auxiliary register pointer buffer. When the ARP is loaded into ST0, the old ARP value is copied to the ARB except during an LST instruction. When the ARB is loaded by way of an LST #1 instruction, the same value is also copied to the ARP.
ARP	Auxiliary register (AR) pointer. ARP selects the AR to be used in indirect addressing. When the ARP is loaded, the old ARP value is copied to the ARB register. ARP can be modified by memory-reference instructions when using indirect addressing, and by the LARP, MAR, and LST instructions. The ARP is also loaded with the same value as ARB when an LST #1 instruction is executed.
C	Carry bit. C is set to 1 if the result of an addition generates a carry, or reset to 0 if the result of a subtraction generates a borrow. Otherwise, C is reset after an addition or set after a subtraction, except if the instruction is ADD or SUB with a 16-bit shift. In these cases, the ADD can only set and the SUB only reset the carry bit, but cannot affect it otherwise. The single-bit shift and rotate instructions also affect C, as well as the SETC, CLRC, and LST #1 instructions. Branch instructions have been provided to branch on the status of C. C is set to 1 on a reset.
CNF	On-chip RAM configuration control bit. If CNF is set to 0, the reconfigurable data dual-access RAM blocks are mapped to data space; otherwise, they are mapped to program space. The CNF can be modified by the SETC CNF, CLRC CNF, and LST #1 instructions. $\overline{\text{RS}}$ sets the CNF to 0.
DP	Data memory page pointer. The 9-bit DP register is concatenated with the seven LSBs of an instruction word to form a direct memory address of 16 bits. DP can be modified by the LST and LDP instructions.
INTM	Interrupt mode bit. When INTM is set to 0, all unmasked interrupts are enabled. When set to 1, all maskable interrupts are disabled. INTM is set and reset by the SETC INTM and CLRC INTM instructions. $\overline{\text{RS}}$ also sets INTM. INTM has no effect on the unmaskable $\overline{\text{RS}}$ and NMI interrupts. Note that INTM is unaffected by the LST instruction. This bit is set to 1 by reset. It is also set to 1 when a maskable interrupt trap is taken.
OV	Overflow flag bit. As a latched overflow signal, OV is set to 1 when overflow occurs in the arithmetic logic unit (ALU). Once an overflow occurs, the OV remains set until a reset, BCND/D on OV/NOV, or LST instructions clear OV.
OVM	Overflow mode bit. When OVM is set to 0, overflowed results overflow normally in the accumulator. When set to 1, the accumulator is set to either its most positive or negative value upon encountering an overflow. The SETC and CLRC instructions set and reset this bit, respectively. LST can also be used to modify the OVM.
PM	Product shift mode. If these two bits are 00, the multiplier’s 32-bit product is loaded into the ALU with no shift. If PM = 01, the PREG output is left-shifted one place and loaded into the ALU, with the LSB zero-filled. If PM = 10, PREG output is left-shifted by four bits and loaded into the ALU, with the LSBs zero-filled. PM = 11 produces a right shift of six bits, sign-extended. Note that the PREG contents remain unchanged. The shift takes place when transferring the contents of the PREG to the ALU. PM is loaded by the SPM and LST #1 instructions. PM is cleared by $\overline{\text{RS}}$.
SXM	Sign-extension mode bit. SXM = 1 produces sign extension on data as it is passed into the accumulator through the scaling shifter. SXM = 0 suppresses sign extension. SXM does not affect the definitions of certain instructions; for example, the ADDS instruction suppresses sign extension regardless of SXM. SXM is set by the SETC SXM and reset by the CLRC SXM instructions, and can be loaded by the LST #1 instruction. SXM is set to 1 by reset.
TC	Test/control flag bit. TC is affected by the BIT, BITT, CMPR, LST #1, and NORM instructions. TC is set to a 1 if a bit tested by BIT or BITT is a 1, if a compare condition tested by CMPR exists between AR (ARP) and AR0, if the exclusive-OR function of the two most significant bits (MSBs) of the accumulator is true when tested by a NORM instruction. The conditional branch, call, and return instructions can execute based on the condition of TC.
XF	XF pin status bit. XF indicates the state of the XF pin, a general-purpose output pin. XF is set by the SETC XF and reset by the CLRC XF instructions. XF is set to 1 by reset.

internal memory

The TMS320C242 device is configured with the following memory modules:

- Dual-access random-access memory (DARAM)
- Mask ROM

dual-access RAM (DARAM)

There are 544 words $\times$ 16 bits of DARAM on the C242 device. The C242 DARAM allows writes to and reads from the RAM in the same cycle. The DARAM is configured in three blocks: block 0 (B0), block 1 (B1), and block 2 (B2). Block 1 contains 256 words and Block 2 contains 32 words, and both blocks are located only in data memory space. Block 0 contains 256 words, and can be configured to reside in either data or program memory space. The SETC CNF (configure B0 as data memory) and CLRC CNF (configure B0 as program memory) instructions allow dynamic configuration of the memory maps through software.

When using on-chip RAM, the C242 runs at full speed with no wait states. The ability of the DARAM to allow two accesses to be performed in one cycle, coupled with the parallel nature of the C242 architecture, enables the device to perform three concurrent memory accesses in any given machine cycle.

ROM

The C242 device contains 4K words of mask-programmable ROM located in program memory space. Customers can arrange to have this ROM programmed with contents unique to any particular application.

illegal access detect

Any access to an illegal address asserts an NMI. This feature is useful to provide a graceful return back to the user code, should an illegal address be accessed inadvertently.

peripherals

The integrated peripherals of the TMS320x24x are described in the following subsections:

- Event-manager (EV2) module
- Analog-to-digital converter (ADC) module
- Serial communications interface (SCI) module
- Watchdog (WD) timer module

event-manager (EV2) module

The event-manager module includes general-purpose (GP) timers, full compare/PWM units, capture units, and quadrature-encoder pulse (QEP) circuits. Figure 9 shows the functions of the event manager.

event-manager (EV2) module (continued)

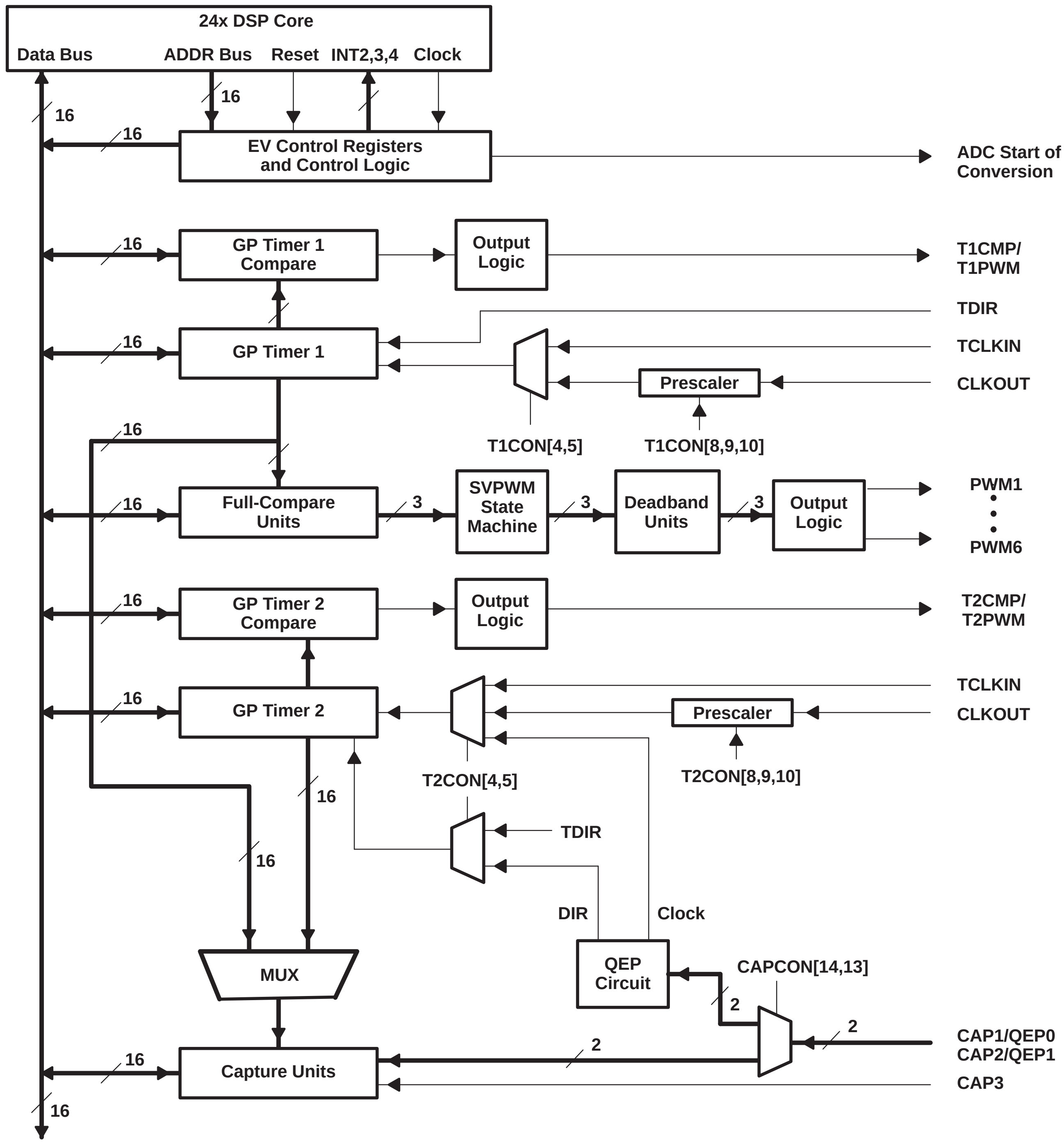


Figure 9. Event-Manager Block Diagram

general-purpose (GP) timers

There are two GP timers on the TMS320x24x. The GP timer x (for x = 1 or 2) includes:

- A 16-bit timer, up-/down-counter, TxCNT, for reads or writes
- A 16-bit timer-compare register, TxCMPR (double-buffered with shadow register), for reads or writes
- A 16-bit timer-period register, TxPR (double-buffered with shadow register), for reads or writes
- A 16-bit timer-control register,TxCON, for reads or writes
- Selectable internal or external input clocks
- A programmable prescaler for internal or external clock inputs
- Control and interrupt logic, for four maskable interrupts: *underflow*, *overflow*, *timer compare*, and *period interrupts*
- A selectable direction input pin (TDIR) (to count up or down when directional up-/down-count mode is selected)

The GP timers can be operated independently or synchronized with each other. The compare register associated with each GP timer can be used for compare function and PWM-waveform generation. There are three continuous modes of operations for each GP timer in up- or up/down-counting operations. Internal or external input clocks with programmable prescaler is used for each GP timer. GP timers also provide the time base for the other event-manager submodules: GP timer 1 for all the compares and PWM circuits, GP timer 2/1 for the capture units and the quadrature-pulse counting operations.

Double-buffering of the period and compare registers allows programmable change of the timer (PWM) period and the compare/PWM pulse width as needed.

full-compare units

There are three full-compare units on TMS320x24x. These compare units use GP timer1 as the time base and generate six outputs for compare and PWM-waveform generation using programmable deadband circuit. The state of each of the six outputs is configured independently. The compare registers of the compare units are double-buffered, allowing programmable change of the compare/PWM pulse widths as needed.

programmable deadband generator

The deadband generator circuit includes three 8-bit counters and an 8-bit compare register. Desired deadband values (from 0 to 24 μs) can be programmed into the compare register for the outputs of the three compare units. The deadband generation can be enabled/disabled for each compare unit output individually. The deadband-generator circuit produces two outputs (with or without deadband zone) for each compare unit output signal. The output states of the deadband generator are configurable and changeable as needed by way of the double-buffered ACTR register.

PWM waveform generation

Up to 8 PWM waveforms (outputs) can be generated simultaneously by TMS320x24x: three independent pairs (six outputs) by the three full-compare units with *programmable deadbands*, and two independent PWMs by the GP-timer compares.

PWM characteristics

Characteristics of the PWMs are as follows:

- 16-bit registers
- Programmable deadband for the PWM output pairs, from 0 to 24 μs
- Minimum deadband width of 50 ns
- Change of the PWM carrier frequency for PWM frequency wobbling as needed
- Change of the PWM pulse widths within and after each PWM period as needed
- External-maskable power and drive-protection interrupts
- Pulse-pattern-generator circuit, for programmable generation of asymmetric, symmetric, and four-space vector PWM waveforms
- Minimized CPU overhead using auto-reload of the compare and period registers

capture unit

The capture unit provides a logging function for different events or transitions. The values of the GP timer 2 counter are captured and stored in the two-level FIFO stacks when selected transitions are detected on capture input pins, CAPx for x = 1, 2, or 3. The capture unit of the TMS320x24x consists of three capture circuits.

- Capture units include the following features:
  - One 16-bit capture control register, CAPCON (R/W)
  - One 16-bit capture FIFO status register, CAPFIFO
  - Selection of GP Timer 2 as the time base
  - Three 16-bit 2-level-deep FIFO stacks, one for each capture unit
  - Three capture input pins CAP1, CAP2, and CAP3, one input pin per each capture unit. [All inputs are synchronized with the device (CPU) clock. In order for a transition to be captured, the input must hold at its current level to meet two rising edges of the device clock. The input pins CAP1 and CAP2 can also be used as QEP inputs to the QEP circuit.]
  - User-specified transition (rising edge, falling edge, or both edges) detection
  - Three maskable interrupt flags, one for each capture unit

quadrature-encoder pulse (QEP) circuit

Two capture inputs (CAP1 and CAP2) can be used to interface the on-chip QEP circuit with a quadrature encoder pulse. Full synchronization of these inputs is performed on-chip. Direction or leading-quadrature pulse sequence is detected, and GP timer 2 is incremented or decremented by the rising and falling edges of the two input signals (four times the frequency of either input pulse).

analog-to-digital converter (ADC) module

A simplified functional block diagram of the ADC module is shown in Figure 10. The ADC module consists of a 10-bit ADC with a built-in sample-and-hold (S/H) circuit. A total of 8 analog input channels is available on the C242. Eight analog inputs are provided by way of an 8-to-1 analog multiplexer. Maximum total conversion time for each ADC unit is 1 µs. Reference voltage for the ADC module is 0−5 V and is supplied externally.

Functions of the ADC module include:

- The ADC unit can perform single or continuous S/H and conversion operations. When in continuous conversion mode, the ADC generates two results every 1700 ns (with a 20-MHz clock and a prescale factor of 1). These two results can be two separate analog inputs.
- Two 2-level-deep FIFO result registers
- Conversion can be started by software, an external signal transition on a device pin (ADCSOC), or by certain event manager events.
- The ADC control register is double-buffered (with a shadow register) and can be written to at any time. A new conversion can start either immediately or when the previous conversion process is completed.
- In single-conversion mode, at the end of each conversion, an interrupt flag is set and the peripheral interrupt request (PIRQ) is generated if it is unmasked/enabled.
- The result of previous conversions stored in data registers will be lost when a third result is stored in the 2-level-deep data FIFO.

A/D overview

The “pseudo” dual ADC is based around a 10-bit string/capacitor converter with the switched capacitor string providing an inherent S/H function. (Note: There is only one converter with only one inherent S/H circuit.) This peripheral behaves as though there are two analog converters, ADC #1 and ADC #2, but in fact, it uses only one converter. This feature makes the A/D software compatible with the C240’s A/D and also allows two values (e.g., voltage and current) to be converted almost simultaneoulsy with one conversion request. V_{CCA} and V_{SSA} pins must be connected to 5 V and analog ground, respectively. Standard isolation techniques must be used while applying power to the ADC module.

The ADC module, shown in Figure 10, has the following features:

- Up to 8 analog inputs, ADCIN00−ADCIN07. The results from converting the inputs ADCIN00−ADCIN07 are placed in one of the ADCFIFO results registers (see Table 10). The digital value of the input analog voltage is derived by:

$$\text{Digital Value} = 1023 \times \frac{\text{Input Analog Voltage} - V_{REFLO}}{V_{REFHI} - V_{REFLO}}$$

- Almost simultaneous measurement of two analog inputs, 1700 ns apart
- Single conversion and continuous conversion modes
- Conversion can be started by software, an internal event, and/or an external event.
- V_{REFHI} and V_{REFLO} (high- and low-voltage) reference inputs
- Two-level-deep digital result registers that contain the digital vaules of completed conversions
- Two programmable ADC module control registers (see Table 10)
- Programmable clock prescaler
- Interrupt or polled operation

A/D overview (continued)

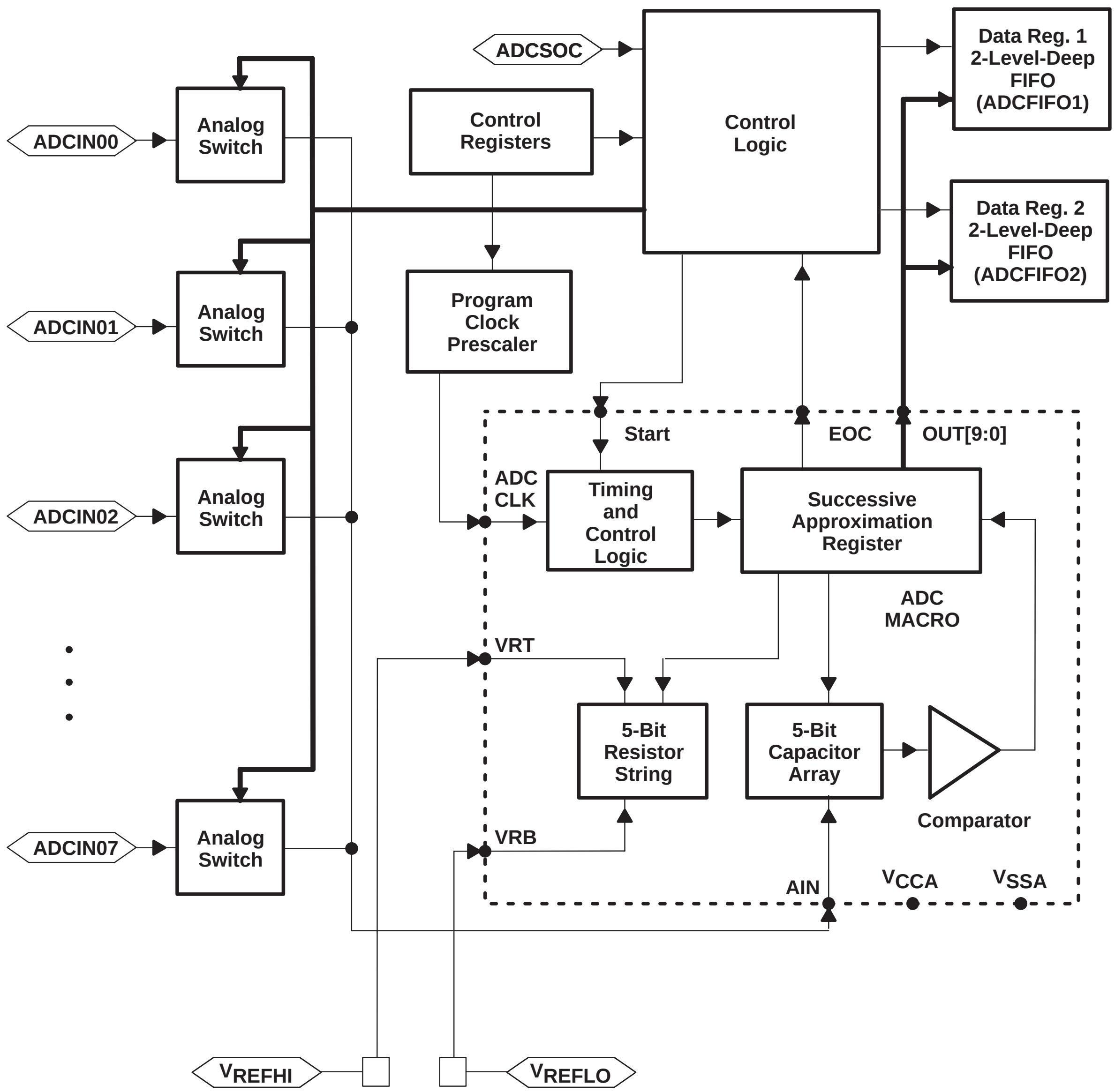


Figure 10. C242 Pseudo Dual Analog-to-Digital Converter (ADC) Module

Table 10. Addresses of ADC Registers

ADDRESS OFFSET	NAME	DESCRIPTION
7032h	ADCTRL1	ADC Control Register 1
7034h	ADCTRL2	ADC Control Register 2
7036h	ADCFIFO1	ADC 2-Level-Deep Data Register FIFO for Pseudo ADC #1
7038h	ADCFIFO2	ADC 2-Level-Deep Data Register FIFO for Pseudo ADC #2

shadowed bits

Many of the control register bits are described as “shadowed”. This means that changing the value of one of these bits does not take effect until the current conversion is complete.

serial communications interface (SCI) module

The C242 device includes a serial communications interface (SCI) module. The SCI module supports digital communications between the CPU and other asynchronous peripherals that use the standard non-return-to-zero (NRZ) format. The SCI receiver and transmitter are double-buffered, and each has its own separate enable and interrupt bits. Both can be operated independently or simultaneously in the full-duplex mode. To ensure data integrity, the SCI checks received data for break detection, parity, overrun, and framing errors. The bit rate is programmable to over 65000 different speeds through a 16-bit baud-select register. Features of the SCI module include:

- Two external pins
  - SCITXD: SCI transmit-output pin
  - SCIRXD: SCI receive-input pin

NOTE: Both pins can be used as GPIO if not used for SCI.

- Baud rate programmable to 64K different rates
  - Up to 1250 Kbps at 20-MHz CPUCLK
- Data word format
  - One start bit
  - Data word length programmable from one to eight bits
  - Optional even/odd/no parity bit
  - One or two stop bits
- Four error-detection flags: parity, overrun, framing, and break detection
- Two wake-up multiprocessor modes: idle-line and address bit
- Half- or full-duplex operation
- Double-buffered receive and transmit functions
- Transmitter and receiver operations can be accomplished through interrupt-driven or polled algorithms with status flags.
  - Transmitter: TXRDY flag (transmitter-buffer register is ready to receive another character) and TX EMPTY flag (transmitter-shift register is empty)
  - Receiver: RXRDY flag (receiver-buffer register is ready to receive another character), BRKDT flag (break condition occurred), and RX ERROR (monitoring four interrupt conditions)
- Separate enable bits for transmitter and receiver interrupts (except BRKDT)
- NRZ (non-return-to-zero) format
- Ten SCI module control registers located in the control register frame beginning at address 7050h

NOTE: All registers in this module are 8-bit registers that are connected to the 16-bit peripheral bus. When a register is accessed, the register data is in the lower byte (7−0), and the upper byte (15−8) is read as zeros. Writing to the upper byte has no effect.

Figure 11 shows the SCI module block diagram.

serial communications interface (SCI) module (continued)

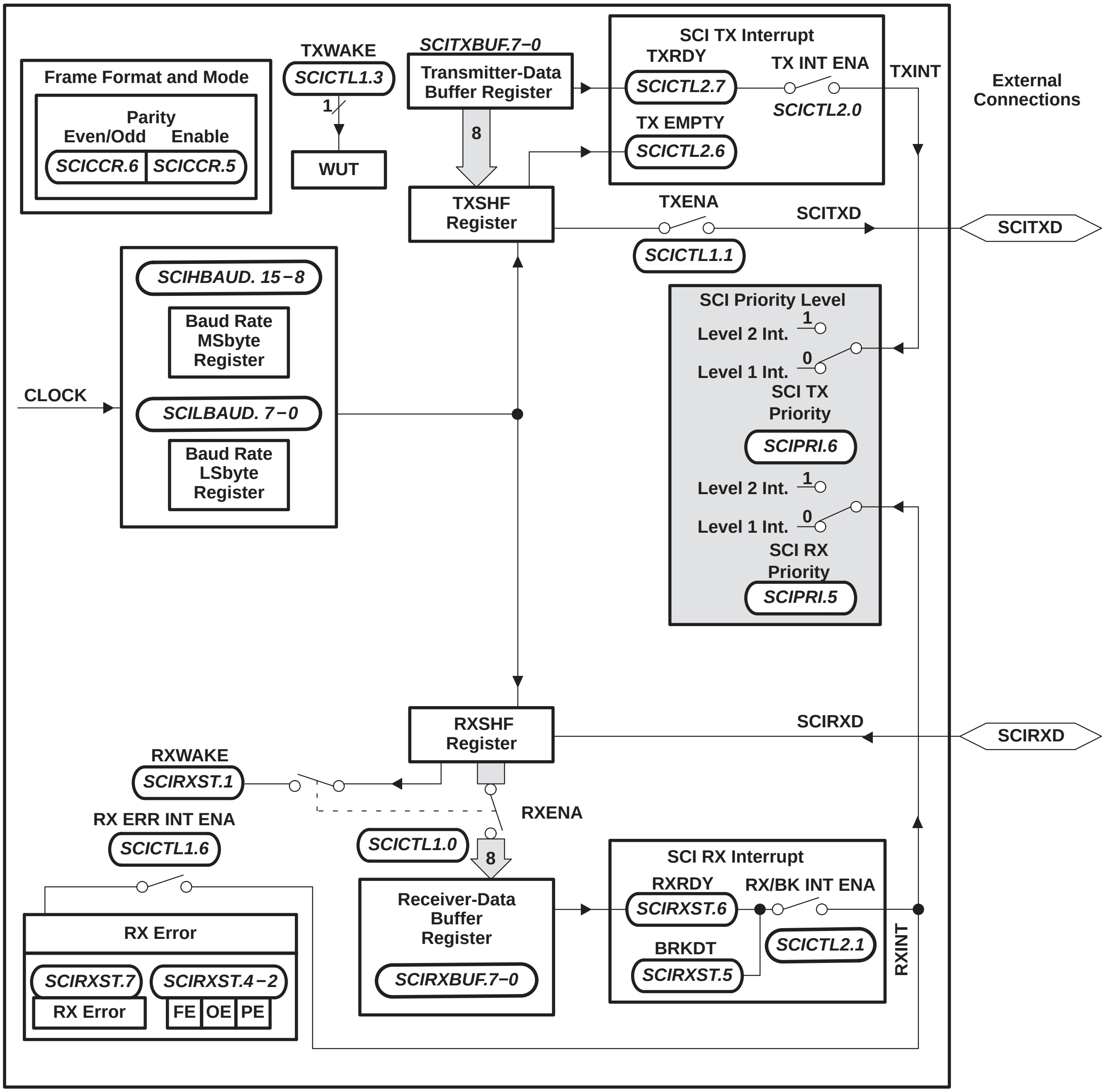


Figure 11. Serial Communications Interface (SCI) Module Block Diagram

watchdog (WD) timer module

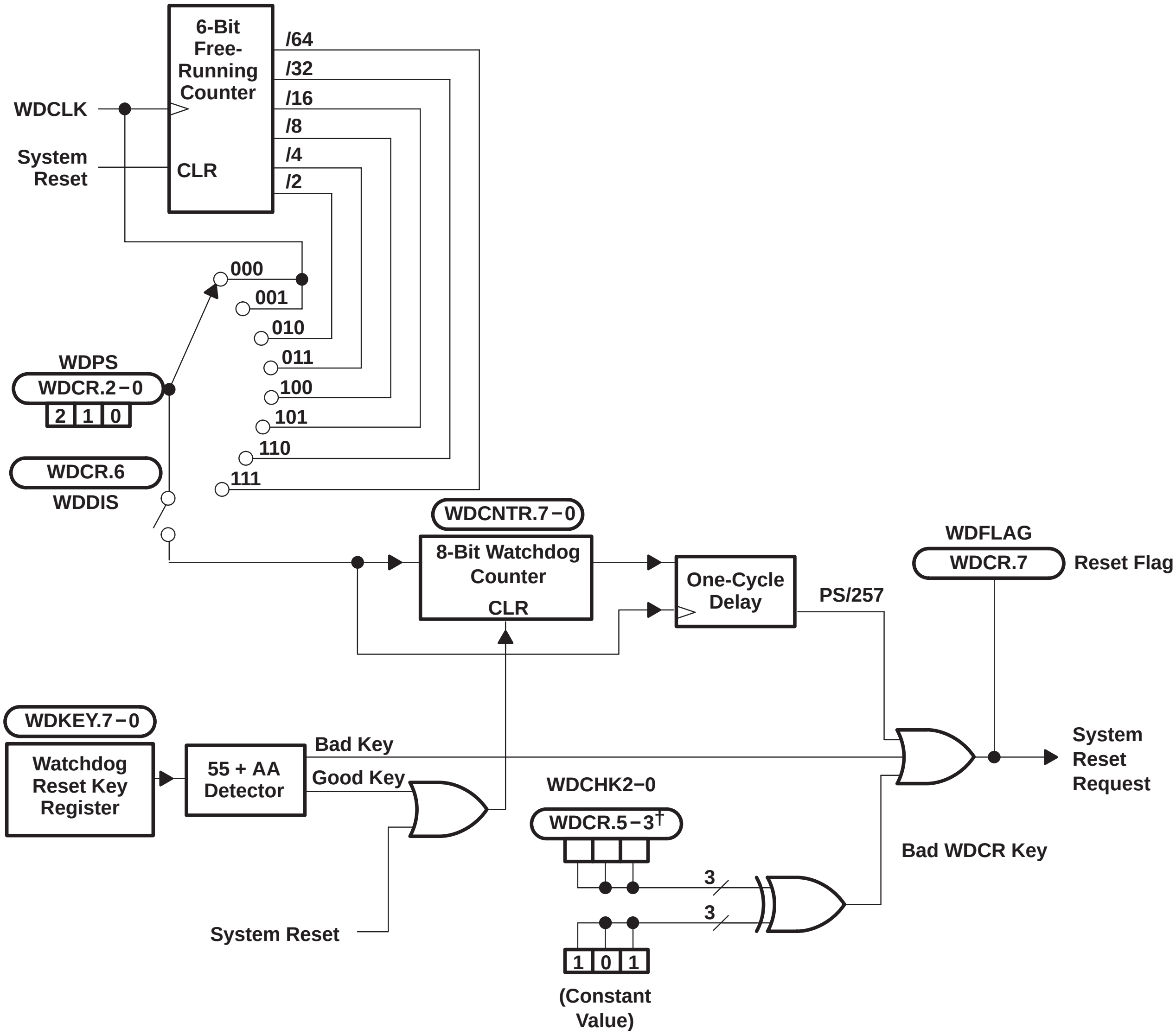
The C242 device includes a watchdog (WD) timer module. The WD function of this module monitors software and hardware operation by generating a system reset if it is not periodically serviced by software by having the correct key written. The WD timer operates independently of the CPU and is always enabled. It does not need any CPU initialization to function. When a system reset occurs, the WD timer defaults to the fastest WD timer rate available (6.55 ms for a 39 062.5-Hz WDCLK signal). As soon as reset is released internally, the CPU starts executing code, and the WD timer begins incrementing. This means that, to avoid a premature reset, WD setup should occur early in the power-up sequence. See Figure 12 for a block diagram of the WD module. The WD module features include the following:

- WD Timer
  - Seven different WD overflow rates ranging from 6.55 ms to 419.43 ms
  - A WD-reset key (WDKEY) register that clears the WD counter when a correct value is written, and generates a system reset if an incorrect value is written to the register
  - WD check bits that initiate a system reset if an incorrect value is written to the WD control register (WDCR)
- Automatic activation of the WD timer, once system reset is released
  - Three WD control registers located in control register frame beginning at address 7020h.

NOTE: All registers in this module are 8-bit registers. When a register is accessed, the register data is in the lower byte, the upper byte is read as zeros. Writing to the upper byte has no effect.

Figure 12 shows the WD block diagram. Table 11 shows the different WD overflow (timeout) selections.

watchdog (WD) timer module (continued)



† Writing to bits WDCR.5−3 with anything but the correct pattern (101) generates a system reset.

Figure 12. Block Diagram of the WD Module

watchdog (WD) timer module (continued)

Table 11. WD Overflow (Timeout) Selections

WD PRESCALE SELECT BITS			WDCLK DIVIDER	39.0625-kHz WDCLK†	
WDPS2	WDPS1	WDPS0		FREQUENCY (Hz)	MINIMUM OVERFLOW (ms)
0	0	X‡	1	152.59	6.55
0	1	0	2	76.29	13.11
0	1	1	4	38.15	26.21
1	0	0	8	19.07	52.43
1	0	1	16	9.54	104.86
1	1	0	32	4.77	209.72
1	1	1	64	2.38	419.43

† Generated by 5-MHz clock
‡ X = Don't care

scan-based emulation

TMS320x2xx devices incorporate scan-based emulation logic for code-development and hardware-development support. Scan-based emulation allows the emulator to control the processor in the system without the use of intrusive cables to the full pinout of the device. The scan-based emulator communicates with the x2xx by way of the IEEE 1149.1-compatible (JTAG) interface. The C242 DSP, like the TMS320F206, TMS320C203, TMS320LC203, and TMS320F243/241, does not include boundary scan. The scan chain of these devices is useful for emulation function only.

development support

Texas Instruments offers an extensive line of development tools for the TMS320C24x generation of DSPs, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules.

The following products support development of x24x-based applications:

Software Development Tools:
Assembler/linker
Simulator
Optimizing ANSI C compiler
Application algorithms
C/Assembly debugger and code profiler

Hardware Development Tools:
Emulator XDS510™ (supports x24x multiprocessor system debug)

The *TMS320 DSP Development Support Reference Guide* (literature number SPRU011) contains information about development support products for all TMS320™ DSP family member devices, including documentation. Refer to this document for further information about TMS320™ DSP documentation or any other TMS320™ DSP support products from Texas Instruments. There is also an additional document, the *TMS320 Third-Party Support Reference Guide* (literature number SPRU052), which contains information from other companies in the industry regarding products related to the TMS320™ DSPs. To receive copies of TMS320™ DSP family literature, contact the Literature Response Center at 800-477-8924.

See Table 12 and Table 13 for complete listings of development support tools for the x24x. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

development support (continued)

Table 12. Development Support Tools

DEVELOPMENT TOOL	PLATFORM	PART NUMBER
Software − Code Generation Tools		
Assembler/Linker	PC™, OS/2™, Windows 3.x/Windows™ 95	TMDS3242850-02
C Compiler/Assembler/Linker	PC, OS/2, Windows 3.x/Windows 95	TMDS3242855-02
C Compiler/Assembler/Linker	Open Windows, HP9000, SPARC™	TMDS3242555-08
Software − Simulation		
C2xx Simulator	SPARC, Open Windows	TMDX324x551-09
Software − Emulation Debug Tools		
Code Composer 4.10, Code Generation 7.0	PC, Windows 3.x, OS/2	TMDS324012xx
TI C Source Debugger − WS	SPARC, SunOS™	TMDX324062xx
Hardware − Emulation Debug Tools		
XDS510XL™ Board (ISA card), w/JTAG cable	PC	TMDS00510
XDS510PP™ Pod (Parallel Port) w/JTAG cable	PC	TMDS00510PP
XDS510WS™ Box w/JTAG cable	SPARC	TMDS00510WS

Table 13. TMS320x24x-Specific Development Tools

DEVELOPMENT TOOL	PLATFORM	PART NUMBER
Hardware − Evaluation/Starter Kits		
TMS320LF2407 EVM	PC, Windows 95, Windows 98	TMDX3P701016
TMS320F240 EVM	PC, Windows 3.x	TMDX326P124X
TMS320F243 EVM	PC, Windows 95	TMDS3P604030

The F240 and F243 Evaluation Modules (EVM) provide designers of motor and motion control applications with a complete and cost-effective way to take their designs from concept to production. These tools offer both a hardware and software development environment and include:

- Flash-based 24x evaluation board
- Code Generation Tools
- Assembler/Linker
- C Compiler (F243 EVM)
- Source code debugger
- C24x Debugger (F240 EVM)
- Code Composer IDE (F243 EVM)
- XDS510PP JTAG-based emulator
- Sample applications code
- Universal 5VDC power supply
- Documentation and cables

device and development support tool nomenclature

To designate the stages in the product development cycle, Texas Instruments assigns prefixes to the part numbers of all TMS320™ DSP devices and support tools. Each TMS320™ DSP family member has one of three prefixes: TMX, TMP, or TMS. Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX / TMDX) through fully qualified production devices/tools (TMS/TMDS). This development flow is defined below.

Device development evolutionary flow:

TMX Experimental device that is not necessarily representative of the final device's electrical specifications

TMP Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification

TMS Fully-qualified production device

Support tool development evolutionary flow:

TMDX Development support product that has not completed TI's internal qualification testing

TMDS Fully qualified development support product

TMX and TMP devices and TMDX development support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

TMS devices and TMDS development support tools have been fully characterized, and the quality and reliability of the device have been fully demonstrated. TI's standard warranty applies.

Predictions show that prototype devices (TMX or TMP) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, FN and PG) and temperature range (for example, L). Figure 13 provides a legend for reading the complete device name for any TMS320x2xx family member.

device and development support tool nomenclature (continued)

TMS 320 C 242 FN (L)

PREFIX
TMX = experimental device
TMP = prototype device
TMS = qualified device

DEVICE FAMILY
320 = TMS320™ DSP Family

TECHNOLOGY
C = CMOS
E = CMOS EPROM
F = Flash EEPROM
LC = Low-voltage CMOS (3.3 V)
VC = Low-voltage CMOS (3 V)

TEMPERATURE RANGE (DEFAULT: 0°C TO 70°C)
L = 0°C to 70°C
A = −40°C to 85°C
S = −40°C to 125°C
Q = −40°C to 125°C, Q 100 Fault Grading

PACKAGE TYPE†
FN = 68-pin PLCC
PG = 64-pin plastic QFP
PGE = 144-pin plastic LQFP

DEVICE
24x DSP
240
241
242
243

† PLCC = Plastic J-Leaded Chip Carrier
QFP = Quad Flatpack
LQFP = Low-Profile Quad Flatpack

Figure 13. TMS320x24x Device Nomenclature

documentation support

Extensive documentation supports all of the TMS320™ DSP family generations of devices from product announcement through applications development. The types of documentation available include: data sheets, such as this document, with design specifications; complete user's guides for all devices and development support tools; and hardware and software applications.

A series of DSP textbooks is published by Prentice-Hall and John Wiley & Sons to support digital signal processing research and education. The TMS320™ DSP newsletter, *Details on Signal Processing*, is published quarterly and distributed to update TMS320 customers on product information.

Updated information on the TMS320™ DSP controllers can be found on the worldwide web at: **http://www.ti.com/dsps**.

To send comments regarding the C242 data sheet (SPRS063), use the *comments@books.sc.ti.com* email address, which is a repository for feedback. For questions and support, contact the Product Information Center listed at the **http://www.ti.com/sc/docs/pic/home.htm** site.

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

Supply voltage range, V_{DD}‡ . . . . −0.3 V to 7 V
Input voltage range . . . . −0.3 V to 7 V
Output voltage range . . . . −0.3 V to 7 V
Input clamp current I_{IK} ($V_I < 0$ or $V_I > V_{DD}$) . . . . ± 20 mA
Output clamp current I_{OK} ($V_O < 0$ or $V_O > V_{DD}$) . . . . ± 20 mA
Operating free-air temperature range, T_A: A version (C242) . . . . −40°C to 85°C
S version (C242) . . . . −40°C to 125°C
Storage temperature range, T_{stg} . . . . −55°C to 150°C

† Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
‡ All voltage values are with respect to V_{SS}.

recommended operating conditions§

			MIN	NOM	MAX	UNIT
V_{DD}	Supply voltage		4.5	5	5.5	V
V_{SS}	Supply ground			0		V
V_{IH}	High-level input voltage	XTAL1/CLKIN	3		V_{DD} + 0.3	V
		All other inputs	2		V_{DD} + 0.3	
V_{IL}	Low-level input voltage	XTAL1/CLKIN	−0.3		0.7	V
		All other inputs	−0.3		0.7	
I_{OH}	High-level output current, V_{OH} = 2.4 V	All outputs			−8	mA
I_{OL}	Low-level output current, V_{OL} = 0.7 V	All outputs			8	mA
T_A	Operating free-air temperature	A version	−40		85	°C
		S version	−40		125	

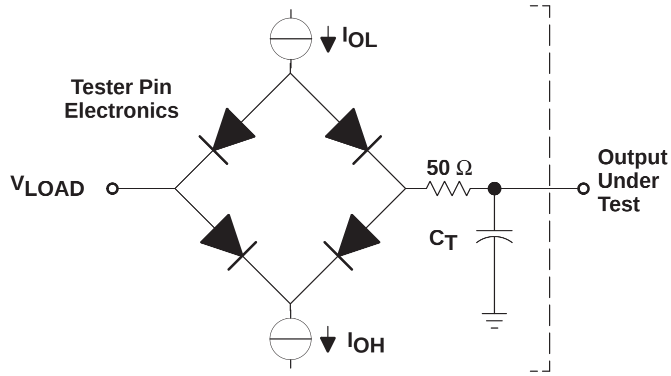
§ Thermal resistance values, Θ_{JA} (junction-to-ambient) and Θ_{JC} (junction-to-case) for the C242 can be found on the mechanical package pages.

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

PARAMETER			TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage		I_{OH} = MAX = 8 mA		2.4			V
V_{OL}	Low-level output voltage		I_{OL} = MAX = 8 mA				0.7	V
I_I	Input current		Pins with pulldown†	V_{IN} = V_{DD}	65	150	350	µA
				V_{IN} = 0 V	−5		5	
			Pins with pullup†	V_{IN} = V_{DD}	−5		5	µA
				V_{IN} = 0 V	−350	−150	−65	
			All other input-only pins		−5		5	µA
I_{OZ}	Output current, high-impedance state (off-state)		V_O = V_{DD} or 0 V		−5	1	5	µA
I_{DD}‡	Supply current, operating mode		$t_{c(CO)}$ = 50 ns			75	100	mA
	Supply current, IDLE1 low-power mode	LPM0	$t_{c(CO)}$ = 50 ns			25	40	mA
	Supply current, IDLE2 low-power mode	LPM1	$t_{c(CO)}$ = 50 ns			20	35	
	Supply current, PLL/OSC power-down mode (HALT)	LPM2				5		mA
C_i	Input capacitance					15		pF
C_o	Output capacitance					15		pF

† See Table 14 for pullup and pulldown information on specific pins.
‡ In operating mode, the CPU is running a dummy code in B0 program memory. In all IDLE modes, the CPU is idle in B0 program memory.

PARAMETER MEASUREMENT INFORMATION



Where: I_{OL} = 2 mA (all outputs)
I_{OH} = 300 µA (all outputs)
V_{LOAD} = 1.5 V
C_T = 110-pF typical load-circuit capacitance

Figure 14. Test Load Circuit

signal transition levels

The data in this section is shown for the 5-V version. Note that some of the signals use different reference voltages, see the recommended operating conditions table. TTL-output levels are driven to a minimum logic-high level of 2.4 V and to a maximum logic-low level of 0.7 V.

Figure 15 shows the TTL-level outputs.

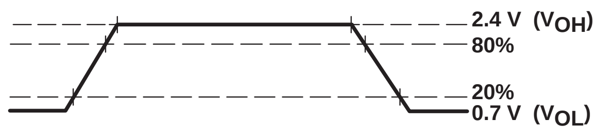


Figure 15. TTL-Level Outputs

TTL-output transition times are specified as follows:

- For a *high-to-low transition*, the level at which the output is said to be no longer high is below 80% of the total voltage range and lower and the level at which the output is said to be low is 20% of the total voltage range and lower.
- For a *low-to-high transition*, the level at which the output is said to be no longer low is 20% of the total voltage range and higher and the level at which the output is said to be high is 80% of the total voltage range and higher.

Figure 16 shows the TTL-level inputs.

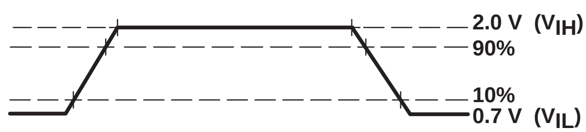


Figure 16. TTL-Level Inputs

TTL-compatible input transition times are specified as follows:

- For a *high-to-low transition* on an input signal, the level at which the input is said to be no longer high is 90% of the total voltage range and lower and the level at which the input is said to be low is 10% of the total voltage range and lower.
- For a *low-to-high transition* on an input signal, the level at which the input is said to be no longer low is 10% of the total voltage range and higher and the level at which the input is said to be high is 90% of the total voltage range and higher.

PARAMETER MEASUREMENT INFORMATION

timing parameter symbology

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

A	A[15:0]	MS	Memory strobe pins $\overline{IS}$, $\overline{DS}$, or $\overline{PS}$
Cl	XTAL1/CLKIN	R	READY
CO	CLKOUT	RD	Read cycle or $\overline{RD}$
D	D[15:0]	RS	RESET pin $\overline{RS}$
INT	$\overline{NMI}$, XINT1, XINT2	W	Write cycle or $\overline{WE}$

Lowercase subscripts and their meanings:

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

Letters and symbols and their meanings:

H	High
L	Low
V	Valid
X	Unknown, changing, or don’t care level
Z	High impedance

general notes on timing parameters

All output signals from the C242 (including CLKOUT) are derived from an internal clock such that all output transitions for a given half-cycle occur with a minimum of skewing relative to each other.

The signal combinations shown in the following timing diagrams may not necessarily represent actual cycles. For actual cycle examples, refer to the appropriate cycle description section of this data sheet.

CLOCK CHARACTERISTICS AND TIMINGS

clock options

PARAMETER
PLL multiply-by-4

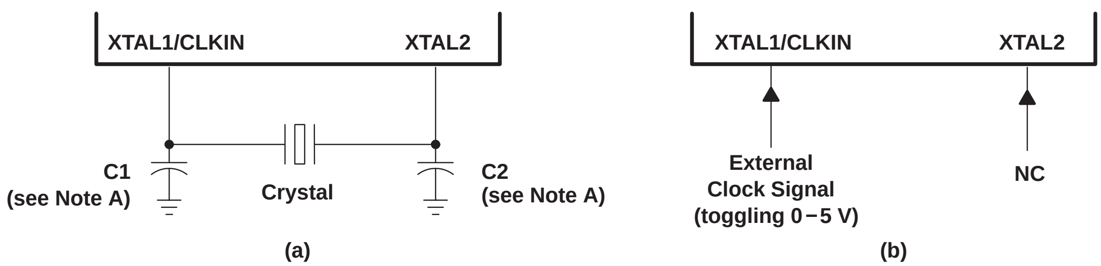
The C242 device includes an on-chip PLL which is hardwired for multiply-by-4 operation. This requires the use of a 5-MHz clock input frequency for 20-MHz device operation. This input clock can be provided from either an external reference crystal or oscillator.

external reference crystal clock option

The internal oscillator is enabled by connecting a crystal across XTAL1/CLKIN and XTAL2 pins as shown in Figure 17a. The crystal should be in fundamental operation and parallel resonant, with an effective series resistance of 30 Ω (typical)† and a power dissipation of 1 mW; it should be specified at a load capacitance of 20 pF.

external reference oscillator clock option

The internal oscillator is disabled by connecting a TTL-level clock signal to XTAL1/CLKIN and leaving the XTAL2 input pin unconnected as shown in Figure 17b.



NOTES: A. For the values of C1 and C2, see the crystal manufacturer's specification.
B. TI recommends that customers have the resonator/crystal vendor characterize the operation of their device with the DSP chip. The resonator/crystal vendor has the equipment and expertise to tune the tank circuit. The vendor can also advise customers regarding the proper component values which ensure startup and stability over the entire operating range.

Figure 17. Recommended Crystal / Clock Connection

† If the input frequency is 5 MHz, the series resistances of the crystal can be between 30 Ω to 150 Ω.

external reference crystal/clock with PLL circuit enabled

timings with the PLL circuit enabled

PARAMETER			MIN	TYP	MAX	UNIT
f_x	Input clock frequency	Oscillator/Resonator	1		5	MHz
		CLKIN	1		5	MHz
C1, C2	Load capacitance			10		pF

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$] (see Figure 18)

PARAMETER		CLOCK MODE	MIN	TYP	MAX	UNIT
$t_{c(CO)}$	Cycle time, CLKOUT		50			ns
$t_{f(CO)}$	Fall time, CLKOUT			4		ns
$t_{r(CO)}$	Rise time, CLKOUT			4		ns
$t_{w(COL)}$	Pulse duration, CLKOUT low		H−3	H	H+3	ns
$t_{w(COH)}$	Pulse duration, CLKOUT high		H −3	H	H+3	ns
t_p	Transition time, PLL synchronized after PLL enabled	before PLL lock, CLKIN multiply by 4			2500$t_{c(CI)}$	ns

timing requirements (see Figure 18)

		EXTERNAL REFERENCE CRYSTAL	MIN	MAX	UNIT
$t_{c(CI)}$	Cycle time†, XTAL1/CLKIN	5 MHz	200		ns
$t_{f(CI)}$	Fall time, XTAL1/CLKIN			5	ns
$t_{r(CI)}$	Rise time, XTAL1/CLKIN			5	ns
$t_{w(CIL)}$	Pulse duration, XTAL1/CLKIN low as a percentage of $t_{c(CI)}$		40	60	%
$t_{w(CIH)}$	Pulse duration, XTAL1/CLKIN high as a percentage of $t_{c(CI)}$		40	60	%

† This device utilizes a fully static design and, therefore, can operate with input clock cycle time [$t_{c(CI)}$] approaching infinity. The device is characterized at frequencies approaching 0 Hz, but is tested at f_{clk} = 6.7 MHz to meet device test time requirements.

NOTE: Timings assume CLKOUT is set to output CPUCLK. CLKOUT is initialized to CPUCLK by power-on reset.Timings assume CLKOUT is set to output CPUCLK. CLKOUT is initialized to CPUCLK by power-on reset.

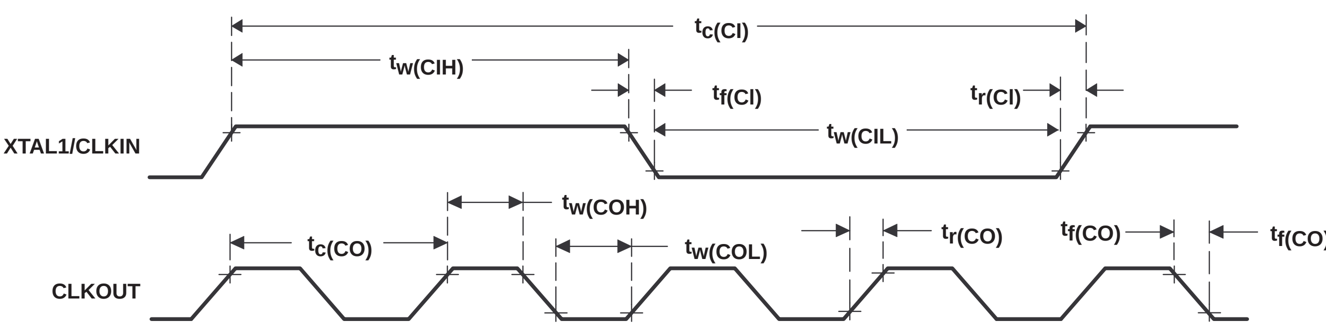


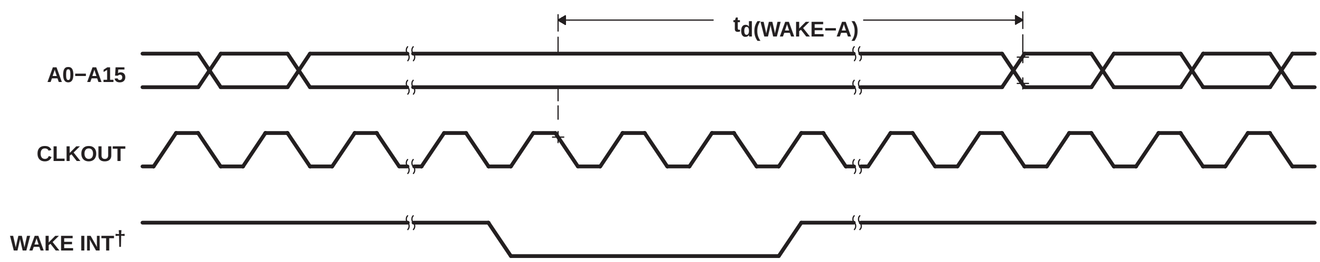
Figure 18. CLKIN-to-CLKOUT Timing for PLL Oscillator Mode, Multiply-by-4 Option with 5-MHz Clock

low-power mode timings

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$]† (see Figure 19 and Figure 20)

PARAMETER		LOW-POWER MODES		MIN	TYP	MAX	UNIT
$t_{d(WAKE\text{-}A)}$	Delay time, CLKOUT switching to program execution resume	IDLE1/IDLE2	LPM0 LPM1		4 + 6 $t_{c(CO)}$	15 × $t_{c(CO)}$	ns
$t_{d(IDLE\text{-}COH)}$	Delay time, Idle instruction executed to CLKOUT high	HALT {PLL/OSC power down}	LPM2		4$t_{c(CO)}$		ns
$t_{d(WAKE\text{-}OSC)}$	Delay time, wakeup interrupt asserted to oscillator running				OSC start-up and PLL lock time		ms
$t_{d(IDLE\text{-}OSC)}$	Delay time, Idle instruction executed to oscillator power off				4$t_{c(CO)}$		µs
$t_{d(EX)}$	Delay time, reset vector executed after $\overline{RS}$ high			36H			ns

† Timings assume CLKOUT is set to output CPUCLK. CLKOUT is initialized to CPUCLK by power-on reset.



† WAKE INT can be any valid interrupt or RESET.

Figure 19. Entry and Exit Timing − LPM0 and LPM1

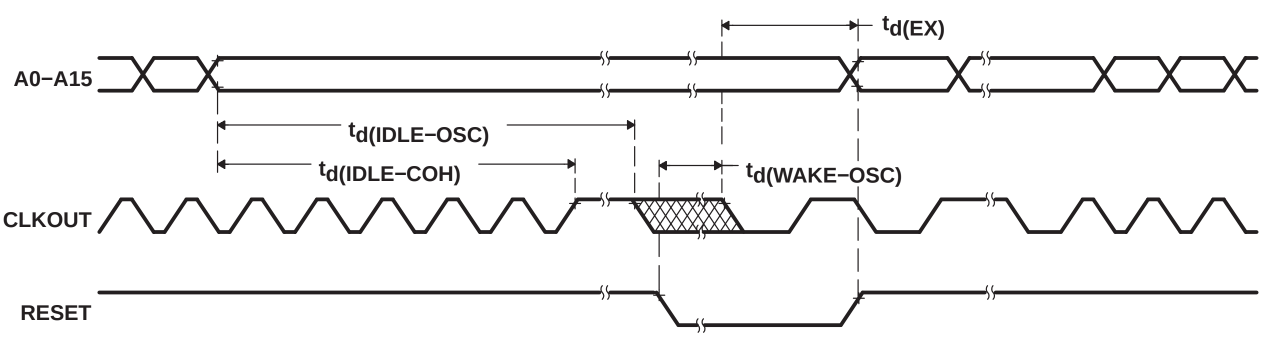


Figure 20. HALT Mode − LPM2

$\overline{RS}$ timings

switching characteristics over recommended operating conditions for a reset [H = 0.5$t_{c(CO)}$] (see Figure 21)

PARAMETER		MIN	MAX	UNIT
$t_{w(RSL1)}$	Pulse duration, $\overline{RS}$ low†	8$t_{c(CO)}$		ns
$t_{d(EX)}$	Delay time, reset vector executed after $\overline{RS}$ high	36H		ns

† The parameter $t_{w(RSL1)}$ refers to the time $\overline{RS}$ is an output.

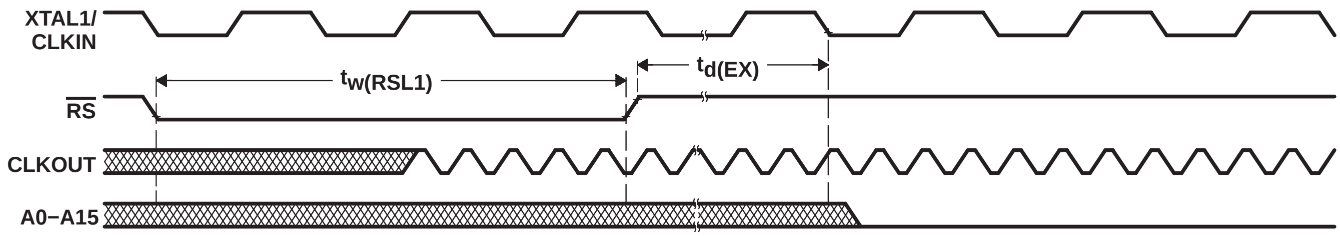
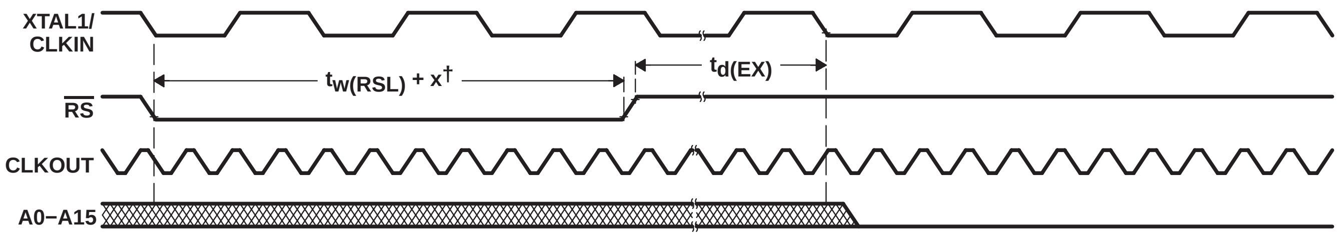


Figure 21. Watchdog Reset Pulse

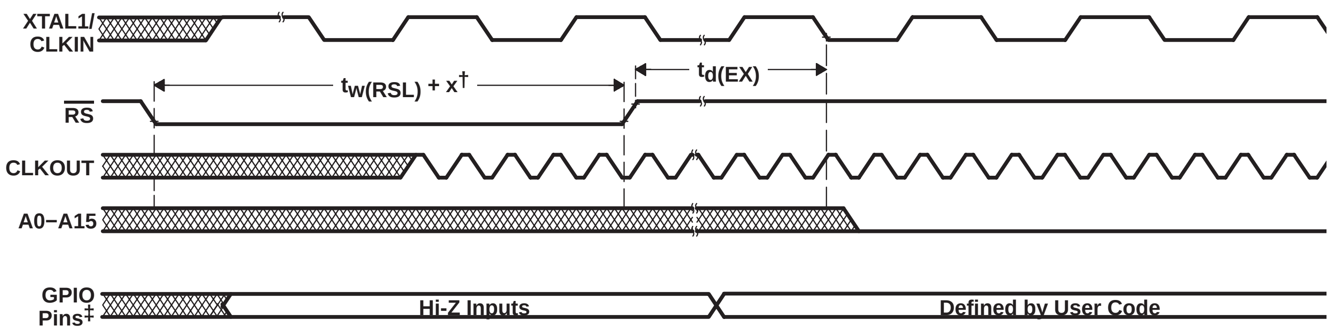
timing requirements for a reset [H = 0.5$t_{c(CO)}$] (see Figure 22)

		MIN	MAX	UNIT
$t_{w(RSL)}$	Pulse duration, $\overline{RS}$ low†	8H		ns
$t_{d(EX)}$	Delay time, reset vector executed after $\overline{RS}$ high	36H		ns

† The parameter $t_{w(RSL)}$ refers to the time $\overline{RS}$ is an input.



Case A. External reset after power-on



Case B. Power-on reset

† The value of x depends on the reset condition as follows: **PLL enabled:** Assuming CLKIN is stable, x=PLL lock-up time. If the internal oscillator is used, x=oscillator lock-up time + PLL lock-up time. In case of resets after power on reset, x=0 (i.e., $t_{w(RSL)}$=8H ns only).

‡ (All GPIO pins *except* CLKOUT and XF pins.) GPIO pins are undefined until XTAL1/CLKIN is valid. This behavior is important to consider while using PWM pins for power-electronic circuits.

Figure 22. Reset Timing

XF and $\overline{\text{BIO}}$ timings

switching characteristics over recommended operating conditions (see Figure 23)

PARAMETER		MIN	MAX	UNIT
$t_{d(XF)}$	Delay time, CLKOUT high to XF high/low	−3	7	ns

timing requirements (see Figure 23)

		MIN	MAX	UNIT
$t_{su(BIO)CO}$	Setup time, $\overline{\text{BIO}}$ low before CLKOUT low	0		ns
$t_{h(BIO)CO}$	Hold time, $\overline{\text{BIO}}$ low after CLKOUT low	19		ns

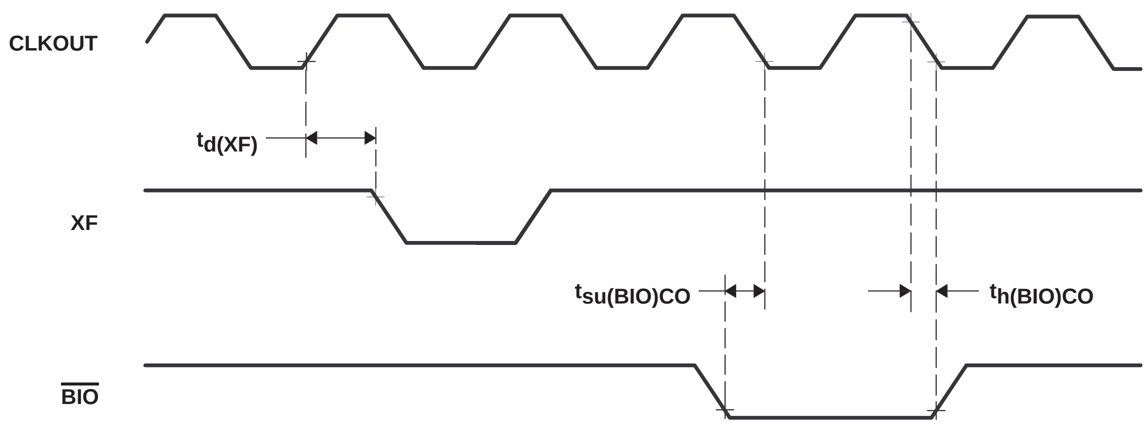


Figure 23. XF and $\overline{\text{BIO}}$ Timing

TIMING EVENT MANAGER INTERFACE

PWM timings

PWM refers to PWM outputs on PWM1, PWM2, PWM3, PWM4, PWM5, PWM6, T1PWM, and T2PWM.

switching characteristics over recommended operating conditions for PWM timing [H = 0.5$t_{c(CO)}$] (see Figure 24)

PARAMETER		MIN	MAX	UNIT
$t_{w(PWM)}$†	Pulse duration, PWM output high/low	2H+5		ns
$t_{d(PWM)CO}$	Delay time, CLKOUT low to PWM output switching		15	ns

† PWM outputs may be 100%, 0%, or increments of $t_{c(CO)}$ with respect to the PWM period.

timing requirements‡ [H = 0.5$t_{c(CO)}$] (see Figure 25)

		MIN	MAX	UNIT
$t_{w(TMRDIR)}$	Pulse duration, TMRDIR low/high	4H+5		ns
$t_{w(TMRCLK)}$	Pulse duration, TMRCLK low as a percentage of TMRCLK cycle time	40	60	%
$t_{wh(TMRCLK)}$	Pulse duration, TMRCLK high as a percentage of TMRCLK cycle time	40	60	%
$t_{c(TMRCLK)}$	Cycle time, TMRCLK	$4 \times t_{c(CO)}$		ns

‡ Parameter TMRDIR is equal to the pin TDIR, and parameter TMRCLK is equal to the pin TCLKIN.

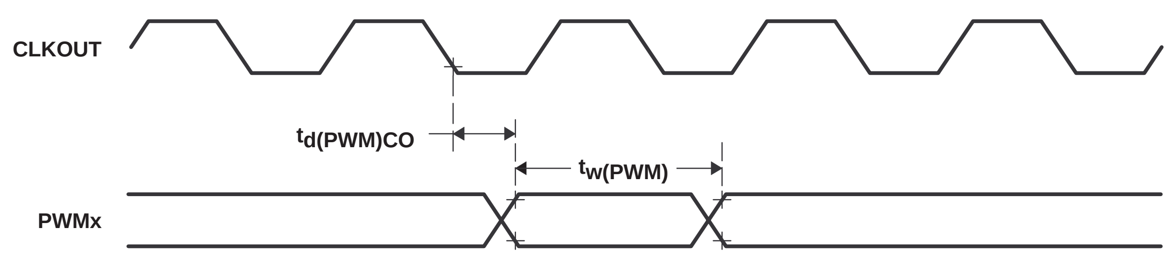


Figure 24. PWM Output Timing

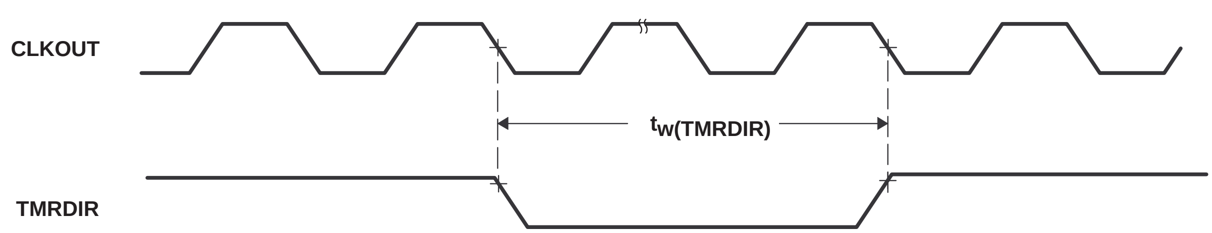


Figure 25. Capture/TMRDIR Timing

capture and QEP timings

CAP refers to CAP1/QEP0/IOPA3, CAP2/QEP1/IOPA4, and CAP3/IOPA5.

timing requirements [H = 0.5$t_{c(CO)}$] (see Figure 26)

		MIN	MAX	UNIT
$t_{w(CAP)}$	Pulse duration, CAP input low/high	4H +15		ns

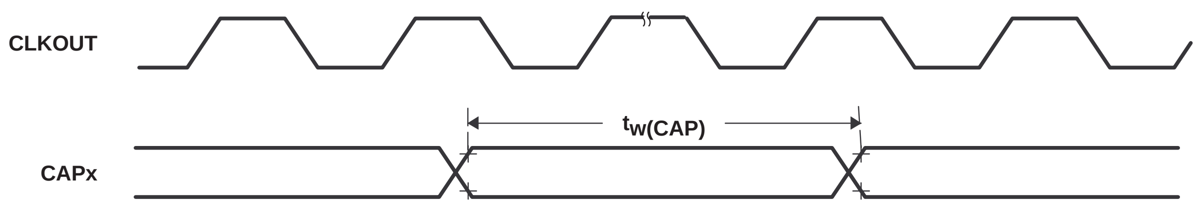


Figure 26. Capture Input and QEP Timing

interrupt timings

INT refers to $\overline{\text{NMI}}$, XINT1, and XINT2/IO. PDP refers to $\overline{\text{PDPINT}}$.

switching characteristics over recommended operating conditions (see Figure 27)

PARAMETER		MIN	MAX	UNIT
$t_{hz(PWM)PDP}$	Delay time, $\overline{\text{PDPINT}}$ low to PWM to high-impedance state		12	ns
$t_{d(INT)}$	Delay time, INT low/high to interrupt-vector fetch	$10t_{c(CO)}$		ns

timing requirements [H = $0.5t_{c(CO)}$] (see Figure 27)

		MIN	MAX	UNIT
$t_{w(INT)}$	Pulse duration, INT input low/high	2H+15		ns
$t_{w(PDP)}$	Pulse duration, $\overline{\text{PDPINT}}$ input low	4H+5		ns

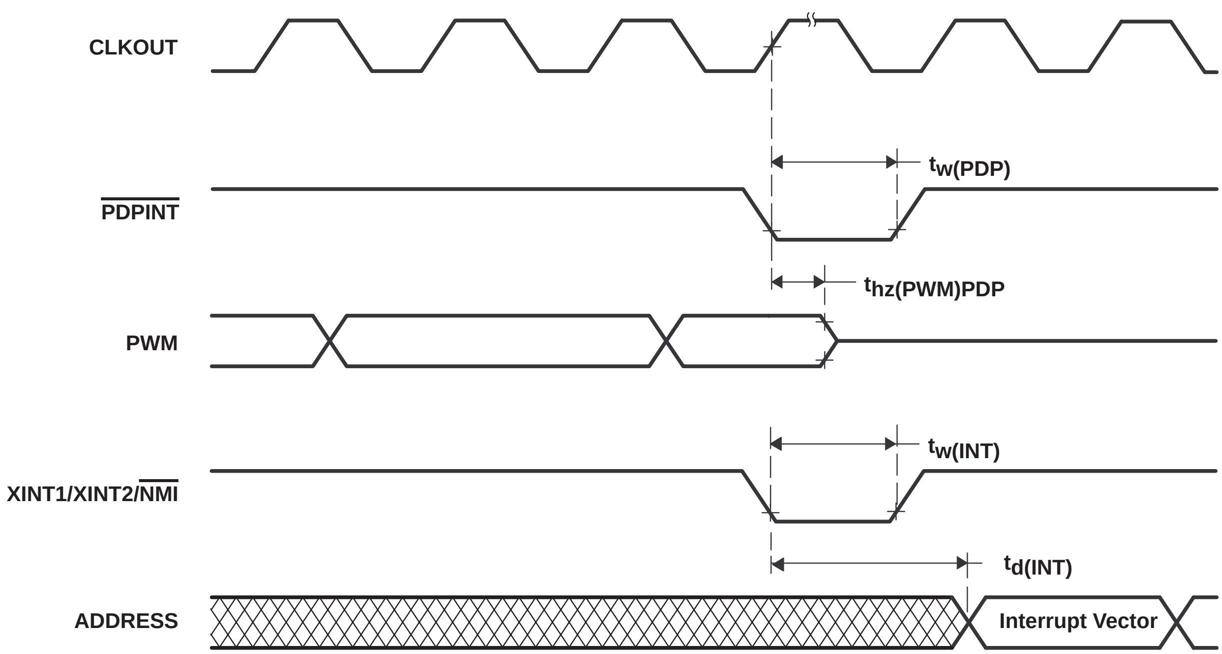


Figure 27. $\overline{\text{PDPINT}}$ and External Interrupt Timings

general-purpose input/output timings

switching characteristics over recommended operating conditions (see Figure 28)

PARAMETER			MIN	MAX	UNIT
$t_{d(GPO)CO}$	Delay time, CLKOUT low to GPIO low/high	All GPIOs		9	ns
$t_{r(GPO)}$	Rise time, GPIO switching low to high	All GPIOs		8	ns
$t_{f(GPO)}$	Fall time, GPIO switching high to low	All GPIOs		6	ns

timing requirements [H = 0.5$t_{c(CO)}$] (see Figure 29)

		MIN	MAX	UNIT
$t_{w(GPI)}$	Pulse duration, GPI high/low	2H+15		ns

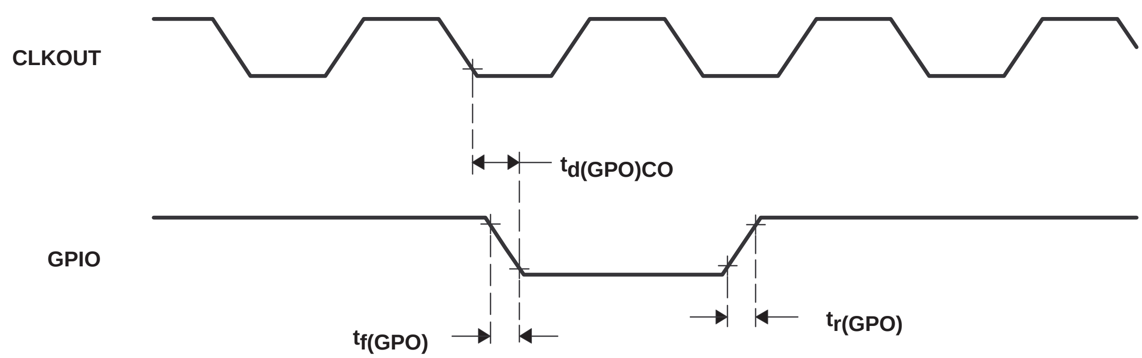


Figure 28. General-Purpose Output Timing

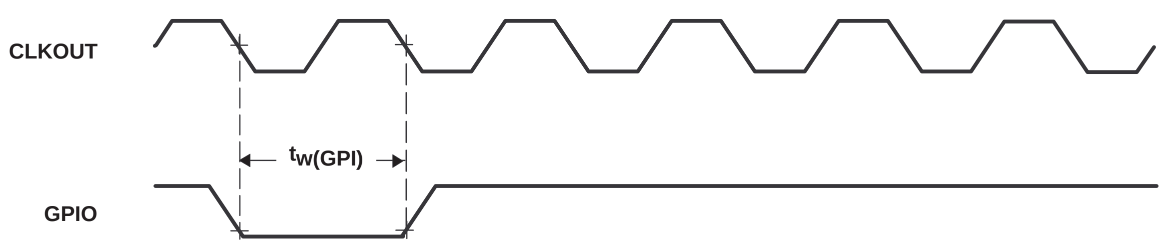


Figure 29. General-Purpose Input Timing

10-bit dual analog-to-digital converter (ADC)

The 10-bit ADC has a separate power bus for its analog circuitry. These pins are referred to as V_{CCA} and V_{SSA}. The power bus isolation is to enhance ADC performance by preventing digital switching noise of the logic circuitry that can be present on V_{SS} and V_{CC} from coupling into the ADC analog stage. All ADC specifications are given with respect to V_{SSA} unless otherwise noted.

Resolution . . . . 10-bit (1024 values)
Monotonic . . . . Assured
Output conversion mode . . . . 000h to 3FFh (000h for $V_I \le V_{SSA}$; 3FFh for $V_I \ge V_{CCA}$)
Conversion time (including sample time) . . . . 1 µs

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CCA}	Analog supply voltage	4.5	5	5.5	V
V_{SSA}	Analog ground		0		V
V_{REFHI}	Analog supply reference source†	V_{REFLO}		V_{CCA}	V
V_{REFLO}	Analog ground reference source†	V_{SSA}		V_{REFHI}	V
V_{AI}	Analog input voltage, ADCIN00−ADCIN07	V_{SSA}		V_{CCA}	V

† V_{REFHI} and V_{REFLO} must be stable, within ±1/2 LSB of the required resolution, during the entire conversion time.

ADC operating frequency

	MIN	MAX	UNIT
ADC operating frequency		20	MHz

operating characteristics over recommended operating condition ranges†

PARAMETER		DESCRIPTION		MIN	MAX	UNIT
I_{CCA}	Analog supply current	V_{CCA} = 5.5 V	Converting		10	mA
			Non-converting		2	mA
		V_{CCA} = V_{REFHI} = 5.5 V	PLL or OSC power down		1	µA
C_{ai}	Analog input capacitance	Typical capacitive load on analog input pin	Non-sampling		10	pF
			Sampling		30	pF
E_{DNL}	Differential nonlinearity error	Difference between the actual step width and the ideal value			±2	LSB‡
E_{INL}	Integral nonlinearity error	Maximum deviation from the best straight line through the ADC transfer characteristics, excluding the quantization error			±2	LSB‡
$t_{d(PU)}$	Delay time, power-up to ADC valid	Time to stabilize analog stage after power-up			10	µs
Z_{AI}	Analog input source impedance	Analog input source impedance for conversions to remain within specifications			10	Ω

† Absolute resolution = 4.89 mV. At V_{REFHI} = 5 V and V_{REFLO} = 0 V, this is one LSB. As V_{REFHI} decreases, V_{REFLO} increases, or both, the LSB size decreases. Therefore, the absolute accuracy and differential/integral linearity errors in terms of LSBs increase.
‡ LSB denotes “Least Significant Bits”. For example, an error of 2 LSB corresponds to (2 * 4.89) = 9.78 mV.

ADC input pin circuit

One of the most common A/D application errors is inappropriate source impedance. In practice, minimum source impedance should be used to limit the error as well as to minimize the required sampling time; however, the source impedance must be smaller than Z_{AI}. A typical ADC input pin circuit is shown in Figure 30.

Figure 30. Typical ADC Input Pin Circuit

internal ADC module timings (see Figure 31)

		MIN	MAX	UNIT
$t_{c(AD)}$	Cycle time, ADC prescaled clock	50		ns
$t_{w(SHC)}$	Pulse duration, total sample/hold and conversion time†	900		ns
$t_{w(SH)}$	Pulse duration, sample and hold time	$3t_{c(AD)}$		ns
$t_{w(C)}$	Pulse duration, total conversion time	$10t_{c(AD)}$		ns
$t_{d(SOC\text{-}SH)}$	Delay time, start of conversion‡ to beginning of sample and hold	$3t_{c(CO)}$		ns
$t_{d(EOC\text{-}FIFO)}$	Delay time, end of conversion to data loaded into result FIFO	$2t_{c(CO)}$		ns
$t_{d(ADCINT)}$	Delay time, ADC flag to ADC interrupt	$2t_{c(CO)}$		ns

† The total sample/hold and conversion time is determined by the summation of $t_{d(SOC\text{-}SH)}$, $t_{w(SH)}$, $t_{w(C)}$, and $t_{d(EOC\text{-}FIFO)}$.
‡ Start of conversion is signaled by the ADCIMSTART bit (ADCTRL1.13) or the ADCSOC bit (ADCTRL1.0) set in software, the external start signal active (ADCSOC), or internal EVSOC signal active.

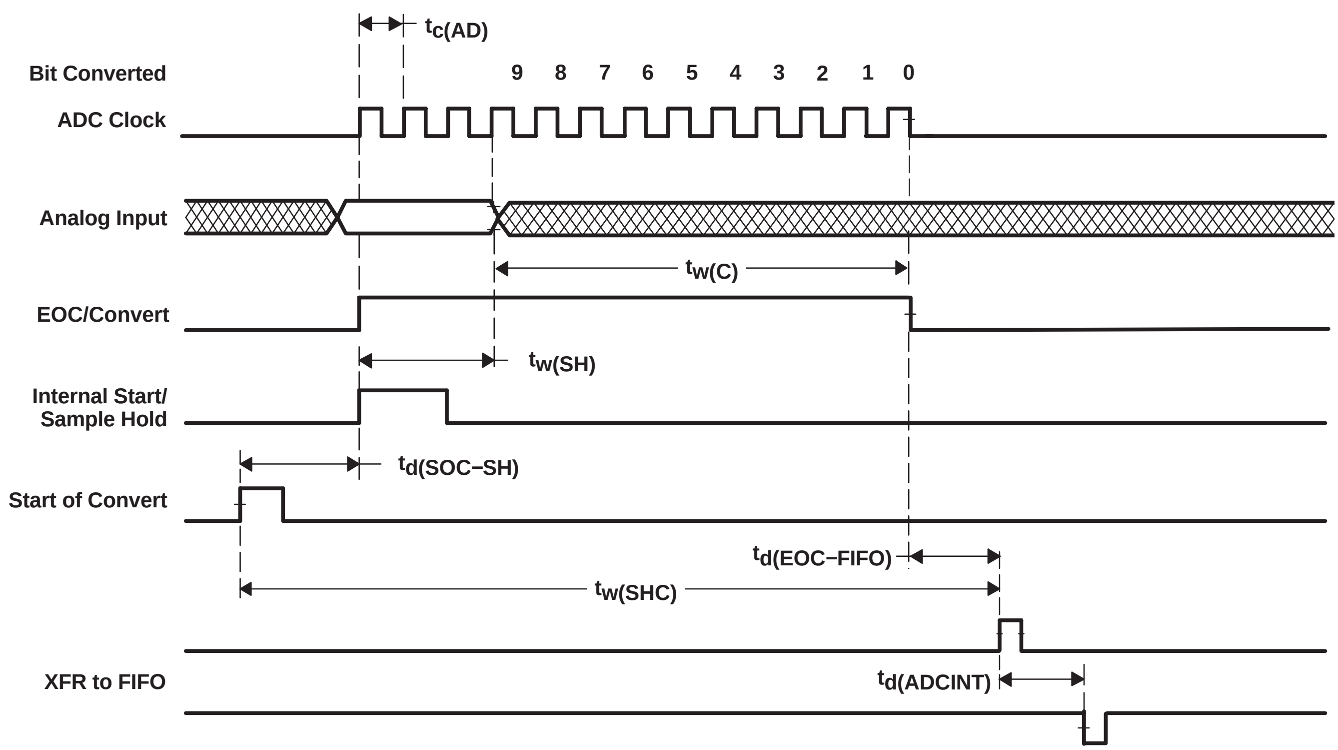


Figure 31. Analog-to-Digital Internal Module Timing

TMS320F241-to-TMS320C242 migration

Table 14 outlines the differences between the TMS320F241 and the TMS320C242 devices, which are important to consider while migrating from one to the other.

Table 14. Differences Between the TMS320F241 and the TMS320C242 Devices

SERIAL NO.	FUNCTION	TMS320F241	TMS320C242
1	Program memory size (on-chip, nonvolatile)	Flash size: 8K words	ROM size: 4K words
2	PMT pin (condition for proper device operation)	PMT pin must be connected to ground	PMT pin can be connected to ground or left unconnected
3	Pullups on the following pins (150 µA typical): $\overline{\text{BIO}}$/IOPC1 $\overline{\text{NMI}}$ XF/IOPC0	Not available	Has pullups on all these pins
4	Pulldowns on the following pins (150 µA typical): T1CMP/T1PWM/IOPB4 TDIR/IOPB6 CAP1/QEP0/IOPA3 CAP2/QEP1/IOPA4 CAP3/IOPA5 IOPC2 IOPC3 IOPC4 IOPC5 IOPC6 IOPC7 XINT1/IOPA2	Not available	Has pulldowns on all these pins
5	SPI and CAN	SPI and CAN are available	SPI and CAN are absent. The corresponding register spaces are illegal and would result in an NMI if accessed.
6	Access to external memory spaces	Access to external Program, Data and I/O space is considered illegal and would assert an NMI in the F241.	The external Program and I/O spaces are implemented as “reserved” addresses and any access will not assert an NMI. However, the external data memory space is illegal.
7	OCRB register	Bits 2−7 select SPI/CAN or GPIO functionality.	These bits are reserved.
8	V_{CCP}/WDDIS pin	Functions are as described in the data sheet.	This pin is ony a WDDIS pin. There is no V_{CCP} functionality, as this is a ROM part.
9	Bit definitions in PIRQRn and PIACKRn registers	Definitions are as described in the data sheet/Reference Guide.	The bit definitions for these two registers differ from F243/F241. See Table 5.
10	NMI and PDPINT interrupts	(Low) level-triggered	(Falling) edge-triggered

register file compilation

Table 15 is a collection of all the programmable registers of the TMS320x24x (provided for a quick reference).

Table 15. Register File Compilation

ADDR	BIT 15 / BIT 7	BIT 14 / BIT 6	BIT 13 / BIT 5	BIT 12 / BIT 4	BIT 11 / BIT 3	BIT 10 / BIT 2	BIT 9 / BIT 1	BIT 8 / BIT 0	REG
	DATA MEMORY SPACE								
	CPU STATUS REGISTERS								
	ARP			OV	OVM	1	INTM	DP(8)	ST0
	DP(7)	DP(6)	DP(5)	DP(4)	DP(3)	DP(2)	DP(1)	DP(0)	
	ARB			CNF	TC	SXM	C	1	ST1
	1	1	1	XF	1	1	PM		
	GLOBAL MEMORY AND CPU INTERRUPT REGISTERS								
00004h	—	—	—	—	—	—	—	—	IMR
	—	—	INT6 MASK	INT5 MASK	INT4 MASK	INT3 MASK	INT2 MASK	INT1 MASK	
00005h	—	—	—	—	—	—	—	—	GREG
	Global Data Memory Configuration Bits (7−0)								
00006h	—	—	—	—	—	—	—	—	IFR
	—	—	INT6 FLAG	INT5 FLAG	INT4 FLAG	INT3 FLAG	INT2 FLAG	INT1 FLAG	
	SYSTEM REGISTERS								
07010h	IRQ0.15	IRQ0.14	IRQ0.13	IRQ0.12	IRQ0.11	IRQ0.10	IRQ0.9	IRQ0.8	PIRQR0
	IRQ0.7	IRQ0.6	IRQ0.5	IRQ0.4	IRQ0.3	IRQ0.2	IRQ0.1	IRQ0.0	
07011h	IRQ1.15	IRQ1.14	IRQ1.13	IRQ1.12	IRQ1.11	IRQ1.10	IRQ1.9	IRQ1.8	PIRQR1
	IRQ1.7	IRQ1.6	IRQ1.5	IRQ1.4	IRQ1.3	IRQ1.2	IRQ1.1	IRQ1.0	
07012h to 07013h	Illegal								
07014h	IAK0.15	IAK0.14	IAK0.13	IAK0.12	IAK0.11	IAK0.10	IAK0.9	IAK0.8	PIACKR0
	IAK0.7	IAK0.6	IAK0.5	IAK0.4	IAK0.3	IAK0.2	IAK0.1	IAK0.0	
07015h	IAK1.15	IAK1.14	IAK1.13	IAK1.12	IAK1.11	IAK1.10	IAK1.9	IAK1.8	PIACKR1
	IAK1.7	IAK1.6	IAK1.5	IAK1.4	IAK1.3	IAK1.2	IAK1.1	IAK1.0	
07016h to 07017h	Illegal								
07018h	—	CLKSRC	LPM1	LPM0	—	—	—	—	SCSR
	—	—	—	—	—	—	—	ILLADR	
07019h to 0701Bh	Illegal								
0701Ch	DIN15	DIN14	DIN13	DIN12	DIN11	DIN10	DIN9	DIN8	DINR
	DIN7	DIN6	DIN5	DIN4	DIN3	DIN2	DIN1	DIN0	
0701Dh	Illegal								
0701Eh	V15	V14	V13	V12	V11	V10	V9	V8	PIVR
	V7	V6	V5	V4	V3	V2	V1	V0	
0701Fh	Illegal								

register file compilation (continued)

Table 15. Register File Compilation (Continued)

ADDR	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8	REG
	BIT 7	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**	
	WD CONTROL REGISTERS								
07020h to 07022h	Illegal								
07023h	D7	D6	D5	D4	D3	D2	D1	D0	WDCNTR
07024h	Illegal								
07025h	D7	D6	D5	D4	D3	D2	D1	D0	WDKEY
07026h to 07028h	Illegal								
07029h	WD FLAG	WDDIS	WDCHK2	WDCHK1	WDCHK0	WDPS2	WDPS1	WDPS0	WDCR
0702Ah to 0702Ch	Illegal								
0702Dh to 07031h	Illegal								
	A-to-D MODULE CONTROL REGISTERS								
07032h	SUSPEND-SOFT	SUSPEND-FREE	ADCIM-START	ADC2EN	ADC1EN	ADCCON-RUN	ADCINTEN	ADCINTFLAG	ADCTRL1
	ADCEOC	ADC2CHSEL			ADC1CHSEL			ADCSOC	
07033h	Illegal								
07034h	—	—	EVSOCP	EXTSOCP	INTPRI	ADCEVSOC	ADCEXTSOC	—	ADCTRL2
	ADCFIFO2		—	ADCFIFO1		ADCPSCALE			
07035h	Illegal								
07036h	D9	D8	D7	D6	D5	D4	D3	D2	ADCFIFO1
	D1	D0	0	0	0	0	0	0	
07037h	Illegal								
07038h	D9	D8	D7	D6	D5	D4	D3	D2	ADCFIFO2
	D1	D0	0	0	0	0	0	0	
07039h to 0703Fh	Illegal								
07040h to 0704Fh	Illegal								
	SERIAL COMMUNICATIONS INTERFACE (SCI) CONFIGURATION CONTROL REGISTERS								
07050h	STOP BITS	EVEN/ODD PARITY	PARITY ENABLE	LOOP BACK ENA	ADDR/IDLE MODE	SCI CHAR2	SCI CHAR1	SCI CHAR0	SCICCR
07051h	—	RX ERR INT ENA	SW RESET	—	TXWAKE	SLEEP	TXENA	RXENA	SCICTL1
07052h	BAUD15 (MSB)	BAUD14	BAUD13	BAUD12	BAUD11	BAUD10	BAUD9	BAUD8	SCIHBAUD
07053h	BAUD7	BAUD6	BAUD5	BAUD4	BAUD3	BAUD2	BAUD1	BAUD0 (LSB)	SCILBAUD
07054h	TXRDY	TX EMPTY	—	—	—	—	RX/BK INT ENA	TX INT ENA	SCICTL2

register file compilation (continued)

Table 15. Register File Compilation (Continued)

ADDR	BIT 15 / BIT 7	BIT 14 / BIT 6	BIT 13 / BIT 5	BIT 12 / BIT 4	BIT 11 / BIT 3	BIT 10 / BIT 2	BIT 9 / BIT 1	BIT 8 / BIT 0	REG
	SERIAL COMMUNICATIONS INTERFACE (SCI) CONFIGURATION CONTROL REGISTERS (CONTINUED)								
07055h	RX ERROR	RXRDY	BRKDT	FE	OE	PE	RXWAKE	—	SCIRXST
07056h	ERXDT7	ERXDT6	ERXDT5	ERXDT4	ERXDT3	ERXDT2	ERXDT1	ERXDT0	SCIRXEMU
07057h	RXDT7	RXDT6	RXDT5	RXDT4	RXDT3	RXDT2	RXDT1	RXDT0	SCIRXBUF
07058h	Illegal								
07059h	TXDT7	TXDT6	TXDT5	TXDT4	TXDT3	TXDT2	TXDT1	TXDT0	SCITXBUF
0705Ah to 0705Eh	Illegal								
0705Fh	—	SCITX PRIORITY	SCIRX PRIORITY	SCI SOFT	SCI FREE	—	—	—	SCIPRI
07060h to 0706Fh	Illegal								
	EXTERNAL INTERRUPT CONTROL REGISTERS								
07070h	XINT1 FLAG	—	—	—	—	—	—	—	XINT1CR
	—	—	—	—	—	XINT1 POLARITY	XINT1 PRIORITY	XINT1 ENA	
07071h	XINT2 FLAG	—	—	—	—	—	—	—	XINT2CR
	—	—	—	—	—	XINT2 POLARITY	XINT2 PRIORITY	XINT2 ENA	
07072h to 0708Fh	Illegal								
	DIGITAL I/O CONTROL REGISTERS								
07090h	CRA.15	CRA.14	CRA.13	CRA.12	CRA.11	CRA.10	CRA.9	CRA.8	OCRA
	CRA.7	CRA.6	CRA.5	CRA.4	CRA.3	CRA.2	CRA.1	CRA.0	
07091h	Illegal								
07092h	—	—	—	—	—	—	CRB.9	CRB.8	OCRB
	—	—	—	—	—	—	CRB.1	CRB.0	
07093h to 07097h	Illegal								
07098h	A7DIR	A6DIR	A5DIR	A4DIR	A3DIR	A2DIR	A1DIR	A0DIR	PADATDIR
	IOPA7	IOPA6	IOPA5	IOPA4	IOPA3	IOPA2	IOPA1	IOPA0	
07099h	Illegal								
0709Ah	B7DIR	B6DIR	B5DIR	B4DIR	B3DIR	B2DIR	B1DIR	B0DIR	PBDATDIR
	IOPB7	IOPB6	IOPB5	IOPB4	IOPB3	IOPB2	IOPB1	IOPB0	
0709Bh	Illegal								
0709Ch	C7DIR	C6DIR	C5DIR	C4DIR	C3DIR	C2DIR	C1DIR	C0DIR	PCDATDIR
	IOPC7	IOPC6	IOPC5	IOPC4	IOPC3	IOPC2	IOPC1	IOPC0	
0709Dh	Illegal								
0709Eh	Reserved						D1DIR	D0DIR	PDDATDIR
							IOPD1	IOPD0	

register file compilation (continued)

Table 15. Register File Compilation (Continued)

ADDR	BIT 15 / BIT 7	BIT 14 / BIT 6	BIT 13 / BIT 5	BIT 12 / BIT 4	BIT 11 / BIT 3	BIT 10 / BIT 2	BIT 9 / BIT 1	BIT 8 / BIT 0	REG
	DIGITAL I/O CONTROL REGISTERS (CONTINUED)								
0709Fh	Illegal								
070A0h to 070FFh	Illegal								
07100h to 071FFh	Illegal								
07200h to 0723Fh	Illegal								
07240h to 073FFh	Illegal								
	GENERAL-PURPOSE (GP) TIMER CONFIGURATION CONTROL REGISTERS								
07400h	—	T2STAT	T1STAT	—		T2TOADC		T1TOADC(1)	GPTCON
	T1TOADC(0)	TCOMPOE	—		T2PIN		T1PIN		
07401h	D15	D14	D13	D12	D11	D10	D9	D8	T1CNT
	D7	D6	D5	D4	D3	D2	D1	D0	
07402h	D15	D14	D13	D12	D11	D10	D9	D8	T1CMPR
	D7	D6	D5	D4	D3	D2	D1	D0	
07403h	D15	D14	D13	D12	D11	D10	D9	D8	T1PR
	D7	D6	D5	D4	D3	D2	D1	D0	
07404h	FREE	SOFT	—	TMODE1	TMODE0	TPS2	TPS1	TPS0	T1CON
	TSWT1	TENABLE	TCLKS1	TCLKS0	TCLD1	TCLD0	TECMPR	SELT1PR	
07405h	D15	D14	D13	D12	D11	D10	D9	D8	T2CNT
	D7	D6	D5	D4	D3	D2	D1	D0	
07406h	D15	D14	D13	D12	D11	D10	D9	D8	T2CMPR
	D7	D6	D5	D4	D3	D2	D1	D0	
07407h	D15	D14	D13	D12	D11	D10	D9	D8	T2PR
	D7	D6	D5	D4	D3	D2	D1	D0	
07408h	FREE	SOFT	—	TMODE1	TMODE0	TPS2	TPS1	TPS0	T2CON
	TSWT1	TENABLE	TCLKS1	TCLKS0	TCLD1	TCLD0	TECMPR	SELT1PR	
07409h to 07410h	Illegal								
	FULL AND SIMPLE COMPARE UNIT REGISTERS								
07411h	CENABLE	CLD1	CLD0	SVENABLE	ACTRLD1	ACTRLD0	FCOMPOE	—	COMCON
	—	—	—	—	—	—	—	—	
07412h	Illegal								
07413h	SVRDIR	D2	D1	D0	CMP6ACT1	CMP6ACT0	CMP5ACT1	CMP5ACT0	ACTR
	CMP4ACT1	CMP4ACT0	CMP3ACT1	CMP3ACT0	CMP2ACT1	CMP2ACT0	CMP1ACT1	CMP1ACT0	
07414h	Illegal								

register file compilation (continued)

Table 15. Register File Compilation (Continued)

ADDR	BIT 15 / BIT 7	BIT 14 / BIT 6	BIT 13 / BIT 5	BIT 12 / BIT 4	BIT 11 / BIT 3	BIT 10 / BIT 2	BIT 9 / BIT 1	BIT 8 / BIT 0	REG
	FULL AND SIMPLE COMPARE UNIT REGISTERS (CONTINUED)								
07415h	—	—	—	—	DBT3	DBT2	DBT1	DBT0	DBTCON
	EDBT3	EDBT2	EDBT1	DBTPS2	DBTPS1	DBTPS0	—	—	
07416h	Illegal								
07417h	D15	D14	D13	D12	D11	D10	D9	D8	CMPR1
	D7	D6	D5	D4	D3	D2	D1	D0	
07418h	D15	D14	D13	D12	D11	D10	D9	D8	CMPR2
	D7	D6	D5	D4	D3	D2	D1	D0	
07419h	D15	D14	D13	D12	D11	D10	D9	D8	CMPR3
	D7	D6	D5	D4	D3	D2	D1	D0	
0741Ah to 0741Fh	Illegal								
	CAPTURE UNIT REGISTERS								
07420h	CAPRES	CAPQEPN		CAP3EN	—	CAP3TSEL	CAP12TSEL	CAP3TOADC	CAPCON
	CAP1EDGE		CAP2EDGE		CAP3EDGE		—		
07421h	Illegal								
07422h	—		CAP3FIFO		CAP2FIFO		CAP1FIFO		CAPFIFO
	—	—	—	—	—	—	—	—	
07423h	D15	D14	D13	D12	D11	D10	D9	D8	CAP1FIFO
	D7	D6	D5	D4	D3	D2	D1	D0	
07424h	D15	D14	D13	D12	D11	D10	D9	D8	CAP2FIFO
	D7	D6	D5	D4	D3	D2	D1	D0	
07425h	D15	D14	D13	D12	D11	D10	D9	D8	CAP3FIFO
	D7	D6	D5	D4	D3	D2	D1	D0	
07426h	Illegal								
07427h	D15	D14	D13	D12	D11	D10	D9	D8	CAP1FBOT
	D7	D6	D5	D4	D3	D2	D1	D0	
07428h	D15	D14	D13	D12	D11	D10	D9	D8	CAP2FBOT
	D7	D6	D5	D4	D3	D2	D1	D0	
07429h	D15	D14	D13	D12	D11	D10	D9	D8	CAP3FBOT
	D7	D6	D5	D4	D3	D2	D1	D0	
0742Ah to 0742Bh	Illegal								

register file compilation (continued)

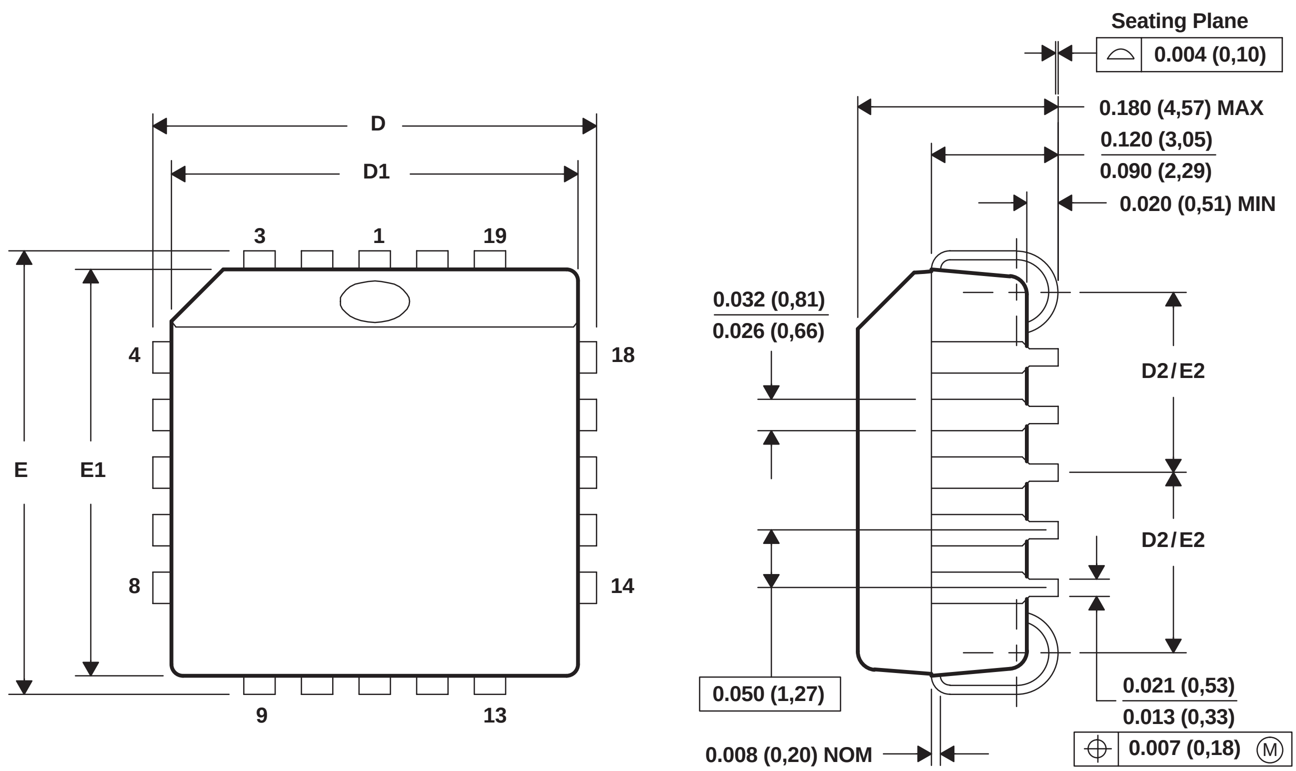
Table 15. Register File Compilation (Continued)

ADDR	BIT 15 / BIT 7	BIT 14 / BIT 6	BIT 13 / BIT 5	BIT 12 / BIT 4	BIT 11 / BIT 3	BIT 10 / BIT 2	BIT 9 / BIT 1	BIT 8 / BIT 0	REG
	EVENT MANAGER (EV) INTERRUPT CONTROL REGISTERS								
0742Ch	—	—	—	—	—	T1OFINT ENA	T1UFINT ENA	T1CINT ENA	EVIMRA
	T1PINT ENA	—	—	—	CMP3INT ENA	CMP2INT ENA	CMP1INT ENA	PDPINT ENA	
0742Dh	—	—	—	—	—	—	—	—	EVIMRB
	—	—	—	—	T2OFINT ENA	T2UFINT ENA	T2CINT ENA	T2PINT ENA	
0742Eh	—	—	—	—	—	—	—	—	EVIMRC
	—	—	—	—	—	CAP3INT ENA	CAP2INT ENA	CAP1INT ENA	
0742Fh	—	—	—	—	—	T1OFINT FLAG	T1UFINT FLAG	T1CINT FLAG	EVIFRA
	T1PINT FLAG	—	—	—	CMP3INT FLAG	CMP2INT FLAG	CMP1INT FLAG	PDPINT FLAG	
07430h	—	—	—	—	—	—	—	—	EVIFRB
	—	—	—	—	T2OFINT FLAG	T2UFINT FLAG	T2CINT FLAG	T2PINT FLAG	
07431h	—	—	—	—	—	—	—	—	EVIFRC
	—	—	—	—	—	CAP3INT FLAG	CAP2INT FLAG	CAP1INT FLAG	
07432h to 0743Fh	Illegal								
07440h to 07FFFh	Illegal								

MECHANICAL DATA

FN (S-PQCC-J)** **PLASTIC J-LEADED CHIP CARRIER**

20 PIN SHOWN



NO. OF PINS **	D / E		D1 / E1		D2 / E2	
	MIN	MAX	MIN	MAX	MIN	MAX
20	0.385 (9,78)	0.395 (10,03)	0.350 (8,89)	0.356 (9,04)	0.141 (3,58)	0.169 (4,29)
28	0.485 (12,32)	0.495 (12,57)	0.450 (11,43)	0.456 (11,58)	0.191 (4,85)	0.219 (5,56)
44	0.685 (17,40)	0.695 (17,65)	0.650 (16,51)	0.656 (16,66)	0.291 (7,39)	0.319 (8,10)
52	0.785 (19,94)	0.795 (20,19)	0.750 (19,05)	0.756 (19,20)	0.341 (8,66)	0.369 (9,37)
68	0.985 (25,02)	0.995 (25,27)	0.950 (24,13)	0.958 (24,33)	0.441 (11,20)	0.469 (11,91)
84	1.185 (30,10)	1.195 (30,35)	1.150 (29,21)	1.158 (29,41)	0.541 (13,74)	0.569 (14,45)

4040005 / B 03/95

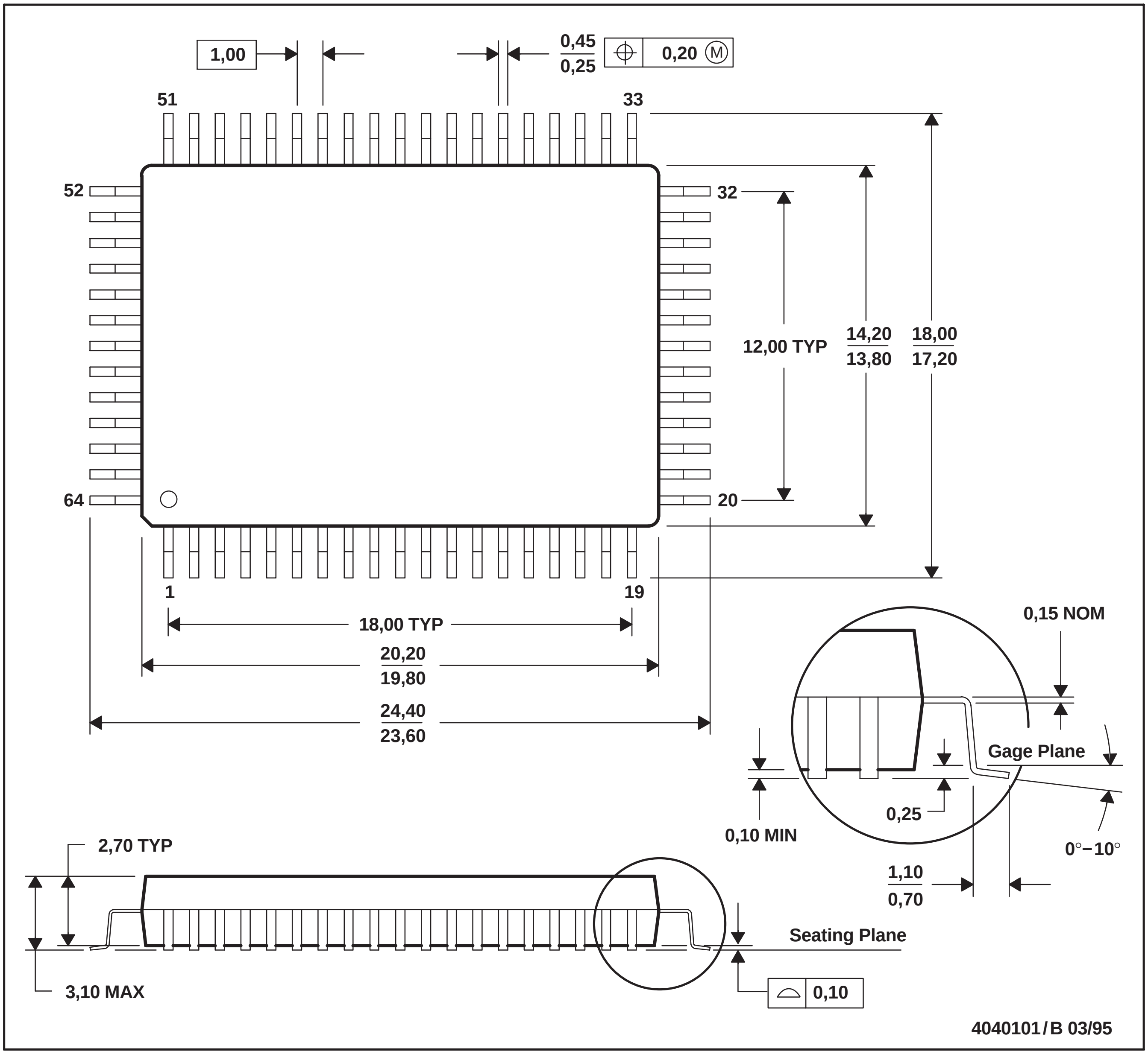
NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-018

Typical Thermal Resistance Characteristics

PARAMETER	DESCRIPTION	°C / W
Θ_{JA}	Junction-to-ambient	48
Θ_{JC}	Junction-to-case	11

MECHANICAL DATA

PG (R-PQFP-G64) **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Contact field sales office to determine if a tighter coplanarity requirement is available for this package.

Typical Thermal Resistance Characteristics

PARAMETER	DESCRIPTION	°C/W
Θ_{JA}	Junction-to-ambient	35
Θ_{JC}	Junction-to-case	11

PACKAGING INFORMATION

Orderable Device	Status [1]	Package Type	Package Drawing	Pins	Package Qty	Eco Plan [2]	Lead/Ball Finish	MSL Peak Temp [3]
TMS320C242PGA	OBSOLETE	QFP	PG	64		TBD	Call TI	Call TI
TMS320C242PGS	OBSOLETE	QFP	PG	64		TBD	Call TI	Call TI

[1] The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

[2] Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check http://www.ti.com/productcontent for the latest availability information and additional product content details.
TBD: The Pb-Free/Green conversion plan has not been defined.
Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.
Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.
Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

[3] MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

Important Information and Disclaimer: The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.