SGUS029 – APRIL 1999

- **Processed to MIL–PRF–38535 (QML)**
- **High-Performance Static CMOS Technology**
- **Includes the T320C2xLP Core CPU**
  - **Object Compatible With the TMS320C2xx Family**
  - **Source Code Compatible With SMJ320C25**
  - **Upwardly Compatible With SMJ320C50**
  - **50-ns Instruction Cycle Time**
- **Memory**
  - **544 Words × 16 Bits of On-Chip Data/Program Dual-Access RAM**
  - **16K Words × 16 Bits of On-Chip Program Flash EEPROM**
  - **224K Words × 16 Bits of Total Memory Address Reach (64K Data, 64K Program and 64K I/O, and 32K Global Memory Space)**
- **Event-Manager Module**
  - **12 Compare/Pulse-Width Modulation (PWM) Channels**
  - **Three 16-Bit General-Purpose Timers With Six Modes, Including Continuous Up and Up/Down Counting**
  - **Three 16-Bit Full-Compare Units With Deadband**
  - **Three 16-Bit Simple-Compare Units**
  - **Four Capture Units (Two With Quadrature Encoder-Pulse Interface Capability)**
- **Dual 10-Bit Analog-to-Digital Conversion Module**
- **28 Individually Programmable, Multiplexed I/O Pins**
- **Phase-Locked-Loop (PLL)-Based Clock Module**
- **Watchdog Timer Module (With Real-Time Interrupt)**
- **Serial Communications Interface (SCI) Module**
- **Serial Peripheral Interface (SPI) Module**
- **Six External Interrupts (Power Drive Protect, Reset, NMI, and Three Maskable Interrupts)**
- **Four 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**
  - **Scan-Based Self-Emulation (XDS510™)**
  - **Third-Party Digital Motor Control and Fuzzy-Logic Development Support**
- **–55°C to 125°C Operating Temperature Range, QML Processing**
- **132-Pin Ceramic Quad Flat Package (HFP Suffix)**

description

The SMJ320F240 is the first member of a new family of digital signal processor (DSP) controllers based on the TMS320C2xx generation of 16-bit fixed-point DSPs. This new family is optimized for digital motor/motion control applications and contains 16K words of flash memory on chip. The DSP controller combines the enhanced TMS320 architectural design of the 'C2xLP 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 compare registers to generate up to 12 PWM outputs, and a dual 10-bit analog-to-digital converter (ADC), which can perform two simultaneous conversions within 6.1 μs.

Table of Contents

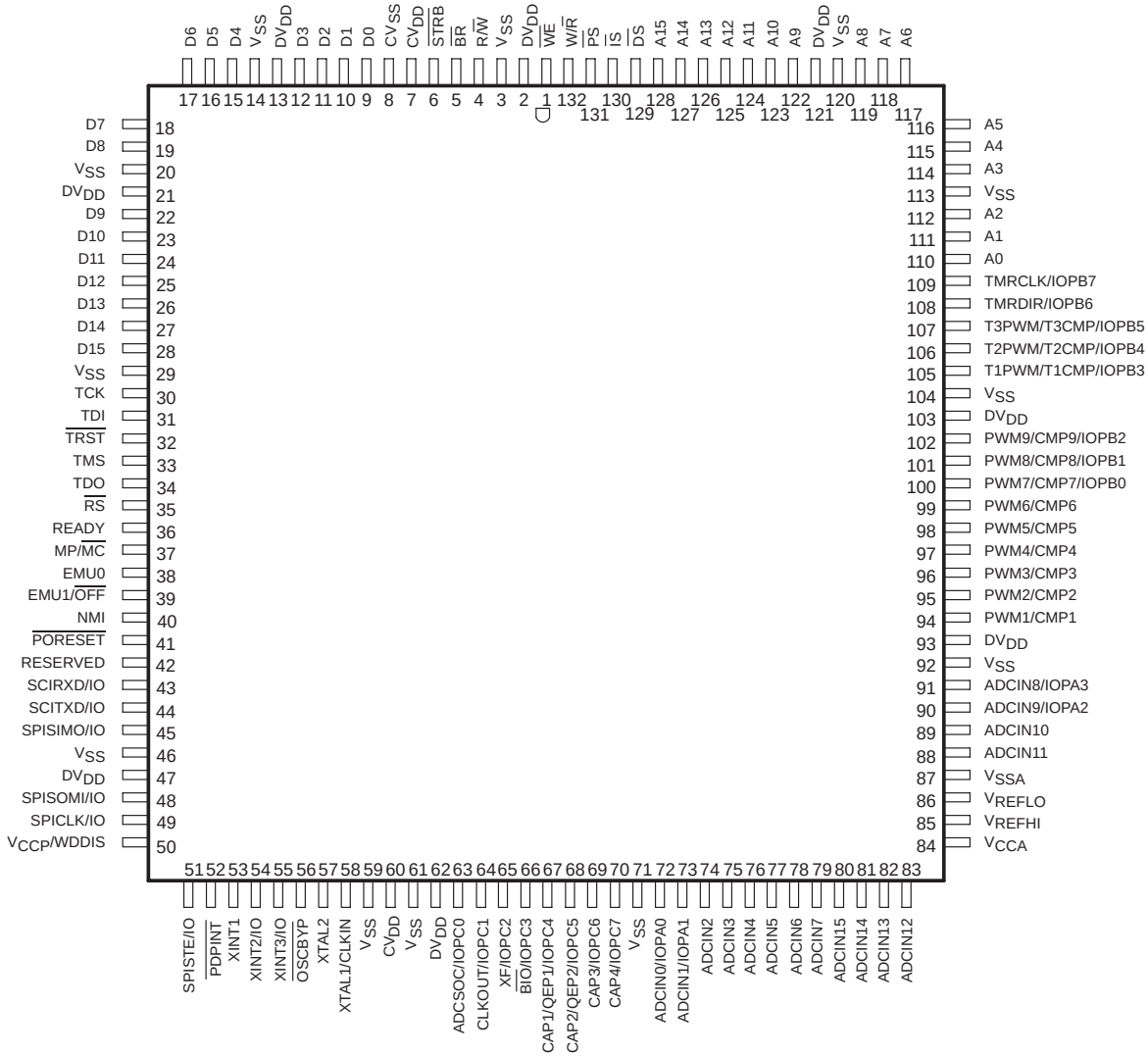
description (continued)

Table 1. Characteristics of the ’F240 DSP Controller

DEVICE	ON-CHIP MEMORY (WORDS)			POWER SUPPLY (V)	CYCLE TIME (ns)	PACKAGE TYPE PIN COUNT
	RAM		FLASH EEPROM			
	DATA	DATA/PROG	PROG			
SMJ320F240	288	256	16K	5	50	HFP 132–P

The functional block diagram provides a high-level description of each component in the ’F240 DSP controller. The SMJ320F240 device is 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 ’F240 that are distributed. These system features include the memory map, device reset, interrupts, digital input/output (I/O), clock generation, and low-power operation.

pinout


1 D6
2 D5
3 D4
4 VSS
5 DVDD
6 D3
7 D2
8 D1
9 D0
10 CVSS
11 CVDD
12 STRB
13 BR
14 R/W
15 VSS
16 DVDD
17 WE
18 W/R
19 PS
20 IS
21 DS
22 A15
23 A14
24 A13
25 A12
26 A11
27 A10
28 A9
29 DVDD
30 VSS
31 A8
32 A7
33 A6
34 A5
35 A4
36 A3
37 VSS
38 A2
39 A1
40 A0
41 TMRCLK/IOPB7
42 TMRDIR/IOPB6
43 T3PWM/T3CMP/IOPB5
44 T2PWM/T2CMP/IOPB4
45 T1PWM/T1CMP/IOPB3
46 VSS
47 DVDD
48 PWM9/CMP9/IOPB2
49 PWM8/CMP8/IOPB1
50 PWM7/CMP7/IOPB0
51 PWM6/CMP6
52 PWM5/CMP5
53 PWM4/CMP4
54 PWM3/CMP3
55 PWM2/CMP2
56 PWM1/CMP1
57 DVDD
58 VSS
59 ADCIN8/IOPA3
60 ADCIN9/IOPA2
61 ADCIN10
62 ADCIN11
63 VSSA
64 VREFLO
65 VREFHI
66 VCCA
67 ADCIN12
68 ADCIN13
69 ADCIN14
70 ADCIN15
71 ADCIN7
72 ADCIN6
73 ADCIN5
74 ADCIN4
75 ADCIN3
76 ADCIN2
77 ADCIN1/IOPA1
78 ADCIN0/IOPA0
79 VSS
80 CAP4/IOPC7
81 CAP3/IOPC6
82 CAP2/QEP2/IOPC5
83 CAP1/QEP1/IOPC4
84 BIO/IOPC3
85 XF/IOPC2
86 CLKOUT/IOPC1
87 ADCSOC/IOPC0
88 DVDD
89 VSS
90 CVDD
91 VSS
92 XTAL1/CLKIN
93 XTAL2
94 OSCBYP
95 XINT3/IO
96 XINT2/IO
97 XINT1
98 PDPINT
99 SPISTE/IO
100 SPICLK/IO
101 SPISOMI/IO
102 VSS
103 DVDD
104 SPISIMO/IO
105 SCITXD/IO
106 SCIRXD/IO
107 RESERVED
108 PORESET
109 NMI
110 EMU1/OFF
111 EMU0
112 MP/MC
113 READY
114 RS
115 TDO
116 TMS
117 TRST
118 TDI
119 TCK
120 VSS
121 D15
122 D14
123 D13
124 D12
125 D11
126 D10
127 D9
128 DVDD
129 VSS
130 D8
131 D7
132 VCCP/WDDIS

Terminal Functions

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
EXTERNAL INTERFACE DATA/ADDRESS SIGNALS			
A0 (LSB)	110	O/Z	Parallel address bus A0 [least significant bit (LSB)] through A15 [most significant bit (MSB)]. A15–A0 are multiplexed to address external data/program memory or I/O. A15–A0 are placed in the high-impedance state when EMU1/$\overline{\text{OFF}}$ is active low and hold their previous states in power-down modes.
A1	111		
A2	112		
A3	114		
A4	115		
A5	116		
A6	117		
A7	118		
A8	119		
A9	122		
A10	123		
A11	124		
A12	125		
A13	126		
A14	127		
A15 (MSB)	128		
D0 (LSB)	9	I/O/Z	Parallel data bus D0 (LSB) through D15 (MSB). D15–D0 are multiplexed to transfer data between the SMJ320F240 and external data/program memory and I/O space (devices). D15–D0 are placed in the high-impedance state when not outputting, when in power-down mode, when reset ($\overline{\text{RS}}$) is asserted, or when EMU1/$\overline{\text{OFF}}$ is active low.
D1	10		
D2	11		
D3	12		
D4	15		
D5	16		
D6	17		
D7	18		
D8	19		
D9	22		
D10	23		
D11	24		
D12	25		
D13	26		
D14	27		
D15 (MSB)	28		
EXTERNAL INTERFACE CONTROL SIGNALS			
$\overline{\text{DS}}$	129	O/Z	Data, program, and I/O space select signals. $\overline{\text{DS}}$, $\overline{\text{PS}}$, and $\overline{\text{IS}}$ are always high unless low-level asserted for communication to a particular external space. They are placed in the high-impedance state during reset, power down, and when EMU1/$\overline{\text{OFF}}$ is active low.
$\overline{\text{PS}}$	131		
$\overline{\text{IS}}$	130		
READY	36	I	Data ready. READY indicates that an external device is prepared for the bus transaction to be completed. If the device is not ready (READY is low), the processor waits one cycle and checks READY again.
R/$\overline{\text{W}}$	4	O/Z	Read/write signal. R/$\overline{\text{W}}$ indicates transfer direction during communication to an external device. It is normally in read mode (high), unless low level is asserted for performing a write operation. It is placed in the high-impedance state during reset, power down, and when EMU1/$\overline{\text{OFF}}$ is active low.
$\overline{\text{STRB}}$	6	O/Z	Strobe. $\overline{\text{STRB}}$ is always high unless asserted low to indicate an external bus cycle. It is placed in the high-impedance state during reset, power down, and when EMU1/$\overline{\text{OFF}}$ is active low.
$\overline{\text{WE}}$	1	O/Z	Write enable. The falling edge of $\overline{\text{WE}}$ indicates that the device is driving the external data bus (D15–D0). Data can be latched by an external device on the rising edge of $\overline{\text{WE}}$. $\overline{\text{WE}}$ is active on all external program, data, and I/O writes. $\overline{\text{WE}}$ goes in the high-impedance state following reset and when EMU1/$\overline{\text{OFF}}$ is active low.
W/$\overline{\text{R}}$	132	O/Z	Write/read. W/$\overline{\text{R}}$ is an inverted form of R/$\overline{\text{W}}$ and can connect directly to the output enable of external devices. W/$\overline{\text{R}}$ is placed in the high-impedance state following reset and when EMU1/$\overline{\text{OFF}}$ is active low.

† I = input, O = output, Z = high impedance

Terminal Functions (Continued)

TERMINAL NAME	NO.	TYPE†	DESCRIPTION
EXTERNAL INTERFACE CONTROL SIGNALS (CONTINUED)			
$\overline{BR}$	5	O/Z	Bus request. $\overline{BR}$ is asserted during access of external global data memory space. $\overline{BR}$ can be used to extend the data memory address space by up to 32K words. $\overline{BR}$ goes in the high-impedance state during reset, power down, and when EMU1/$\overline{OFF}$ is active low.
V_{CCP}/WDDIS	50	I	Flash-programming voltage supply. If V_{CCP} = 5 V, then WRITE/ERASE can be made to the ENTIRE on-chip flash memory block—that is, for programming the flash. If V_{CCP} = 0 V, then WRITE/ERASE of the flash memory is not allowed, thereby protecting the entire memory block from being overwritten. WDDIS also functions as a hardware watchdog disable. The watchdog timer is disabled when V_{CCP}/WDDIS = 5 V and bit 6 in WDCR is set to 1.
ADC INPUTS (UNSHARED)			
ADCIN2	74	I	Analog inputs to the first ADC
ADCIN3	75	I	
ADCIN4	76	I	
ADCIN5	77	I	
ADCIN6	78	I	
ADCIN7	79	I	
ADCIN10	89	I	Analog inputs to the second ADC
ADCIN11	88	I	
ADCIN12	83	I	
ADCIN13	82	I	
ADCIN14	81	I	
ADCIN15	80	I	
BIT I/O AND SHARED FUNCTIONS PINS			
ADCIN0/IOPA0	72	I/O	Bidirectional digital I/O. Analog input to the first ADC. ADCIN0/IOPA0 is configured as a digital input by all device resets.
ADCIN1/IOPA1	73	I/O	Bidirectional digital I/O. Analog input to the first ADC. ADCIN1/IOPA1 is configured as a digital input by all device resets.
ADCIN9/IOPA2	90	I/O	Bidirectional digital I/O. Analog input to the second ADC. ADCIN9/IOPA2 is configured as a digital input by all device resets.
ADCIN8/IOPA3	91	I/O	Bidirectional digital I/O. Analog input to the second ADC. ADCIN8/IOPA3 is configured as a digital input by all device resets.
PWM7/CMP7/IOPB0	100	I/O/Z	Bidirectional digital I/O. Simple compare/PWM 1 output. The state of PWM7/CMP7/IOPB0 is determined by the simple compare/PWM and the simple action control register (SACTR). It goes to the high-impedance state when unmasked $\overline{PDPINT}$ goes active low. PWM7/CMP7/IOPB0 is configured as a digital input by all device resets.
PWM8/CMP8/IOPB1	101	I/O/Z	Bidirectional digital I/O. Simple compare/PWM 2 output. The state of PWM8/CMP8/IOPB1 is determined by the simple compare/PWM and the SACTR. It goes to the high-impedance state when unmasked $\overline{PDPINT}$ goes active low. PWM8/CMP8/IOPB1 is configured as a digital input by all device resets.
PWM9/CMP9/IOPB2	102	I/O/Z	Bidirectional digital I/O. Simple compare/PWM 3 output. The state of PWM9/CMP9/IOPB2 is determined by the simple compare/PWM and SACTR. It goes to the high-impedance state when unmasked $\overline{PDPINT}$ goes active low. PWM9/CMP9/IOPB2 is configured as a digital input by all device resets.

† I = input, O = output, Z = high impedance

Terminal Functions (Continued)

TERMINAL NAME	NO.	TYPE†	DESCRIPTION
BIT I/O AND SHARED FUNCTIONS PINS (CONTINUED)			
T1PWM/T1CMP/ IOPB3	105	I/O/Z	Bidirectional digital I/O. Timer 1 compare output. T1PWM/T1CMP/IOPB3 goes to the high-impedance state when unmasked PDPINT goes active low. This pin is configured as a digital input by all device resets.
T2PWM/T2CMP/ IOPB4	106	I/O/Z	Bidirectional digital I/O. Timer 2 compare output. T2PWM/T2CMP/IOPB4 goes to the high-impedance state when unmasked PDPINT goes active low. This pin is configured as a digital input by all device resets.
T3PWM/T3CMP/ IOPB5	107	I/O/Z	Bidirectional digital I/O. Timer 3 compare output. T3PWM/T3CMP/IOPB5 goes to the high-impedance state when unmasked PDPINT goes active low. This pin is configured as a digital input by all device resets.
TMRDIR/IOPB6	108	I/O	Bidirectional digital I/O. Direction signal for the timers. Up-counting direction if TMRDIR/IOPB6 is low, down-counting direction if this pin is high. This pin is configured as a digital input by all device resets.
TMRCLK/IOPB7	109	I/O	Bidirectional digital I/O. External clock input for general-purpose timers. This pin is configured as a digital input by all device resets.
ADCSOC/IOPC0	63	I/O	Bidirectional digital I/O. External start of conversion input for ADC. This pin is configured as a digital input by all device resets.
CAP1/QEP1/IOPC4	67	I/O	Bidirectional digital I/O. Capture 1 or QEP 1 input. This pin is configured as a digital input by all device resets.
CAP2/QEP2/IOPC5	68	I/O	Bidirectional digital I/O. Capture 2 or QEP 2 input. This pin is configured as a digital input by all device resets.
CAP3/IOPC6	69	I/O	Bidirectional digital I/O. Capture 3 input. This pin is configured as a digital input by all device resets.
CAP4/IOPC7	70	I/O	Bidirectional digital I/O. Capture 4 input. This pin is configured as a digital input by all device resets.
XF/IOPC2	65	I/O	Bidirectional digital I/O. External flag output (latched software-programmable signal). XF is used for signaling other processors in multiprocessing configurations or as a general-purpose output pin. This pin is configured as an external flag output by all device resets.
$\overline{\text{BIO}}$/IOPC3	66	I/O	Bidirectional digital I/O. Branch control input. $\overline{\text{BIO}}$ is polled by the BIOZ instruction. If $\overline{\text{BIO}}$ is low, the CPU executes a branch. 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.
CLKOUT/IOPC1	64	I/O	Bidirectional digital I/O. Clock output. Clock output is selected by the CLKSRC bits in the SYSCR register. This pin is configured as a DSP clock output by a power-on reset.
SERIAL COMMUNICATIONS INTERFACE (SCI) AND BIT I/O PINS			
SCITXD/IO	44	I/O	SCI asynchronous serial port transmit data, or general-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
SCIRXD/IO	43	I/O	SCI asynchronous serial port receive data, or general-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.

† I = input, O = output, Z = high impedance

Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
SERIAL PERIPHERAL INTERFACE (SPI) AND BIT I/O PINS			
SPISIMO/IO	45	I/O	SPI slave in, master out , or general-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
SPISOMI/IO	48	I/O	SPI slave out, master in, or general-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
SPICLK/IO	49	I/O	SPI clock, or general-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
SPISTE/IO	51	I/O	SPI slave transmit enable (optional), or general-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
COMPARE SIGNALS			
PWM1/CMP1 PWM2/CMP2 PWM3/CMP3 PWM4/CMP4 PWM5/CMP5 PWM6/CMP6	94 95 96 97 98 99	O/Z	Compare units compare or PWM outputs. The state of these pins is determined by the compare/PWM and the full action control register (ACTR). CMP1–CMP6 go to the high-impedance state when unmasked $\overline{\text{PDPINT}}$ goes active low, and when reset ($\overline{\text{RS}}$) is asserted.
INTERRUPT AND MISCELLANEOUS SIGNALS			
$\overline{\text{RS}}$	35	I/O	Reset input. $\overline{\text{RS}}$ causes the SMJ320F240 to terminate execution and sets PC = 0. When $\overline{\text{RS}}$ is brought to a high level, execution begins at location zero of program memory. $\overline{\text{RS}}$ affects (or sets to zero) various registers and status bits. In addition, $\overline{\text{RS}}$ is a bidirectional (open-drain output) pin. If $\overline{\text{RS}}$ is left undriven, then a 20-Ω pull-up resistor should be used.
MP/$\overline{\text{MC}}$	37	I	MP/$\overline{\text{MC}}$ (microprocessor/microcomputer) select. If MP/$\overline{\text{MC}}$ is low, internal program memory is selected. If it is high, external program memory is selected.
NMI	40	I	Nonmaskable interrupt. When NMI is activated, the device is interrupted regardless of the state of the INTM bit of the status register. NMI has programmable polarity.
$\overline{\text{PORESET}}$	41	I	Power-on reset. $\overline{\text{PORESET}}$ causes the SMJ320F240 to terminate execution and sets PC = 0. When $\overline{\text{PORESET}}$ is brought to a high level, execution begins at location zero of program memory. $\overline{\text{PORESET}}$ affects (or sets to zero) the same registers and status bits as $\overline{\text{RS}}$. In addition, $\overline{\text{PORESET}}$ initializes the PLL control registers.
XINT1	53	I	External user interrupt no. 1
XINT2/IO	54	I/O	External user interrupt no. 2. General-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
XINT3/IO	55	I/O	External user interrupt no. 3. General-purpose bidirectional I/O. This pin is configured as a digital input by all device resets.
$\overline{\text{PDPINT}}$	52	I	Maskable power-drive protection interrupt. If $\overline{\text{PDPINT}}$ is unmasked and it goes active low, the timer compare outputs immediately go to the high-impedance state.
CLOCK SIGNALS			
XTAL2	57	O	PLL oscillator output. XTAL2 is tied to one side of a reference crystal when the device is in PLL mode (CLKMD[1:0] = 1x, CKCR0.7–6). This pin can be left unconnected in oscillator bypass mode ($\overline{\text{OSCBYP}} \le V_{IL}$). This pin goes in the high-impedance state when EMU1/$\overline{\text{OFF}}$ is active low.
XTAL1/CLKIN	58	I/Z	PLL oscillator input. XTAL1/CLKIN is tied to one side of a reference crystal in PLL mode (CLKMD[1:0] = 1x, CKCR0.7–6), or is connected to an external clock source in oscillator bypass mode ($\overline{\text{OSCBYP}} \le V_{IL}$).
$\overline{\text{OSCBYP}}$	56	I	Bypass oscillator if low

† I = input, O = output, Z = high impedance

Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
SUPPLY SIGNALS			
CV_{SS}	8	I	Digital core logic ground reference
V_{SS}	3 14 20 29 46 59 61 71 92 104 113 120	I	Digital logic ground reference
V_{SSA}	87	I	Analog ground reference
DV_{DD} (See Note 1)	2 13 21 47 62 93 103 121	I	Digital I/O logic supply voltage
CV_{DD} (See Note 1)	7 60	I	Digital core logic supply voltage
V_{CCA}	84	I	Analog supply voltage
V_{REFHI}	85	I	ADC analog voltage reference high
V_{REFLO}	86	I	ADC analog voltage reference low
TEST SIGNALS			
TCK	30	I	IEEE standard test clock. TCK is normally a free-running clock signal with a 50% duty cycle. The changes on test-access port (TAP) input signals (TMS and TDI) are clocked into the TAP controller, instruction register, or selected test data register of the 'C2xx core on the rising edge of TCK. Changes at the TAP output signal (TDO) occur on the falling edge of TCK.
TDI	31	I	IEEE standard test data input (TDI). TDI is clocked into the selected register (instruction or data) on a rising edge of TCK.
TDO	34	O/Z	IEEE standard test data output (TDO). The contents of the selected register (instruction or data) are shifted out of TDO on the falling edge of TCK. TDO is in the high-impedance state when $\overline{OFF}$ is active low.
TMS	33	I	IEEE standard test mode select. This serial control input is clocked into the TAP controller on the rising edge of TCK.

† I = input, O = output, Z = high impedance

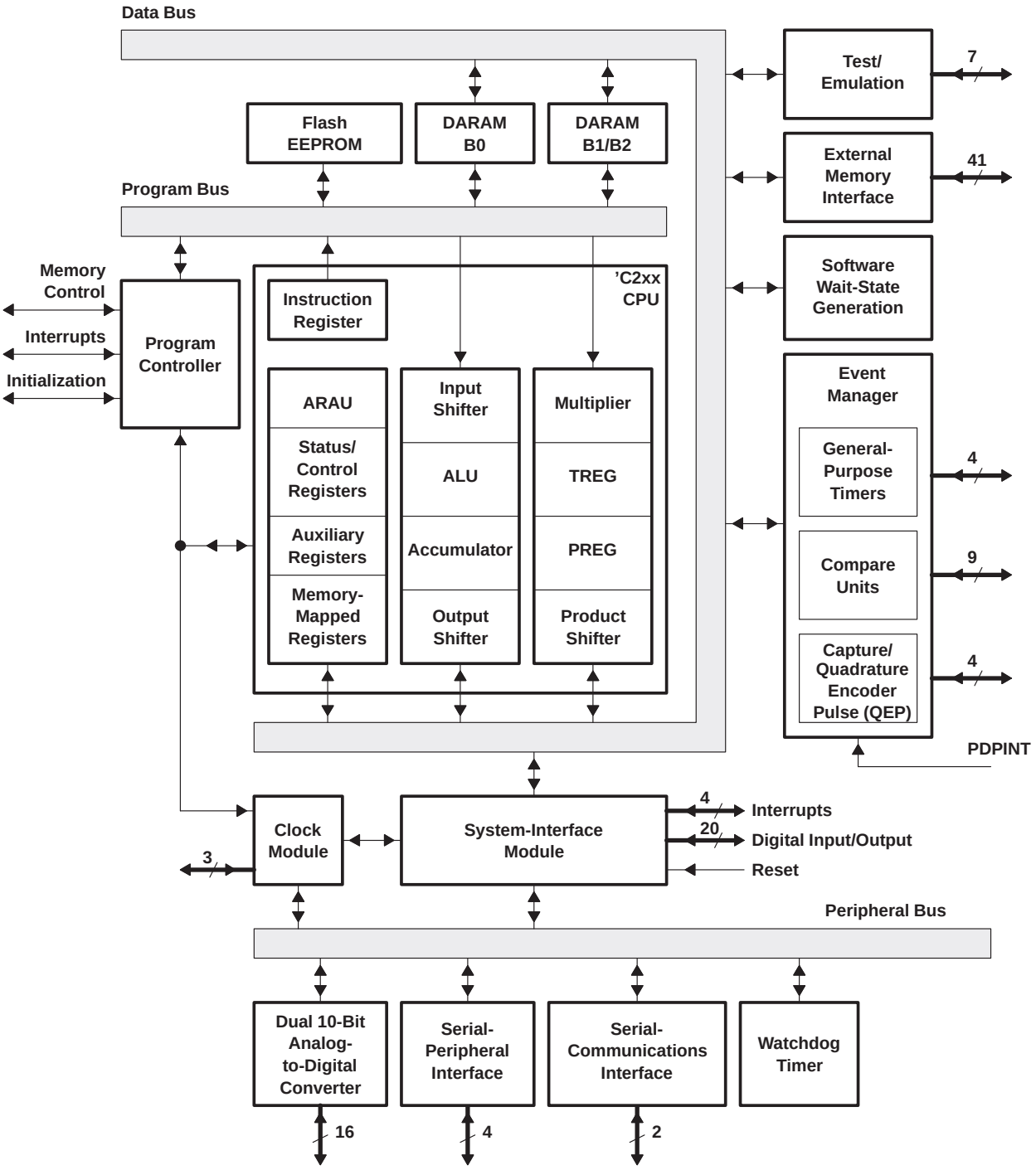
NOTE 1: V_{DD} refers to supply voltage types CV_{DD} (digital core supply voltage), DV_{DD} (digital I/O supply voltage), and V_{DDP} (programming voltage supply). All voltages are measured with respect to V_{SS}.

Terminal Functions (Continued)

TERMINAL NAME	TERMINAL NO.	TYPE†	DESCRIPTION
TEST SIGNALS (CONTINUED)			
$\overline{\text{TRST}}$	32	I	IEEE standard test reset. $\overline{\text{TRST}}$, when active low, 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	38	I/O/Z	Emulator pin 0. When $\overline{\text{TRST}}$ is driven low, EMU0 must be high for activation of the $\overline{\text{OFF}}$ condition. When $\overline{\text{TRST}}$ is driven high, EMU0 is used as an interrupt to or from the emulator system and is defined as input/output through the scan.
EMU1/$\overline{\text{OFF}}$	39	I/O/Z	Emulator pin 1/disable all outputs. When $\overline{\text{TRST}}$ is driven high, EMU1/$\overline{\text{OFF}}$ is used as an interrupt to or from the emulator system and is defined as input/output through JTAG scan. When $\overline{\text{TRST}}$ is driven low, this pin is configured as $\overline{\text{OFF}}$. When EMU1/$\overline{\text{OFF}}$ is active low, it puts all output drivers in the high-impedance state. $\overline{\text{OFF}}$ is used exclusively for testing and emulation purposes (not for multiprocessing applications); therefore, for $\overline{\text{OFF}}$ condition, the following conditions apply: $\overline{\text{TRST}}$ = low, EMU0 = high, EMU1/$\overline{\text{OFF}}$ = low.
RESERVED	42	I	Reserved for test. This pin has an internal pulldown and must be left unconnected for the 'F240.

† I = input, O = output, Z = high impedance

functional block diagram



TEXAS INSTRUMENTS
POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251–1443

device memory map

The SMJ320F240 implements three separate address spaces for program memory, data memory, and I/O. Each space accommodates a total of 64K 16-bit words. Within the 64K words of data space, the 256 to 32K words at the top of the address range can be defined to be external global memory in increments of powers of two, as specified by the contents of the global memory allocation register (GREG). Access to global memory is arbitrated using the global memory bus request ($\overline{BR}$) signal.

On the ’F240, the first 96 (0–5Fh) data memory locations are either allocated for memory-mapped registers or are reserved. This memory-mapped register space contains various control and status registers including those for the CPU.

All the on-chip peripherals of the ’F240 device 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 memory map.

device memory map (continued)

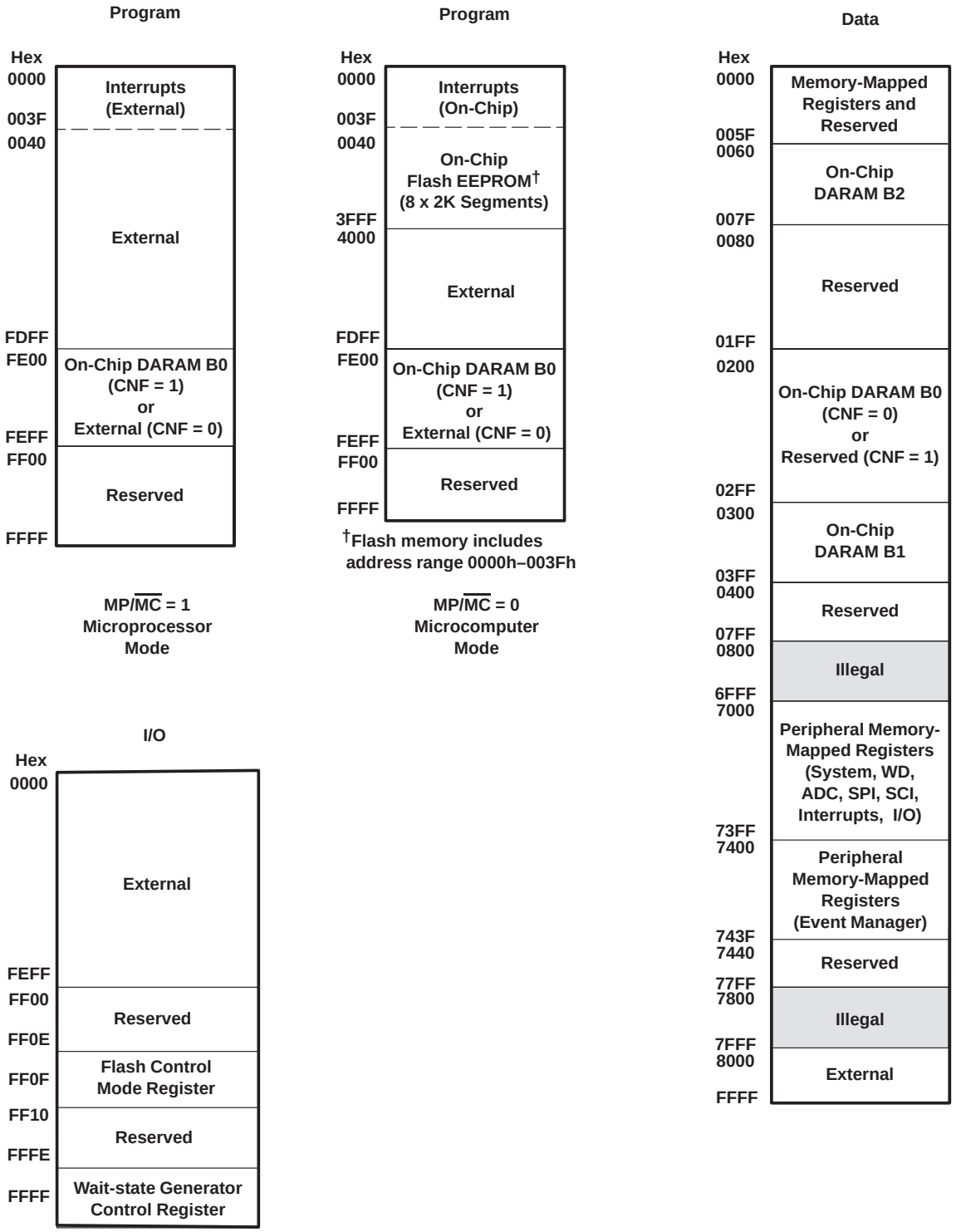


Figure 1. SMJ320F240 Memory Map

peripheral memory map

The SMJ320F240 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).

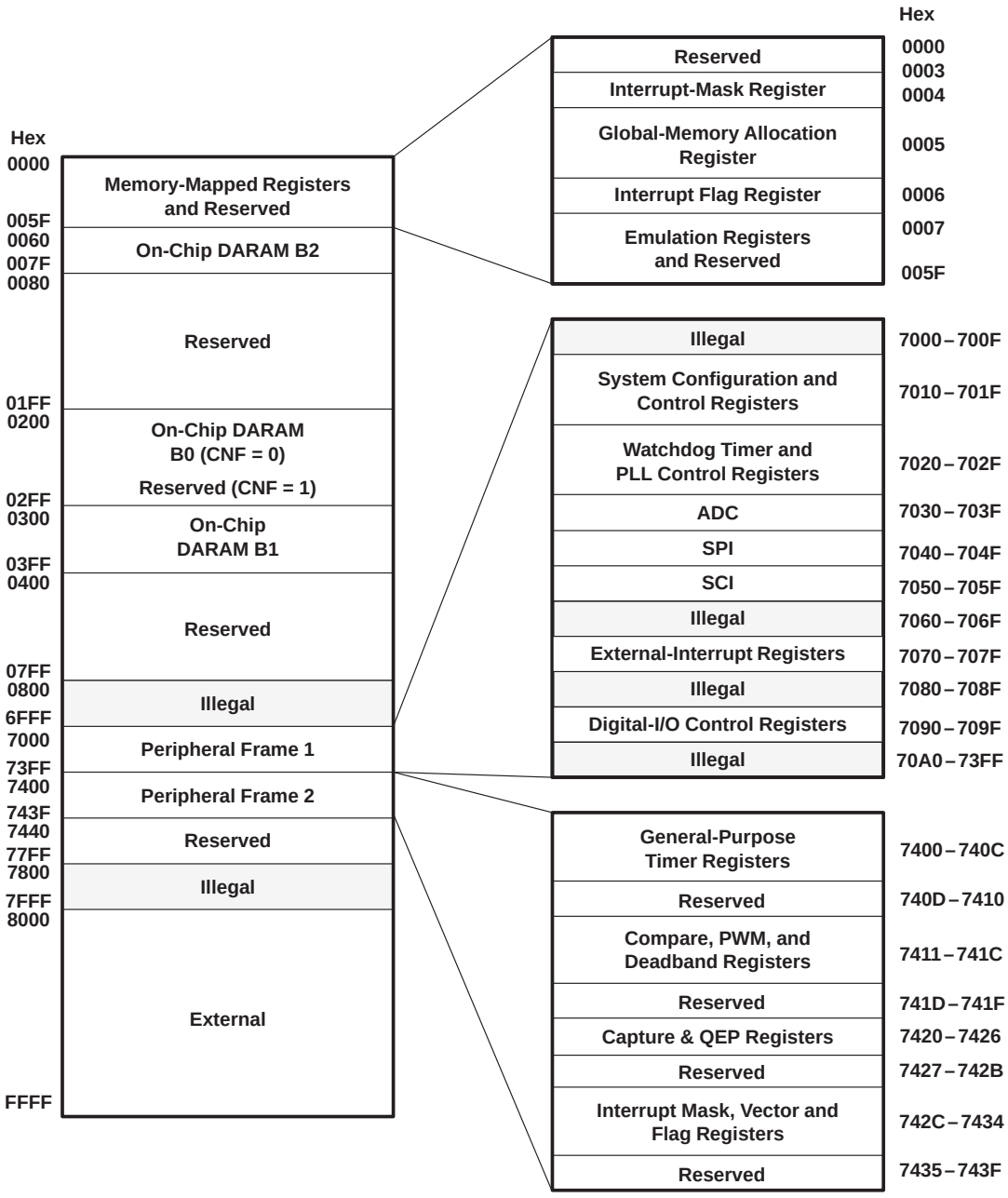


Figure 2. Peripheral Memory Map

digital I/O and shared pin functions

The ’F240 has a total of 28 pins shared between primary functions and I/Os. These pins are divided into two groups:

- Group1 — Primary functions shared with I/Os belonging to dedicated I/O ports, Port A, Port B, and Port C.
- Group2 — Primary functions belonging to peripheral modules which also have a built-in I/O feature as a secondary function (for example, SCI, SPI, external interrupts, and PLL clock modules).

group1 shared I/O pins

The control structure for Group1 type shared I/O pins is shown in Figure 3. The only exception to this configuration is the CLKOUT/IOPC1 pin. In Figure 3, 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.

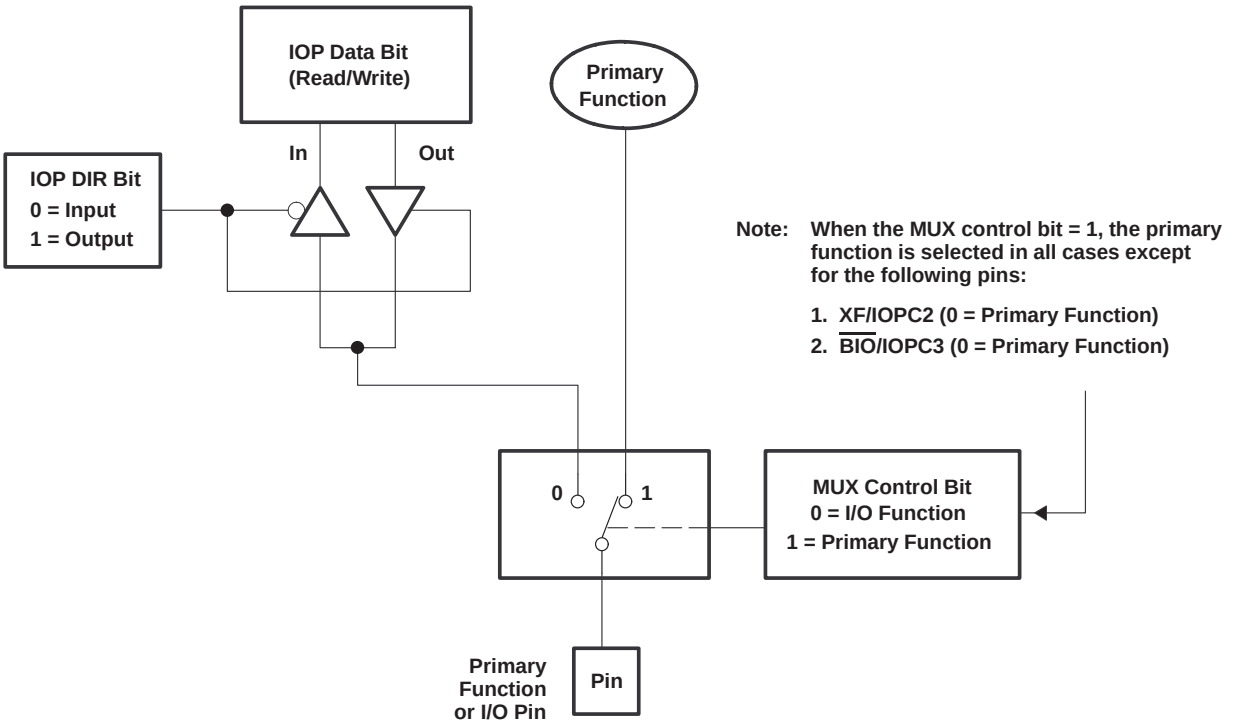


Figure 3. Shared Pin Configuration

A summary of Group1 pin configurations and associated bits is shown in Table 2.

group1 shared I/O pins (continued)

Table 2. Group1 Shared Pin Configurations

PIN #	MUX CONTROL REGISTER (name.bit #)	PIN FUNCTION SELECTED		I/O PORT DATA AND DIRECTION†		
		(CRx.n = 1)	(CRx.n = 0)	REGISTER	DATA BIT #	DIR BIT #
72	OCRA.0	ADCIN0	IOPA0	PADATDIR	0	8
73	OCRA.1	ADCIN1	IOPA1	PADATDIR	1	9
90	OCRA.2	ADCIN9	IOPA2	PADATDIR	2	10
91	OCRA.3	ADCIN8	IOPA3	PADATDIR	3	11
100	OCRA.8	PWM7/CMP7	IOPB0	PBDATDIR	0	8
101	OCRA.9	PWM8/CMP8	IOPB1	PBDATDIR	1	9
102	OCRA.10	PWM9/CMP9	IOPB2	PBDATDIR	2	10
105	OCRA.11	T1PWM/T1CMP	IOPB3	PBDATDIR	3	11
106	OCRA.12	T2PWM/T2CMP	IOPB4	PBDATDIR	4	12
107	OCRA.13	T3PWM/T3CMP	IOPB5	PBDATDIR	5	13
108	OCRA.14	TMRDIR	IOPB6	PBDATDIR	6	14
109	OCRA.15	TMRCLK	IOPB7	PBDATDIR	7	15
63	OCRB.0	ADCSOC	IOPC0	PCDATDIR	0	8
64	SYSCR.7–6					
	0 0	IOPC1		PCDATDIR	1	9
	0 1	WDCLK		—	—	—
	1 0	SYSCLK		—	—	—
	1 1	CPUCLK		—	—	—
65	OCRB.2	IOPC2	XF	PCDATDIR	2	10
66	OCRB.3	IOPC3	$\overline{\text{BIO}}$	PCDATDIR	3	11
67	OCRB.4	CAP1/QEP1	IOPC4	PCDATDIR	4	12
68	OCRB.5	CAP2/QEP2	IOPC5	PCDATDIR	5	13
69	OCRB.6	CAP3	IOPC6	PCDATDIR	6	14
70	OCRB.7	CAP4	IOPC7	PCDATDIR	7	15

† Valid only if the I/O function is selected on the pin.

group2 shared I/O pins

Group2 shared pins belong to peripherals that have built-in general-purpose I/O capability. Control and configuration for these pins are achieved by setting the appropriate bits within the control and configuration registers of the peripherals. Table 3 lists the Group2 shared pins.

Table 3. Group2 Shared Pin Configurations

PIN #	PRIMARY FUNCTION	REGISTER	ADDRESS	PERIPHERAL MODULE
43	SCIRXD	SCIPC2	705Eh	SCI
44	SCITXD	SCIPC2	705Eh	SCI
45	SPISIMO	SPIPC2	704Eh	SPI
48	SPISOMI	SPIPC2	704Eh	SPI
49	SPICLK	SPIPC1	704Dh	SPI
51	SPISTE	SPIPC1	704Dh	SPI
54	XINT2	XINT2CR	7078h	External Interrupts
55	XINT3	XINT3CR	707Ah	External Interrupts

digital I/O control registers

Table 4 lists the registers available to the digital I/O module. As with other ’F240 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

device reset and interrupts

The SMJ320F240 software-programmable interrupt structure supports flexible on-chip and external interrupt configurations to meet real-time interrupt-driven application requirements. The ’F240 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.
- **Hardware-generated interrupts** are requested by external pins or by on-chip peripherals. There are two types:
  - *External interrupts* are generated by one of five external pins corresponding to the interrupts XINT1, XINT2, XINT3, PDPINT, and NMI. The first four 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. NMI, which is not maskable, takes priority over peripheral interrupts and software-generated interrupts. It can be locked out only by an already executing NMI or a reset.
  - *Peripheral interrupts* are initiated internally by these on-chip peripheral modules: the event manager, SPI, SCI, watchdog/real-time interrupt (WD/RTI), 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.

device reset and interrupts (continued)

- **Software-generated interrupts** for the ’F240 device include:
  - *The INTR instruction.* This instruction allows initialization of any ’F240 interrupt with software. Its operand indicates to which interrupt vector location 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 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 five possible causes of a reset, as shown in Figure 4. Three of these causes are internally generated; the other two causes, the $\overline{\text{RS}}$ and $\overline{\text{PORESET}}$ pins, are controlled externally.

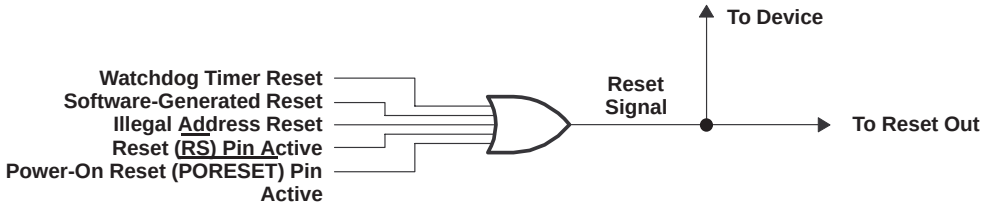


Figure 4. Reset Signals

The five 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.)
- **Software-generated reset.** This is implemented with the system control register (SYSCR). Clearing the RESET0 bit (bit 14) or setting the RESET1 bit (bit 15) causes a system reset.
- **Illegal address reset.** The system and peripheral module control register frame address map contains unimplemented address locations in the ranges labeled illegal. Any access to an address located in the Illegal ranges generate an illegal-address reset.
- **Reset pin active.** To generate an external reset pulse on the $\overline{\text{RS}}$ pin, a low-level pulse duration of as little as a few nanoseconds is usually effective; however, pulses of one SYSCLK cycle are necessary to ensure that the device recognizes the reset signal.
- **Power-on reset pin active.** To generate a power-on reset pulse on the $\overline{\text{PORESET}}$ pin, a low-level pulse of one SYSCLK cycle is necessary to ensure that the device recognizes the reset signal.

Once a reset source is activated, the external $\overline{\text{RS}}$ pin is driven (active) low for a minimum of eight SYSCLK cycles. This allows the SMJ320F240 device to reset external system components. Additionally, if a brown-out condition ($V_{CC} < V_{CC}$min for several microseconds causing $\overline{\text{PORESET}}$ to go low) occurs or the $\overline{\text{RS}}$ pin is held low, then the reset logic holds the device in a reset state for as long as these actions are active.

reset (continued)

The occurrence of a reset condition causes the SMJ320F240 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. In the case of a power-on reset, the PLL control registers are initialized to zero. The program needs to recognize power-on resets and configure the PLL for correct operation.

After a reset, the program can check the power-on reset flag (PORST flag, SYSSR.15), the illegal address flag (ILLADR flag, SYSSR.12), the software reset flag (SWRST flag, SYSSR.10), and the watchdog reset flag (WDRST flag, SYSSR.9) to determine the source of the reset. A reset does not clear these flags.

$\overline{RS}$ and $\overline{PORESET}$ must be held low until the clock signal is valid and V_{CC} is within the operating range. In addition, $\overline{PORESET}$ must be driven low when V_{CC} drops below the minimum operating voltage.

hardware-generated interrupts

All the hardware interrupt lines of the DSP core are given a priority rank from 1 to 10 (1 being highest). When more than one of these hardware interrupts is pending acknowledgment, the interrupt of highest rank gets acknowledged first. The others are acknowledged in order after that. Of those ten lines, six are for maskable interrupt lines (INT1–INT6) and one is for the nonmaskable interrupt (NMI) line. INT1–INT6 and NMI have the priorities shown in Table 5.

Table 5. Interrupt Priorities at the Level of the DSP Core

INTERRUPT	PRIORITY AT THE DSP CORE
RESET	1
TI RESERVED†	2
NMI	3
INT1	4
INT2	5
INT3	6
INT4	7
INT5	8
INT6	9
TI RESERVED†	10

† TI Reserved means that the address space is reserved for Texas Instruments.

The inputs to these lines are controlled by the system module and the event manager as summarized in Table 6 and shown in Figure 5.

Table 6. Interrupt Lines Controlled by the System Module and Event Manager

PERIPHERAL	INTERRUPT LINES
System Module	INT1 INT5 NMI INT6
Event Manager	INT2 INT3 INT4

hardware-generated interrupts (continued)

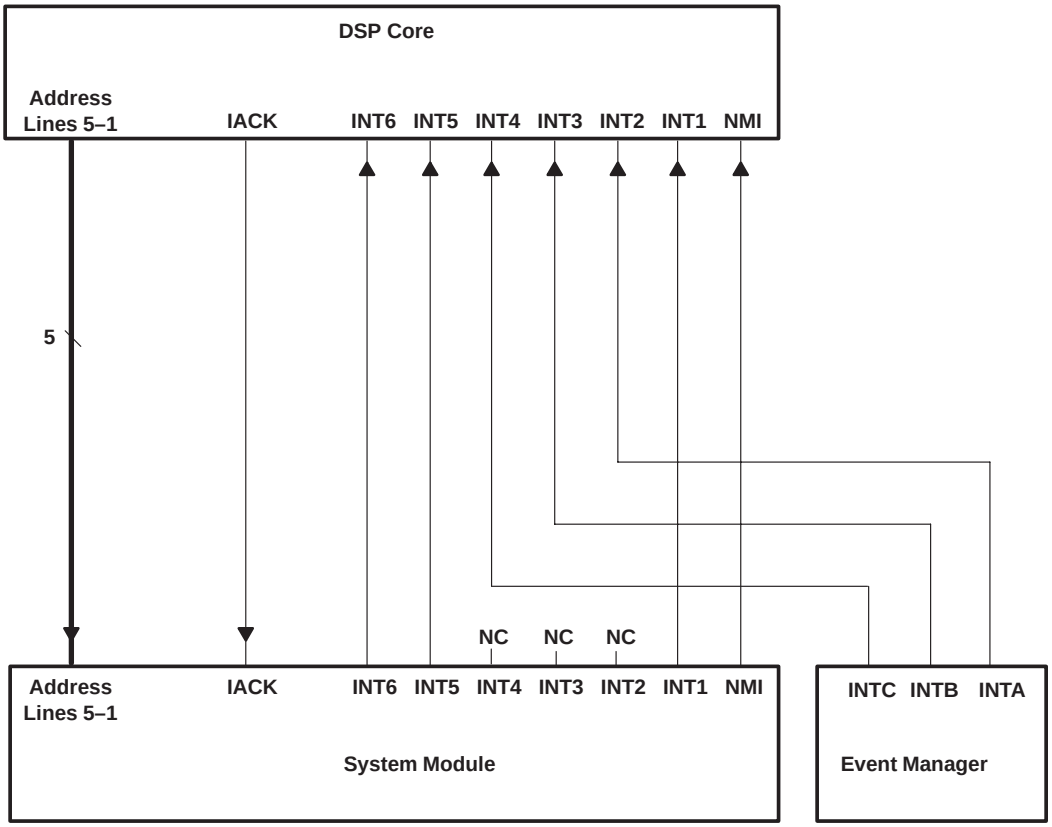


Figure 5. DSP Interrupt Structure

At the level of the system module and the event manager, each of the maskable interrupt lines (INT1–INT6) is connected to multiple maskable interrupt sources. Sources connected to interrupt line INT1 are called Level 1 interrupts; sources connected to interrupt line INT2 are called Level 2 interrupts; and so on. For each interrupt line, the multiple sources also have a set priority ranking. The source with the highest priority has its interrupt request responded to by the DSP core first.

Figure 6 shows the sources and priority ranking for the interrupts controlled by the system module. For each interrupt chain, the interrupt source of highest priority is at the top. Priority decreases from the top of the chain to the bottom. Figure 7 shows the interrupt sources and priority ranking for the event manager interrupts.

hardware-generated interrupts (continued)

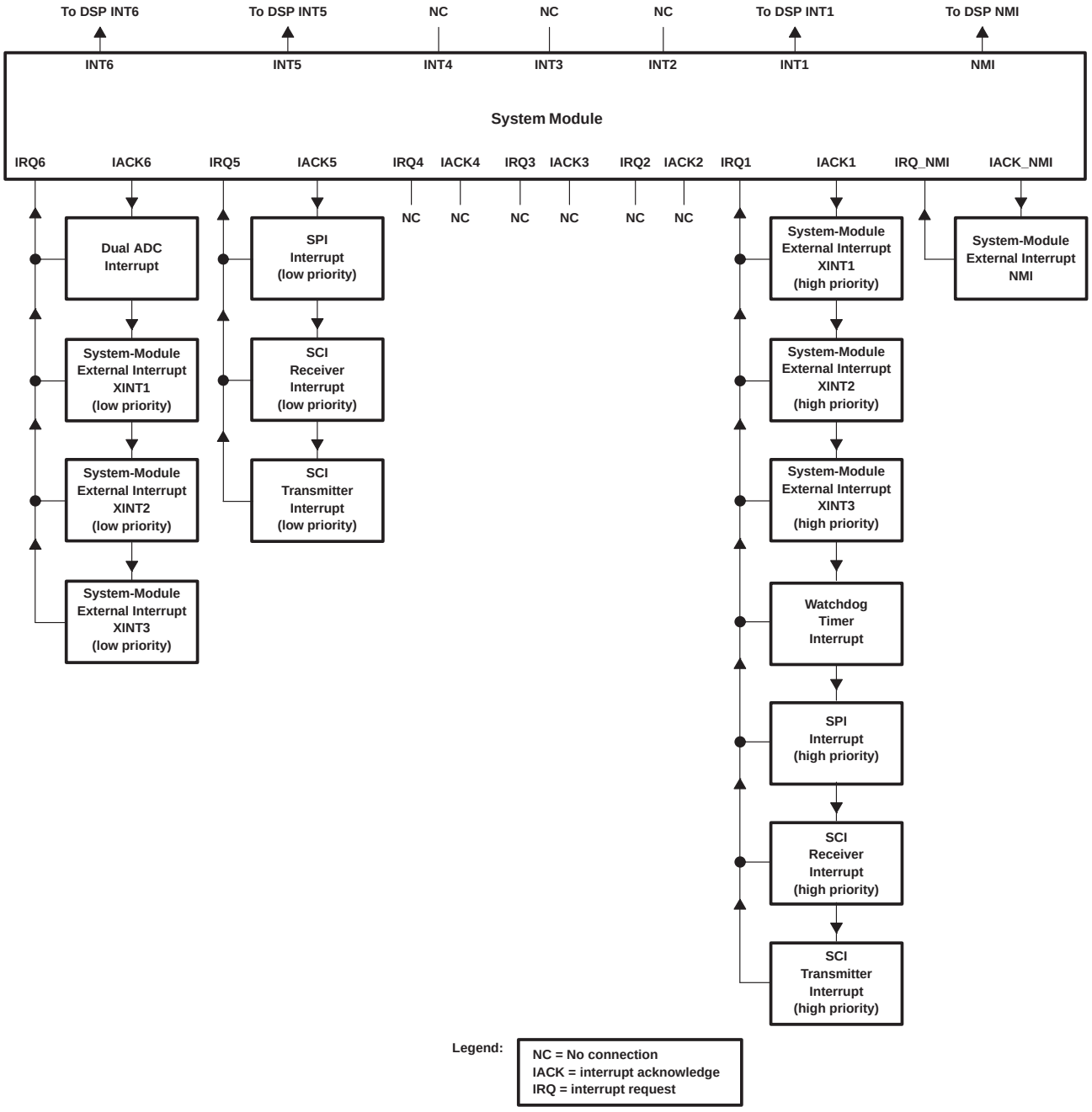


Figure 6. System-Module Interrupt Structure

hardware-generated interrupts (continued)

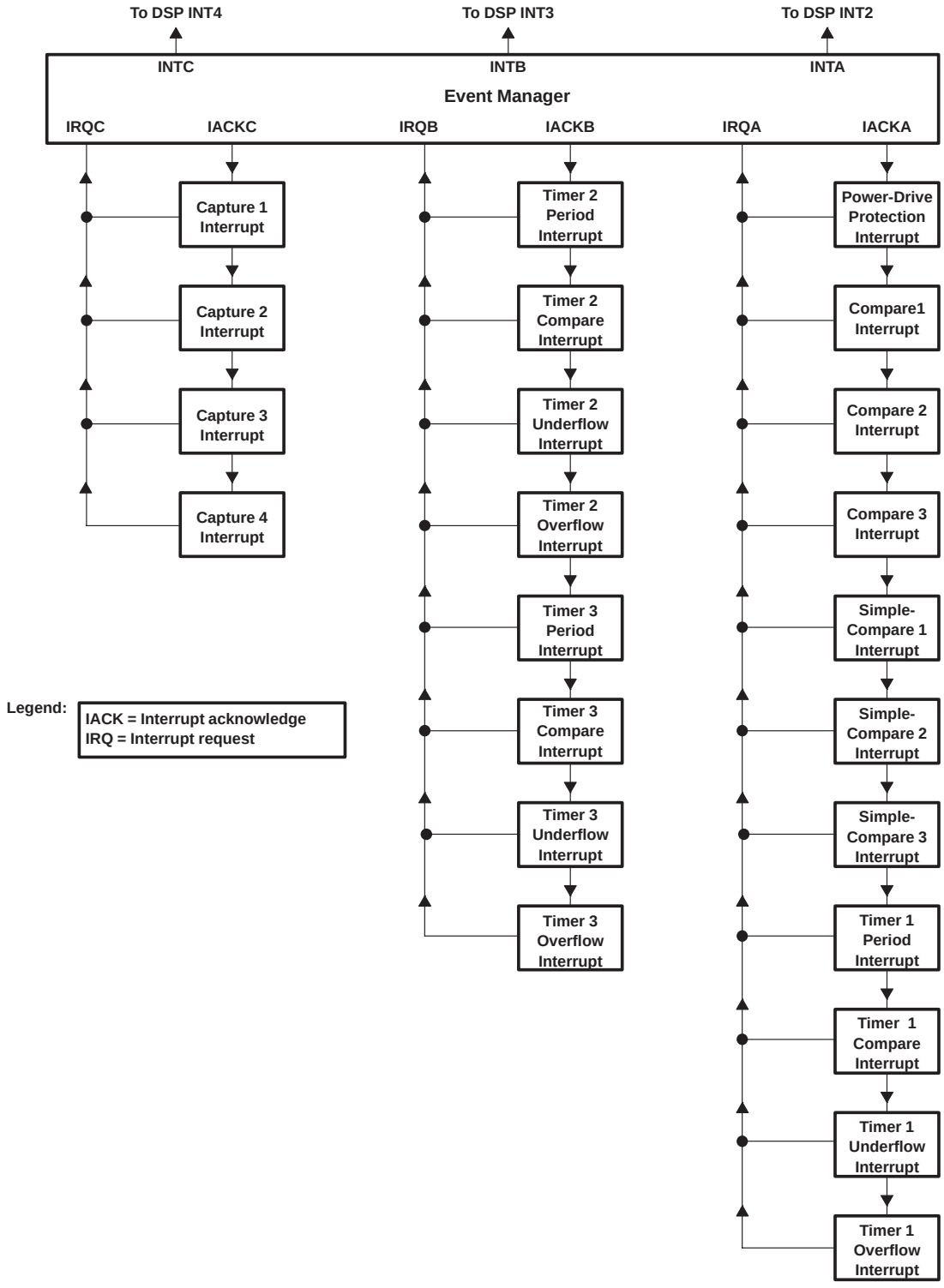


Figure 7. Event-Manager Interrupt Structure

hardware-generated interrupts (continued)

Each of the interrupt sources has its own control register with a flag bit and an enable bit. When an interrupt request is received, the flag bit in the corresponding control register is set. If the enable bit is also set, a signal is sent to arbitration logic, which can simultaneously receive similar signals from one or more of the other control registers. The arbitration logic compares the priority level of competing interrupt requests, and it passes the interrupt of highest priority to the CPU. The corresponding flag is set in the interrupt flag register (IFR), indicating that the interrupt is pending. The CPU then must decide whether to acknowledge the request. Maskable hardware interrupts are acknowledged only after certain conditions are met:

- **Priority is highest.** When more than one hardware interrupt is requested at the same time, the ’F240 services them according to the set priority ranking.
- **INTM bit is 0.** The interrupt mode (INTM) bit, bit 9 of status register ST0, enables or disables all maskable interrupts:
  - When INTM = 0, all unmasked interrupts are enabled.
  - When INTM = 1, all unmasked interrupts are disabled.

  INTM is set to 1 automatically when the CPU acknowledges an interrupt (except when initiated by the TRAP instruction) and at reset. It can be set and cleared by software.
- **IMR mask bit is 1.** Each of the maskable interrupt lines has a mask bit in the interrupt mask register (IMR). To unmask an interrupt line, set its IMR bit to 1.

When the CPU acknowledges a maskable hardware interrupt, it jams the instruction bus with the INTR instruction. This instruction forces the PC to the appropriate address from which the CPU fetches the software vector. This vector leads to an interrupt service routine.

Usually, the interrupt service routine reads the peripheral-vector-address offset from the peripheral-vector-address register (see Table 7) to branch to code that is meant for the specific interrupt source that initiated the interrupt request. The ’F240 includes a phantom-interrupt vector offset (0000h), which is a system interrupt integrity feature that allows a controlled exit from an improper interrupt sequence. If the CPU acknowledges a request from a peripheral when, in fact, no peripheral has requested an interrupt, the phantom-interrupt vector is read from the interrupt-vector register.

Table 7 summarizes the interrupt sources, overall priority, vector address/offset, source, and function of each interrupt available on the SMJ320F240.

hardware-generated interrupts (continued)

Table 7. ’F240 Interrupt Locations and Priorities

INTERRUPT NAME	OVERALL PRIORITY	DSP-CORE INTERRUPT, AND ADDRESS	PERIPHERAL VECTOR ADDRESS	PERIPHERAL VECTOR ADDRESS OFFSET	MASKABLE?	’F240 SOURCE PERIPHERAL MODULE	FUNCTION INTERRUPT
$\overline{\text{RS}}$	1 Highest	$\overline{\text{RS}}$ 0000h	N/A		N	Core, SD	External, system reset (RESET)
RESERVED	2	INT7 0026h	N/A	N/A	N	DSP Core	Emulator trap
NMI	3	NMI 0024h	N/A	0002h	N	Core, SD	External user interrupt
XINT1 XINT2 XINT3	4 5 6	INT1 0002h (System)	SYSIVR (701Eh)	0001h 0011h 001Fh	Y	SD	High-priority external user interrupts
SPIINT	7			0005h	Y	SPI	High-priority SPI interrupt
RXINT	8			0006h	Y	SCI	SCI receiver interrupt (high priority)
TXINT	9			0007h	Y	SCI	SCI transmitter interrupt (high priority)
WDTINT	10			0010h	Y	WDT	Watchdog timer interrupt
PDPINT	11	INT2 0004h (Event Manager Group A)	EVIVRA (7432h)	0020h	Y	External	Power-drive protection Int.
CMP1INT	12			0021h	Y	EV.CMP1	Full Compare 1 interrupt
CMP2INT	13			0022h	Y	EV.CMP2	Full Compare 2 interrupt
CMP3INT	14			0023h	Y	EV.CMP3	Full Compare 3 interrupt
SCMP1INT	15			0024h	Y	EV.CMP4	Simple compare 1 interrupt
SCMP2INT	16			0025h	Y	EV.CMP5	Simple compare 2 interrupt
SCMP3INT	17			0026h	Y	EV.CMP6	Simple compare 3 interrupt
TPINT1	18			0027h	Y	EV.GPT1	Timer1-period interrupt
TCINT1	19			0028h	Y	EV.GPT1	Timer1-compare interrupt
TUFINT1	20			0029h	Y	EV.GPT1	Timer1-underflow interrupt
TOFINT1	21			002Ah	Y	EV.GPT1	Timer1-overflow interrupt
TPINT2	22	INT3 0006h (Event Manager Group B)	EVIVRB (7433h)	002Bh	Y	EV.GPT2	Timer2-period interrupt
TCINT2	23			002Ch	Y	EV.GPT2	Timer2-compare interrupt
TUFINT2	24			002Dh	Y	EV.GPT2	Timer2-underflow interrupt
TOFINT2	25			002Eh	Y	EV.GPT2	Timer2-overflow interrupt
TPINT3	26			002Fh	Y	EV.GPT3	Timer3-period interrupt
TCINT3	27			0030h	Y	EV.GPT3	Timer3-compare interrupt
TUFINT3	28			0031h	Y	EV.GPT3	Timer3-underflow interrupt
TOFINT3	29			0032h	Y	EV.GPT3	Timer3-overflow interrupt
CAPINT1	30	INT4 0008h (Event Manager Group C)	EVIVRC (7434h)	0033h	Y	EV.CAP1	Capture 1 interrupt
CAPINT2	31			0034h	Y	EV.CAP2	Capture 2 interrupt
CAPINT3	32			0035h	Y	EV.CAP3	Capture 3 interrupt
CAPINT4	33			0036h	Y	EV.CAP4	Capture 4 interrupt

hardware-generated interrupts (continued)

Table 7. ’F240 Interrupt Locations and Priorities (Continued)

INTERRUPT NAME	OVERALL PRIORITY	DSP-CORE INTERRUPT, AND ADDRESS	PERIPHERAL VECTOR ADDRESS	PERIPHERAL VECTOR ADDRESS OFFSET	MASKABLE?	’F240 SOURCE PERIPHERAL MODULE	FUNCTION INTERRUPT
SPIINT	34	INT5	SYSIVR (701Eh)	0005h	Y	SPI	Low-priority SPI interrupt
RXINT	35	000Ah (System)		0006h	Y	SCI	SCI receiver interrupt (low priority)
TXINT	36			0007h	Y	SCI	SCI transmitter interrupt (low priority)
ADCINT	37	INT6	SYSIVR	0004h	Y	ADC	Analog-to-digital interrupt
XINT1 XINT2 XINT3	38 39 40	000Ch (System)	(701Eh)	0001h 0011h 001Fh	Y Y Y	External pins	Low-priority external user interrupts
RESERVED	41	000Eh	N/A		Y	DSP Core	Used for analysis
TRAP	N/A	0022h	N/A		N/A		TRAP instruction vector

external interrupts

The ’F240 has five external interrupts. These interrupts include:

- **XINT1.** Type A interrupt. 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 input pin. XINT1 can also be programmed to trigger an interrupt on either the rising or the falling edge.
- **NMI.** Type A interrupt. The NMI control register (at 7072h) provides control and status for this interrupt. NMI is a nonmaskable external interrupt or a general-purpose input pin. NMI can also be programmed to trigger an interrupt on either the rising or the falling edge.
- **XINT2.** Type C interrupt. The XINT2 control register (at 7078h) 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.
- **XINT3.** Type C interrupt. The XINT3 control register (at 707Ah) provides control and status for this interrupt. XINT3 can be used as a high-priority (Level 1) or low-priority (Level 6) maskable interrupt or as a general-purpose I/O pin. XINT3 can also be programmed to trigger an interrupt on either the rising or the falling edge.
- **PDPINT.** This interrupt is provided for safe operation of the power converter and motor drive. This maskable interrupt can put the timers and PWM output pins in the high-impedance state and inform the CPU in case of motor drive abnormalities such as overvoltage, overcurrent, and excessive temperature rise. PDPINT is a Level 2 interrupt.

external interrupts (continued)

Table 8 is a summary of the external interrupt capability of the ’F240.

Table 8. External Interrupt Types and Functions

EXTERNAL INTERRUPT	CONTROL REGISTER NAME	CONTROL REGISTER ADDRESS	INTERRUPT TYPE	CAN DO NMI?	DIGITAL I/O PIN	MASKABLE?
XINT1	XINT1CR	7070h	A	No	Input only	Yes (Level 1 or 6)
NMI	NMICR	7072h	A	Yes	Input only	No
XINT2	XINT2CR	7078h	C	No	I/O	Yes (Level 1 or 6)
XINT3	XINT3CR	707Ah	C	No	I/O	Yes (Level 1 or 6)
PDPINT	EVIMRA	742Ch	N/A	N/A	N/A	Yes (Level 2)

clock generation

The SMJ320F240 has an on-chip, 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 an external fundamental crystal, or oscillator.

The PLL-based clock module provides two basic modes of operation: oscillator mode and clock-in mode.

- oscillator mode
  This mode allows the use of a 4-, 6-, or 8-MHz external reference crystal to provide the time base to the device. The internal oscillator circuitry is initialized by software to select the desired CPUCLK frequency, which can be the input clock frequency, the input clock frequency divided by 2 (default), or a clock frequency determined by the PLL.
- Clock-in mode
  This mode allows the internal crystal oscillator circuitry to be bypassed. The device clocks are generated from an external clock source input on the XTAL1/CLKIN pin. The device can be configured by software to operate on the input clock frequency, the input clock frequency divided by 2, or a clock frequency determined by the PLL.

The ’F240 runs on two clock frequencies: the CPU clock (CPUCLK) frequency, and the system clock (SYSCLK) frequency. The CPU, memories, external memory interface, and event manager run at the CPUCLK frequency. All other peripherals run at the SYSCLK frequency. The CPUCLK runs at 2x or 4x the frequency of the SYSCLK; for example, for 2x, CPUCLK = 20 MHz and SYSCLK = 10 MHz. There is also a clock for the watchdog timer, WDCLK. This clock has a nominal frequency of 16384 Hz (214 Hz) when XTAL1/CLKIN is a power of two or a sum of two powers of two; for example, 4194304 Hz (222 Hz), 6291456 (222 + 221 Hz), or 8388608 Hz (223 Hz).

The clock module includes three external pins:

1. XTAL1/CLKIN clock source/crystal input
2. XTAL2 output to crystal
3. $\overline{\text{OSCBYP}}$ oscillator bypass

For the external pins, if $\overline{\text{OSCBYP}} \geq V_{IH}$, then the oscillator is enabled and if $\overline{\text{OSCBYP}} \leq V_{IL}$, then the oscillator is bypassed and the device is in clock-in mode. In clock-in mode, an external TTL clock must be applied to the XTAL1/CLKIN pin. The XTAL2 pin can be left unconnected.

clock generation (continued)

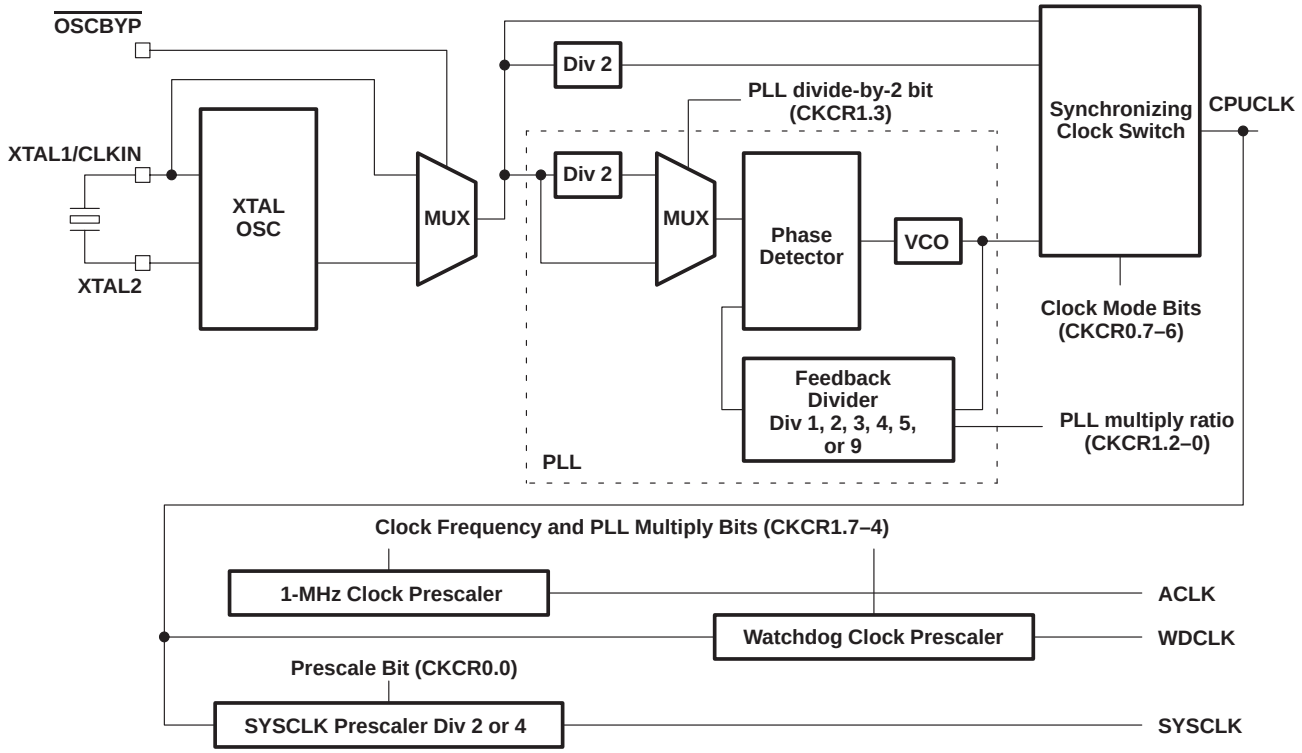


Figure 8. PLL Clock Module Block Diagram

low-power modes

The SMJ320F240 has four low-power modes (idle 1, idle 2, PLL power down, and oscillator power down). The low-power modes reduce the operating power by reducing or stopping the activity of various modules (by stopping their clocks). The two PLLPM bits of the clock module control register, CKCR0, select which of the low-power modes the device enters when executing an IDLE instruction. Reset or an unmasked interrupt from any source causes the device to exit from idle 1 low-power mode. A real-time interrupt from the watchdog timer module causes the device to exit from all low-power modes except oscillator power down. This is a wake-up interrupt. When enabled, reset or any of the four external interrupts (NMI, XINT1, XINT2, or XINT3) causes the device to exit from any of the low-power modes (idle 1, idle 2, PLL power down, and oscillator power down). The external interrupts are all wake-up interrupts. The maskable external interrupts (XINT1, XINT2, and XINT3) must be enabled individually and globally to bring the device out of a low-power mode properly. It is, therefore, important to ensure that the desired low-power-mode exit path is enabled before entering a low-power mode. Figure 9 shows the wake-up sequence from a power down. Table 9 summarizes the low-power modes.

low-power modes (continued)

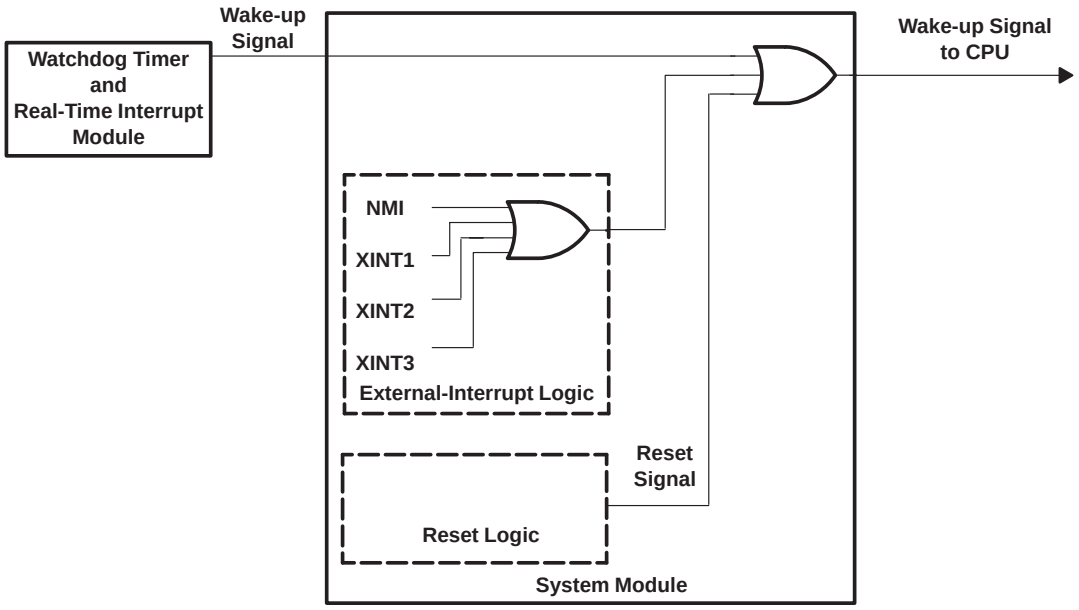


Figure 9. Waking Up the Device From Power Down

Table 9. Low-Power Modes

LOW-POWER MODE	PLLPM(x) BITS IN CKCR0[2:3]	CPUCLK STATUS	SYSCLK STATUS	WDCLK STATUS	PLL STATUS	OSC STATUS	EXIT CONDITION	TYPICAL POWER
Run	XX	On	On	On	On	On	—	80 mA
Idle 1	00	Off	On	On	On	On	Any interrupt or reset	50 mA
Idle 2	01	Off	Off	On	On	On	Wake-up interrupt or reset	7 mA
PLL Power Down	10	Off	Off	On	Off	On	Wake-up interrupt or reset	1 mA
OSC Power Down	11	Off	Off	Off	Off	Off	Wake-up interrupt or reset	400 μA

low-power modes (continued)

Table 10. Legend for the ’F240 Internal Hardware Functional Block Diagram

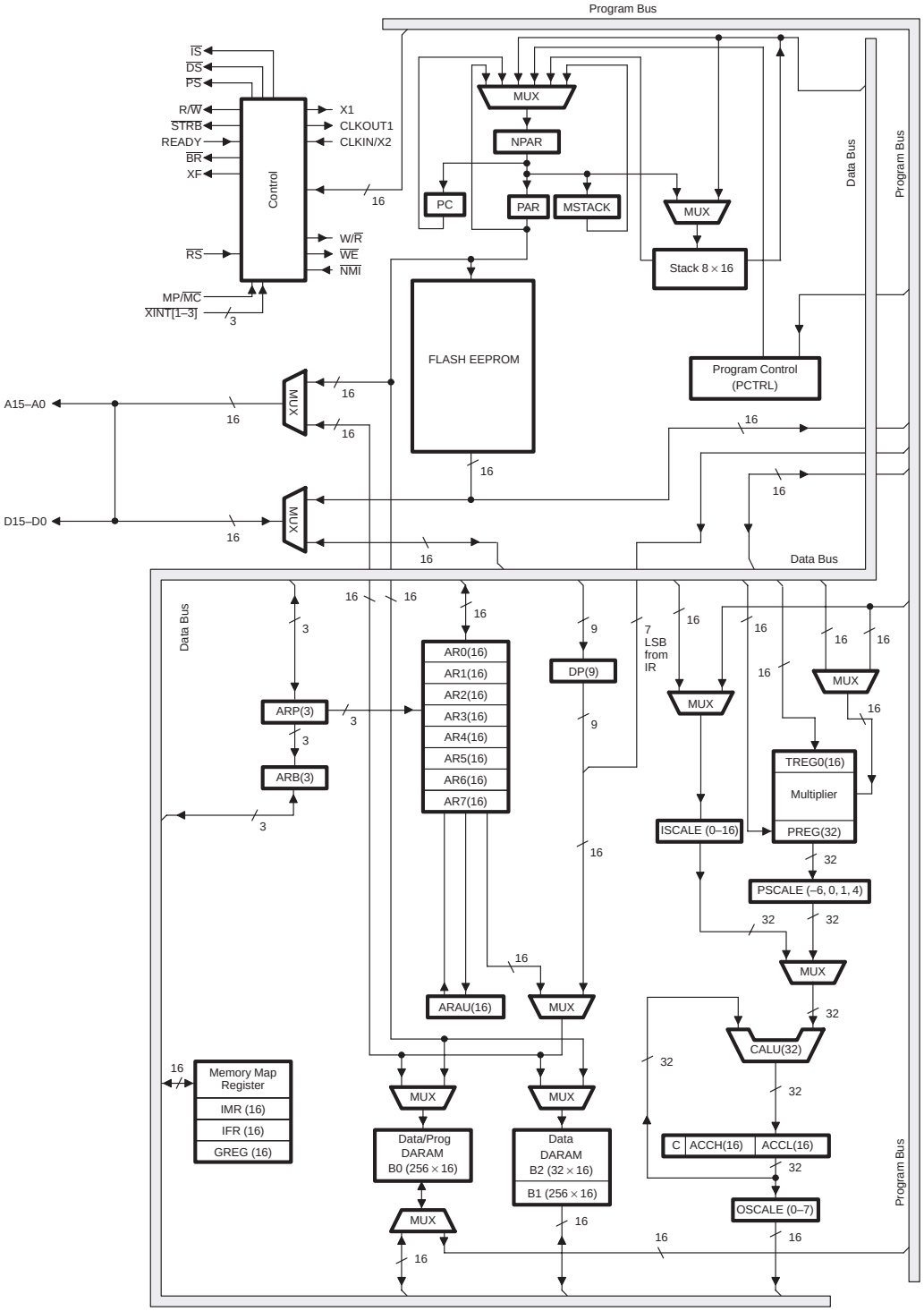
SYMBOL	NAME	DESCRIPTION
ACC	Accumulator	32-bit register that stores the results and provides input for subsequent CALU operations. Also includes shift and rotate capabilities
ARAU	Auxiliary Register Arithmetic Unit	An unsigned, 16-bit arithmetic unit used to calculate indirect addresses using the auxiliary 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 SMJ320C2xx 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 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.
IMR	Interrupt Mask Register	IMR individually masks or enables the seven interrupts.
IFR	Interrupt Flag Register	The 7-bit IFR indicates that the SMJ320C2xx 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.

low-power modes (continued)

Table 10. Legend for the ’F240 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 ’C20x 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.

functional block diagram of the SMJ320F240 DSP CPU



NOTES: A. Symbol descriptions appear in Table 10.
B. For clarity the data and program buses are shown as single buses although they include address and data bits.

’F240 DSP core CPU

The SMJ320F240 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 that are ROM. This, coupled with a four-deep pipeline, allows the ’F240 devices to execute most instructions in a single cycle.

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, thereby 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 10 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 11 lists status register field definitions.

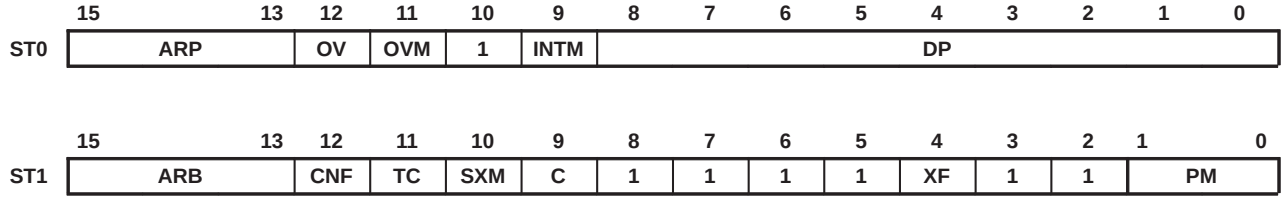


Figure 10. Status and Control Register Organization

Table 11. 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.0
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. $\overline{RS}$ 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. $\overline{RS}$ and $\overline{IACK}$ also set INTM. INTM has no effect on the unmaskable $\overline{RS}$ 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.

status and control registers (continued)

Table 11. Status Register Field Definitions (Continued)

FIELD	FUNCTION
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 $\overline{RS}$.
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. 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 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.

central processing unit

The SMJ320F240 central processing unit (CPU) contains a 16-bit scaling shifter, a 16 x 16-bit parallel multiplier, a 32-bit central arithmetic logic unit (CALU), a 32-bit accumulator, and additional shifters at the outputs of both the accumulator and the multiplier. This section describes the CPU components and their functions. The functional block diagram shows the components of the CPU.

input scaling shifter

The SMJ320F240 provides a scaling shifter with a 16-bit input connected to the data bus and a 32-bit output connected to the CALU. This shifter operates as part of the path of data coming from program or data space to the CALU and requires no cycle overhead. It is used to align the 16-bit data coming from memory to the 32-bit CALU. This is necessary for scaling arithmetic as well as aligning masks for logical operations.

The scaling shifter produces a left shift of 0 to 16 on the input data. The LSBs of the output are filled with zeros; the MSBs can either be filled with zeros or sign-extended, depending upon the value of the SXM bit (sign-extension mode) of status register ST1. The shift count is specified by a constant embedded in the instruction word or by a value in TREG. The shift count in the instruction allows for specific scaling or alignment operations specific to that point in the code. The TREG base shift allows the scaling factor to be adaptable to the system's performance.

multiplier

The SMJ320F240 devices use a 16 x 16-bit hardware multiplier that is capable of computing a signed or an unsigned 32-bit product in a single machine cycle. All multiply instructions, except the MPYU (multiply unsigned) instruction, perform a signed multiply operation. That is, two numbers being multiplied are treated as 2s-complement numbers, and the result is a 32-bit 2s-complement number. Two registers are associated with the multiplier:

- 16-bit temporary register (TREG) that holds one of the operands for the multiplier
- 32-bit product register (PREG) that holds the product

multiplier (continued)

Four product shift modes (PM) are available at the PREG output (PSCALE). These shift modes are useful for performing multiply/accumulate operations, performing fractional arithmetic, or justifying fractional products. The PM field of status register ST1 specifies the PM shift mode, as shown in Table 12.

Table 12. PSCALE Product Shift Modes

PM	SHIFT	DESCRIPTION
00	No shift	Product feed to CALU or data bus with no shift
01	Left 1	Removes the extra sign bit generated in a 2s-complement multiply to produce a Q31 product
10	Left 4	Removes the extra 4 sign bits generated in a 16x13 2s-complement multiply to a produce a Q31 product when using the multiply by a 13-bit constant
11	Right 6	Scales the product to allow up to 128 product accumulation without the possibility of accumulator overflow

The product can be shifted one bit to compensate for the extra sign bit gained in multiplying two 16-bit 2s-complement numbers (MPY instruction). A four-bit shift is used in conjunction with the MPY instruction with a short immediate value (13 bits or less) to eliminate the four extra sign bits gained in multiplying a 16-bit number by a 13-bit number. Finally, the output of PREG can be right-shifted 6 bits to enable the execution of up to 128 consecutive multiply/accumulates without the possibility of overflow.

The LT (load TREG) instruction normally loads TREG to provide one operand (from the data bus), and the MPY (multiply) instruction provides the second operand (also from the data bus). A multiplication also can be performed with a 13-bit immediate operand when using the MPY instruction. Then a product is obtained every two cycles. When the code is executing multiple multiplies and product sums, the CPU supports the pipelining of the TREG load operations with CALU operations using the previous product. The pipeline operations that run in parallel with loading the TREG include: load ACC with PREG (LTP); add PREG to ACC (LTA); add PREG to ACC and shift TREG input data (DMOV) to next address in data memory (LTD); and subtract PREG from ACC (LTS).

Two multiply/accumulate instructions (MAC and MACD) fully utilize the computational bandwidth of the multiplier, allowing both operands to be processed simultaneously. The data for these operations can be transferred to the multiplier each cycle by way of the program and data buses. This facilitates single-cycle multiply/accumulates when used with the repeat (RPT) instruction. In these instructions, the coefficient addresses are generated by program address generation (PAGEN) logic, while the data addresses are generated by data address generation (DAGEN) logic. This allows the repeated instruction to access the values from the coefficient table sequentially and step through the data in any of the indirect addressing modes.

The MACD instruction, when repeated, supports filter constructs (weighted running averages) so that as the sum-of-products is executed, the sample data is shifted in memory to make room for the next sample and to throw away the oldest sample.

The MPYU instruction performs an unsigned multiplication, which greatly facilitates extended-precision arithmetic operations. The unsigned contents of TREG are multiplied by the unsigned contents of the addressed data memory location, with the result placed in PREG. This process allows the operands of greater than 16 bits to be broken down into 16-bit words and processed separately to generate products of greater than 32 bits. The SQRA (square/add) and SQRS (square/subtract) instructions pass the same value to both inputs of the multiplier for squaring a data memory value.

After the multiplication of two 16-bit numbers, the 32-bit product is loaded into the 32-bit product register (PREG). The product from PREG can be transferred to the CALU or to data memory by way of the SPH (store product high) and SPL (store product low). Note: the transfer of PREG to either the CALU or data bus passes through the PSCALE shifter, and therefore is affected by the product shift mode defined by PM. This is important when saving PREG in an interrupt-service-routine context save as the PSCALE shift effects cannot be modeled

multiplier (continued)

in the restore operation. PREG can be cleared by executing the MPY #0 instruction. The product register can be restored by loading the saved low half into TREG and executing a MPY #1 instruction. The high half, then, is loaded using the LPH instruction.

central arithmetic logic unit

The SMJ320F240 central arithmetic logic unit (CALU) implements a wide range of arithmetic and logical functions, the majority of which execute in a single clock cycle. This ALU is referred to as central to differentiate it from a second ALU used for indirect-address generation called the auxiliary register arithmetic unit (ARAU). Once an operation is performed in the CALU, the result is transferred to the accumulator (ACC) where additional operations, such as shifting, can occur. Data that is input to the CALU can be scaled by ISCALE when coming from one of the data buses (DRDB or PRDB) or scaled by PSCALE when coming from the multiplier.

The CALU is a general-purpose arithmetic/logic unit that operates on 16-bit words taken from data memory or derived from immediate instructions. In addition to the usual arithmetic instructions, the CALU can perform Boolean operations, facilitating the bit manipulation ability required for a high-speed controller. One input to the CALU is always provided from the accumulator, and the other input can be provided from the product register (PREG) of the multiplier or the output of the scaling shifter (that has been read from data memory or from the ACC). After the CALU has performed the arithmetic or logical operation, the result is stored in the accumulator.

The SMJ320F240 devices support floating-point operations for applications requiring a large dynamic range. The NORM (normalization) instruction is used to normalize fixed-point numbers contained in the accumulator by performing left shifts. The four bits of the TREG define a variable shift through the scaling shifter for the LACT/ADDT/SUBT (load/add to /subtract from accumulator with shift specified by TREG) instructions. These instructions are useful in floating-point arithmetic where a number needs to be denormalized — that is floating-point to fixed-point conversion. They are also useful in execution of an automatic gain control (AGC) going into a filter. The BITT (bit test) instruction provides testing of a single bit of a word in data memory based on the value contained in the four LSBs of TREG.

The CALU overflow saturation mode can be enabled/disabled by setting/resetting the OVM bit of ST0. When the CALU is in the overflow saturation mode and an overflow occurs, the overflow flag is set and the accumulator is loaded with either the most positive or the most negative value representable in the accumulator, depending on the direction of the overflow. The value of the accumulator at saturation is 07FFFFFFh (positive) or 080000000h (negative). If the OVM (overflow mode) status register bit is reset and an overflow occurs, the overflowed results are loaded into the accumulator with modification. (Logical operations cannot result in overflow.)

The CALU can execute a variety of branch instructions that depend on the status of the CALU and the accumulator. These instructions can be executed conditionally based on any meaningful combination of these status bits. For overflow management, these conditions include the OV (branch on overflow) and EQ (branch on accumulator equal to zero). In addition, the BACC (branch to address in accumulator) instruction provides the ability to branch to an address specified by the accumulator (computed goto). Bit test instructions (BIT and BITT), which do not affect the accumulator, allow the testing of a specified bit of a word in data memory.

The CALU also has an associated carry bit that is set or reset depending on various operations within the device. The carry bit allows more efficient computation of extended-precision products and additions or subtractions. It also is useful in overflow management. The carry bit is affected by most arithmetic instructions as well as the single-bit shift and rotate instructions. It is not affected by loading the accumulator, logical operations, or other such non-arithmetic or control instructions.

The ADDC (add to accumulator with carry) and SUBB (subtract from accumulator with borrow) instructions use the previous value of carry in their addition/subtraction operation.

central arithmetic logic unit (continued)

The one exception to the operation of the carry bit is in the use of ADD with a shift count of 16 (add to high accumulator) and SUB with a shift count of 16 (subtract from high accumulator) instructions. This case of the ADD instruction can set the carry bit only if a carry is generated, and this case of the SUB instruction can reset the carry bit only if a borrow is generated; otherwise, neither instruction affects it.

Two conditional operands, C and NC, are provided for branching, calling, returning, and conditionally executing, based upon the status of the carry bit. The SETC, CLRC, and LST #1 instructions also can be used to load the carry bit. The carry bit is set to one on a hardware reset.

accumulator

The 32-bit accumulator is the registered output of the CALU. It can be split into two 16-bit segments for storage in data memory. Shifters at the output of the accumulator provide a left shift of 0 to 7 places. This shift is performed while the data is being transferred to the data bus for storage. The contents of the accumulator remain unchanged. When the post-scaling shifter is used on the high word of the accumulator (bits 16–31), the MSBs are lost and the LSBs are filled with bits shifted in from the low word (bits 0–15). When the post-scaling shifter is used on the low word, the LSBs are zero-filled.

The SFL and SFR (in-place one-bit shift to the left/right) instructions and the ROL and ROR (rotate to the left/right) instructions implement shifting or rotating of the contents of the accumulator through the carry bit. The SXM bit affects the definition of the SFR (shift accumulator right) instruction. When SXM = 1, SFR performs an arithmetic right shift, maintaining the sign of the accumulator data. When SXM = 0, SFR performs a logical shift, shifting out the LSBs and shifting in a zero for the MSB. The SFL (shift accumulator left) instruction is not affected by the SXM bit and behaves the same in both cases, shifting out the MSB and shifting in a zero. Repeat (RPT) instructions can be used with the shift and rotate instructions for multiple-bit shifts.

auxiliary registers and auxiliary-register arithmetic unit (ARAU)

The ’F240 provides a register file containing eight auxiliary registers (AR0–AR7). The auxiliary registers are used for indirect addressing of the data memory or for temporary data storage. Indirect auxiliary-register addressing allows placement of the data memory address of an instruction operand into one of the auxiliary registers. These registers are referenced with a 3-bit auxiliary register pointer (ARP) that is loaded with a value from 0 through 7, designating AR0 through AR7, respectively. The auxiliary registers and the ARP can be loaded from data memory, the ACC, the product register, or by an immediate operand defined in the instruction. The contents of these registers also can be stored in data memory or used as inputs to the CALU.

The auxiliary register file (AR0–AR7) is connected to the ARAU. The ARAU can autoindex the current auxiliary register while the data memory location is being addressed. Indexing either by ±1 or by the contents of the AR0 register can be performed. As a result, accessing tables of information does not require the CALU for address manipulation; therefore, the CALU is free for other operations in parallel.

internal memory

The SMJ320F240 devices are configured with the following memory modules:

- Dual-access random-access memory (DARAM)
- Flash EEPROM

dual-access RAM (DARAM)

There are 544 words × 16 bits of DARAM on the ’F240 device. The ’F240 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.

dual-access RAM (DARAM) (continued)

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 block 0 as program memory, instructions can be downloaded from external program memory into on-chip RAM and then executed. When using on-chip RAM, or high-speed external memory, the ’F240 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 ’F240

architecture enables the device to perform three concurrent memory accesses in any given machine cycle. Externally, the READY line can be used to interface the ’F240 to slower, less expensive external memory. Downloading programs from slow off-chip memory to on-chip RAM can speed processing while cutting system costs.

flash EEPROM

Flash EEPROM provides an attractive alternative to masked program ROM. Like ROM, flash is a nonvolatile memory type; however, it has the advantage of “in-target” reprogrammability. The SMJ320F240 incorporates one 16K $\times$ 16-bit flash EEPROM module in program space. This type of memory expands the capabilities of the SMJ320F240 in the areas of prototyping, early field-testing, and single-chip applications.

Unlike most discrete flash memory, the ’F240 flash does not require a dedicated state machine, because the algorithms for programming and erasing the flash are executed by the DSP core. This enables several advantages, including: reduced chip size and sophisticated, adaptive algorithms. For production programming, the IEEE Standard 1149.1 (JTAG) scan port provides easy access to the on-chip RAM for downloading the algorithms and flash code. Other key features of the flash include zero-wait-state access rate and single 5-V power supply.

An erased bit in the SMJ320F240 flash is read as a logic 1, and a programmed bit is read as a logic 0. The flash requires a block-erase of the entire 16K module; however, any combination of bits can be programmed. The following four algorithms are required for flash operations: clear, erase, flash-write, and program. For an explanation of these algorithms and a complete description of the flash EEPROM, see the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282), which is available during the 2nd quarter of 1998.

flash serial loader

The on-chip flash is shipped with a serial bootloader code programmed at the following addresses: 0x0000–0x00FFh. All other flash addresses are in an erased state. The serial bootloader can be used to program the on-chip Flash memory with user’s code. During the Flash programming sequence, the on-chip data RAM is used to load and execute the clear, erase, and program algorithms. See the TMS320F240 Serial Bootloader application report (currently located at **ftp://ftp.ti.com/pub/tms320bbs/c24xfiles/f240boot.pdf** to understand on-chip flash programming using the serial bootloader code. Look for further C2000 information using the DSP link at **www.ti.com**.

flash control mode register

The flash control mode register is located at I/O address FF0Fh. This register offers two options: register access mode and array access mode. Register access mode gives access to the four control registers in the memory space decoded for the flash module. These registers are used to control erasing, programming, and testing of the flash array. Register access mode is enabled by activating an `OUT` command with dummy data.

The `OUT xxxx, FF0Fh` instruction makes the flash registers accessible for reads and/or writes. After executing `OUT xxxx, FF0Fh`, the flash control registers are accessed in the memory space decoded for the flash module and the flash array cannot be accessed. The four registers are repeated every four address locations within the flash module’s decoded range.

After completing all the necessary reads and/or writes to the control registers, an `IN xxxx, FF0Fh` instruction (with dummy data) is executed to place the flash array back in array access mode. After executing the `IN xxxx, FF0Fh` instruction, the flash array is accessed in the decoded space and the flash registers are

flash control mode register (continued)

not available. Switching between the register access mode and the array access mode is done by issuing the `IN` and `OUT` instructions. The memory content in these instructions (denoted by `xxxx`) is not relevant. See the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282) for a detailed description of the flash programming algorithms.

peripherals

The integrated peripherals of the SMJ320F240 are described in the following subsections:

- External memory interface
- Event manager (EV)
- Dual analog-to-digital converter (ADC)
- Serial peripheral interface (SPI)
- Serial communications interface (SCI)
- Watchdog timer (WD)

external memory interface

The SMJ320F240 can address up to 64K words × 16 bits of memory or registers in each of the program, data, and I/O spaces. On-chip memory, when enabled, removes some of this off-chip range. In data space, the high 32K words can be mapped dynamically as either local or global using the GREG register. A data-memory access mapped as global asserts $\overline{BR}$ low (with timing similar to the address bus).

The CPU of the SMJ320F240 schedules a program fetch, data read, and data write on the same machine cycle. This is because, from on-chip memory, the CPU can execute all three of these operations in the same cycle. However, the external interface multiplexes the internal buses to one address and one data bus. The external interface sequences these operations to complete the data write first, then the data read, and finally the program read.

The ’F240 supports a wide range of system interfacing requirements. Program, data, and I/O address spaces provide interface to memory and I/O, maximizing system throughput. The full 16-bit address and data bus, along with the $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ space-select signals allow addressing of 64K 16-bit words in program and I/O space. Due to the on-chip peripherals, external data space is addressable to 32K 16-bit words.

I/O design is simplified by having I/O treated the same way as memory. I/O devices are mapped into the I/O address space using the processor’s external address and data buses in the same manner as memory-mapped devices.

The ’F240 external parallel interface provides control signals to facilitate interfacing to the device. The R/$\overline{W}$ output signal is provided to indicate whether the current cycle is a read or a write. The $\overline{STRB}$ output signal provides a timing reference for all external cycles.

Interface to memory and I/O devices of varying speeds is accomplished by using the READY input. When transactions are made with slower devices, the ’F240 processor waits until the other device completes its function and signals the processor by way of the READY input. Once a ready indication is provided from the external device, execution continues. On the ’F240 device, the READY input must be driven (active high) to complete reads or writes to internal data I/O-memory-mapped registers and all external addresses.

The bus request ($\overline{BR}$) signal is used in conjunction with the other ’F240 interface signals to arbitrate external global-memory accesses. Global memory is external data-memory space in which the $\overline{BR}$ signal is asserted at the beginning of the access. When an external global-memory device receives the bus request, it responds by asserting the ready signal after the global-memory access is arbitrated and the global access is completed.

external memory interface (continued)

The SMJ320F240 supports zero-wait-state reads on the external interface. However, to avoid bus conflicts, writes take two cycles. This allows the ’F240 to buffer the transition of the data bus from input to output (or output to input) by a half cycle. In most systems, SMJ320F240 ratio of reads to writes is significantly large to minimize the overhead of the extra cycle on writes.

Wait states can be generated when accessing slower external resources. The wait states operate on machine-cycle boundaries and are initiated either by using the ready signal or using the software wait-state generator. Ready can be used to generate any number of wait states.

event-manager (EV) module

The event-manager module includes general-purpose (GP) timers, full compare units, capture units, and quadrature-encoder pulse (QEP) circuits. Figure 11 shows the functions of the event manager.

general-purpose (GP) timers

There are three GP timers on the SMJ320F240. The GP timer x (for x = 1, 2, 3) includes:

- A 16-bit timer up-, up/down-counter, TxCNT for reads or writes
- A 16-bit timer-compare register (double-buffered with shadow register), TxCMPR for reads or writes
- A 16-bit timer-period register (double-buffered with shadow register), TxPR for reads or writes
- A 16-bit timer-control register,TxCON for reads or writes
- Selectable internal or external input clocks
- A programmable prescalar for internal or external clock inputs
- Control and interrupt logic for four maskable interrupts: underflow, overflow, timer compare, and period interrupts
- A timer-compare output pin with configurable active-low and active-high states, as well as forced-low and forced-high states.
- A selectable direction (TMRDIR) input pin (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. A 32-bit GP timer also can be configured using GP timer 2 and 3. The compare register associated with each GP timer can be used for compare function and PWM-waveform generation. There are two single and three continuous modes of operation 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. The state of each GP timer/compare output is configurable by the general-purpose timer-control register (GPTCON). 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 1 or 2 for the simple compares to generate additional compare or PWMs, GP timer 2 or 3 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.

general-purpose (GP) timers (continued)

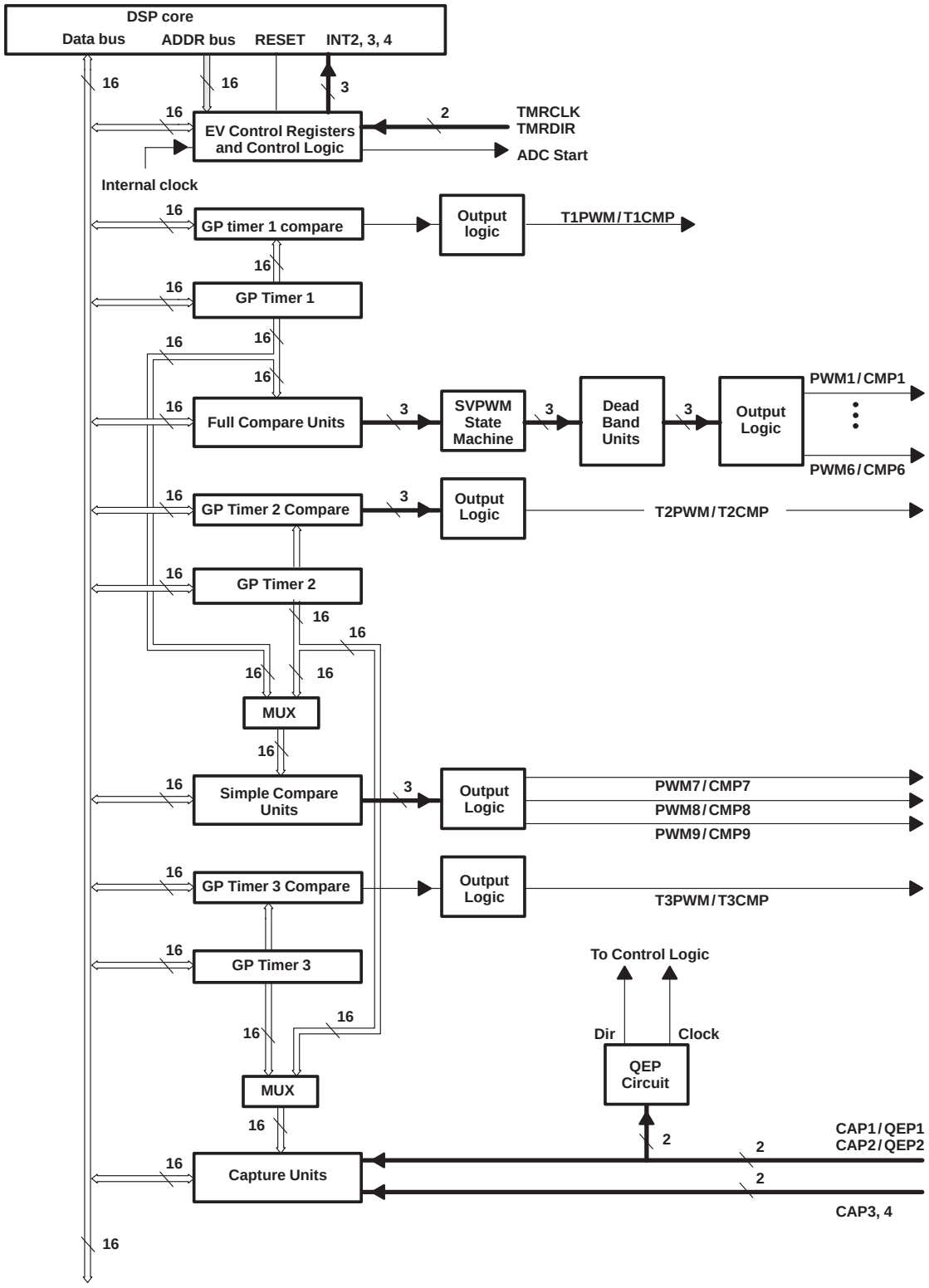


Figure 11. Event-Manager Block Diagram

full compare units

There are three full compare units on SMJ320F240. These compare units use GP timer1 as the timebase 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 value (from 0 to 102 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.

simple compares

SMJ320F240 is equipped with three simple compares that can be used to generate three additional independent compare or high-precision PWM waveforms. GP timer1 or 2 can be selected as the timebase for the three simple compares. The states of the outputs of the three simple compares are configurable as low-active, high-active, forced-low, or forced-high independently. Simple compare registers are double-buffered, allowing programmable change of the compare/PWM pulse widths as needed. The state of the simple-compare outputs is configurable and changeable as needed by way of the double-buffered SACTR register.

compare/PWM waveform generation

Up to 12 compare and/or PWM waveforms (outputs) can be generated simultaneously by SMJ320F240: three independent pairs (six outputs) by the three full compare units with *programmable deadbands*, three independent compares or PWMs (three outputs) by the simple compares, and three independent compare and PWMs (three outputs) by the GP-timer compares.

compare/PWMs characteristics

Characteristics of the compare/PWMs are as follow:

- 16-bit, 50-ns resolutions
- Programmable deadband for the PWM output pairs, from 0 to 102 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 and/or GP timer 3 counter are captured and stored in the two-level first-in first-out (FIFO) stacks when selected transitions are detected on capture input pins, CAPx for x = 1, 2, 3, or 4. The capture unit of the SMJ320F240 consists of four capture circuits.

capture unit (continued)

- The capture unit includes the following features:
  - One 16-bit capture-control register, CAPCON, for reads or writes
  - One 16-bit capture-FIFO status register, CAPFIFO, with eight MSBs for read-only operations, and eight LSBs for write-only operations
  - Optional selection of GP timer 2 and/or GP timer 3 through two 16-bit multiplexers (MUXs). One MUX selects a GP timer for capture circuits 3 and 4, and the other MUX selects a GP timer for capture circuits 1 and 2.
  - Four 16 bit x 2 FIFO stack registers, one two-level FIFO stack register per capture circuit. The top register of each stack is a read-only register, FIFOx, where x = 1, 2, 3, or 4.
  - Four possible Schmitt-triggered capture-input pins (CAPx, x = 1 to 4) with one input pin per capture unit
  - The input pins CAP1 and CAP2 also can be used as inputs to the QEP circuit.
  - User-specified edge-detection mode at the input pins
  - Four maskable interrupts/flags, CAPINTx, where x = 1, 2, 3, or 4

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 or 3 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 12. The ADC module consists of two 10-bit ADCs with two built-in sample-and-hold (S/H) circuits. A total of 16 analog input channels is available on the SMJ320F240. Eight analog inputs are provided for each ADC unit by way of an 8-to-1 analog multiplexer. Minimum total conversion time for each ADC unit is 6.1 μs. Total accuracy for each converter is ±1.5 LSB. Reference voltage for the ADC module needs to be supplied externally through the two reference pins, V_{REFHI} and V_{REFLO}. The digital result is expressed as:

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

Functions of the ADC module include:

- Two input channels (one for each ADC unit) that can be sampled and converted simultaneously
- Each ADC unit can perform single or continuous S/H and conversion operations.
- Two 2-level-deep FIFO result registers for ADC units 1 and 2
- ADC module (both A/D converters) can start operation by software instruction, external signal transition on a device pin, or by event-manager events on each of the GP timer/compare output and the capture 4 pins.
- The ADC control register is double-buffered (with shadow register) and can be written to at any time. A new conversion of ADC can start immediately or when the previous conversion process is completed according to the control register bits.
- At the end of each conversion, an interrupt flag is set and an interrupt is generated if it is unmasked/enabled.

analog-to-digital converter (ADC) module (continued)

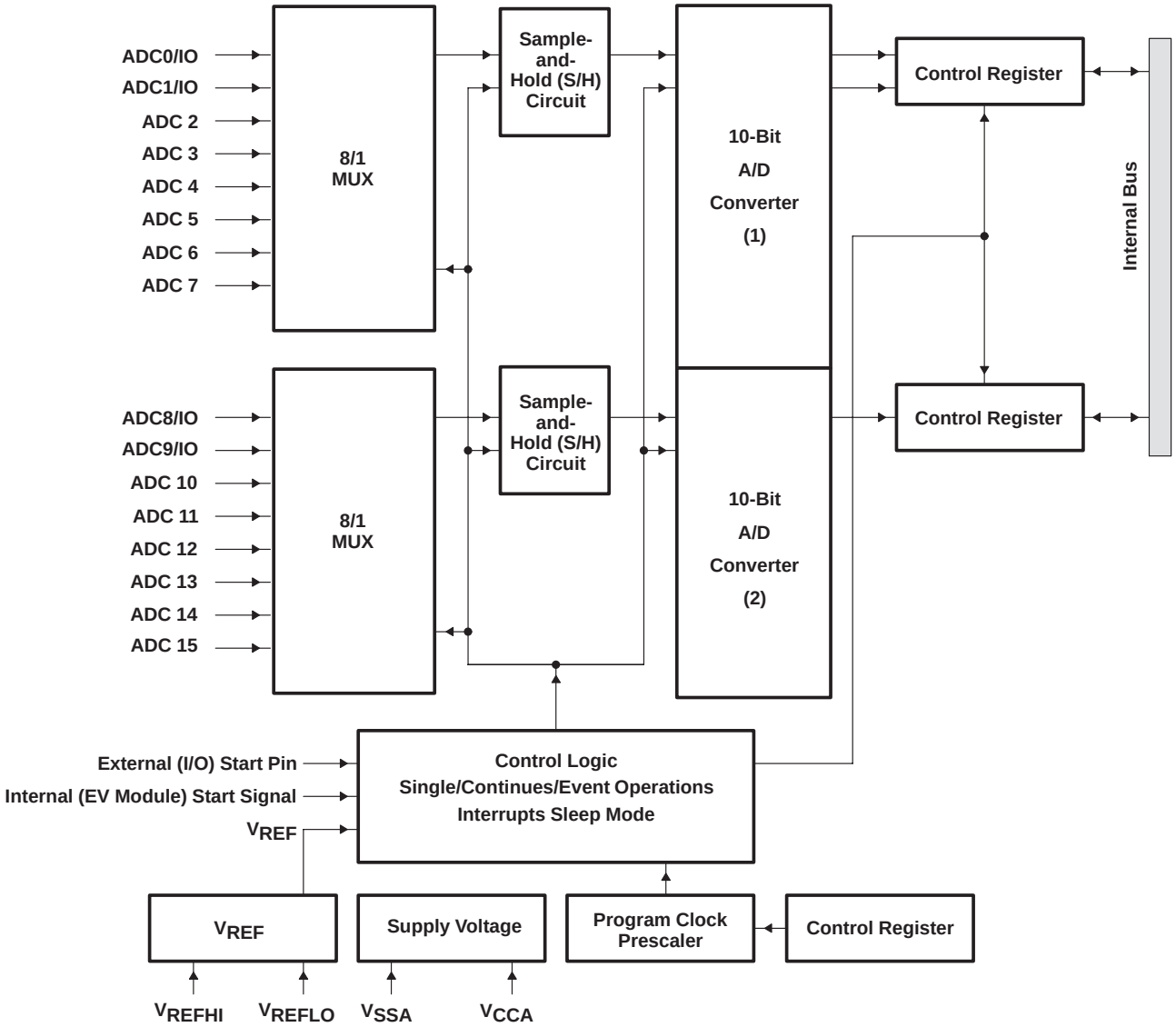


Figure 12. Analog-to-Digital Converter Module

serial peripheral interface (SPI) module

The SMJ320F240 devices include the four-pin serial peripheral interface (SPI) module. The SPI is a high-speed synchronous serial-I/O port that allows a serial bit stream of programmed length (one to eight bits) to be shifted into and out of the device at a programmable bit-transfer rate. Normally, the SPI is used for communications between the DSP controller and external peripherals or another processor. Typical applications include external I/O or peripheral expansion through devices such as shift registers, display drivers, and ADCs. Multidevice communications are supported by the master/slave operation of the SPI.

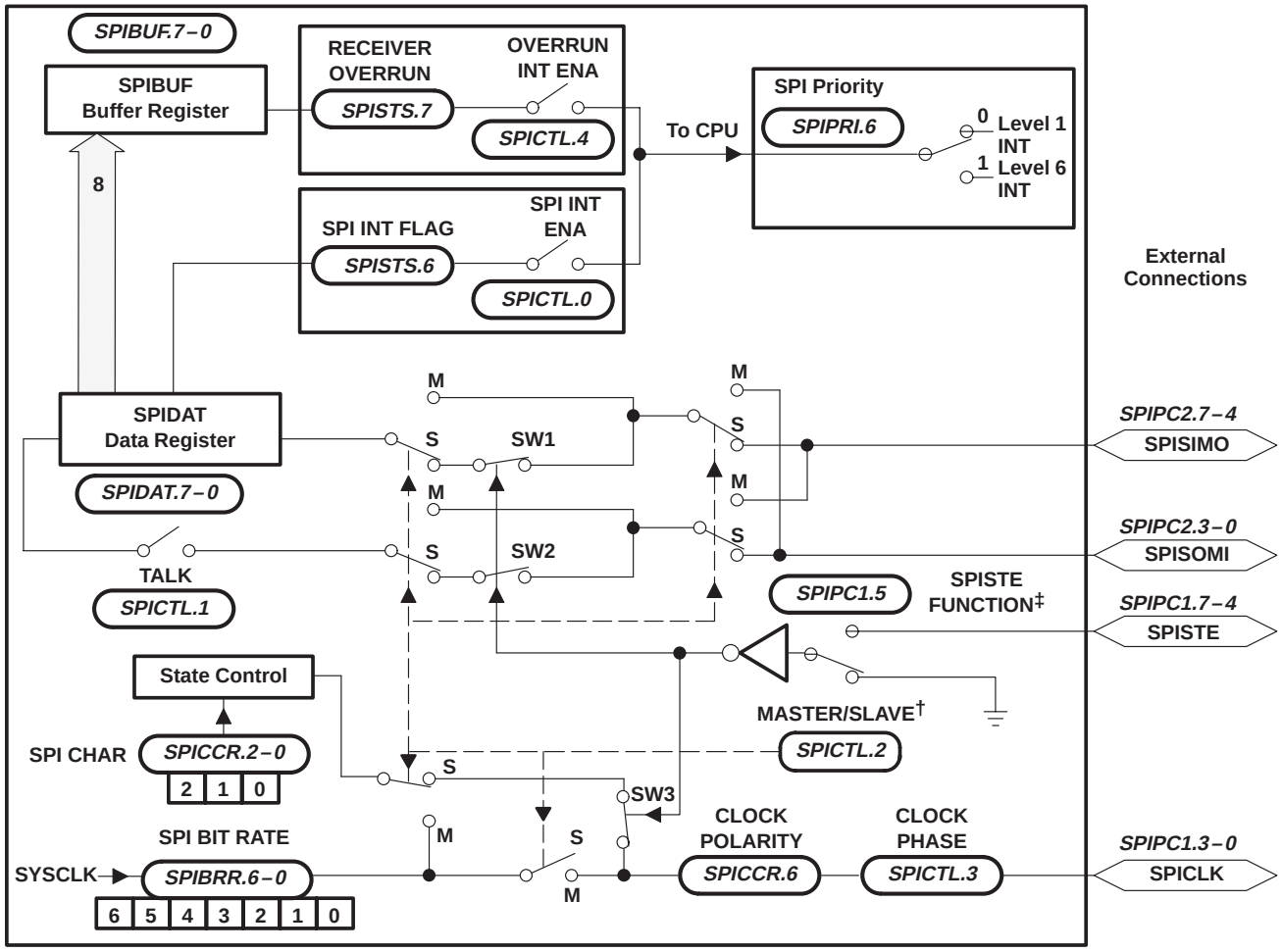
The SPI module features include the following:

- Four external pins:
  - SPISOMI: SPI slave-output/master-input pin, or general-purpose bidirectional I/O pin
  - SPISIMO: SPI slave-input/master-output pin, or general-purpose bidirectional I/O pin
  - SPISTE: SPI slave-transmit-enable pin, or general-purpose bidirectional I/O pin
  - SPICLK: SPI serial-clock pin, or general-purpose bidirectional I/O pin
- Two operational modes: master and slave
- Baud rate: 125 different programmable rates / 2.5 Mbps at 10-MHz SYSCLK
- Data word format: one to eight data bits
- Four clocking schemes controlled by clock polarity and clock-phase bits include:
  - Falling edge without phase delay: SPICLK active high. SPI transmits data on the falling edge of the SPICLK signal and receives data on the rising edge of the SPICLK signal.
  - Falling edge with phase delay: SPICLK active high. SPI transmits data one half-cycle ahead of the falling edge of the SPICLK signal and receives data on the falling edge of the SPICLK signal.
  - Rising edge without phase delay: SPICLK inactive low. SPI transmits data on the rising edge of the SPICLK signal and receives data on the falling edge of the SPICLK signal.
  - Rising edge with phase delay: SPICLK inactive low. SPI transmits data one half-cycle ahead of the falling edge of the SPICLK signal and receives data on the rising edge of the SPICLK signal.
- Simultaneous receive and transmit operations (transmit function can be disabled in software)
- Transmitter and receiver operations are accomplished through either interrupt-driven or polled algorithms.
- Ten SPI module control registers: Located in control register frame beginning at address 7040h.

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 13 is a block diagram of the SPI in slave mode.

serial peripheral Interface (SPI) module (continued)



† The diagram is shown in the slave mode.
‡ The SPISTE pin is shown as being disabled, meaning that data cannot be transmitted in this mode. Note that SW1, SW2, and SW3 are closed in this configuration.

Figure 13. Four-Pin Serial Peripheral Interface Module Block Diagram†

serial communications interface (SCI) module

The SMJ320F240 devices include 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 (baud) is programmable to over 65 000 different speeds through a 16-bit baud-select register. Features of the SCI module include:

- Two external pins
  - SCITXD: SCI transmit-output pin or general-purpose bidirectional I/O pin
  - SCIRXD: SCI receive-input pin or general-purpose bidirectional I/O pin
- Baud rate programmable to 64K different rates
  - Up to 625 Kbps at 10-MHz SYSCLK
- 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
- Eleven 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 interfaced 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 14 shows the SCI module block diagram.

serial communications interface (SCI) module (continued)

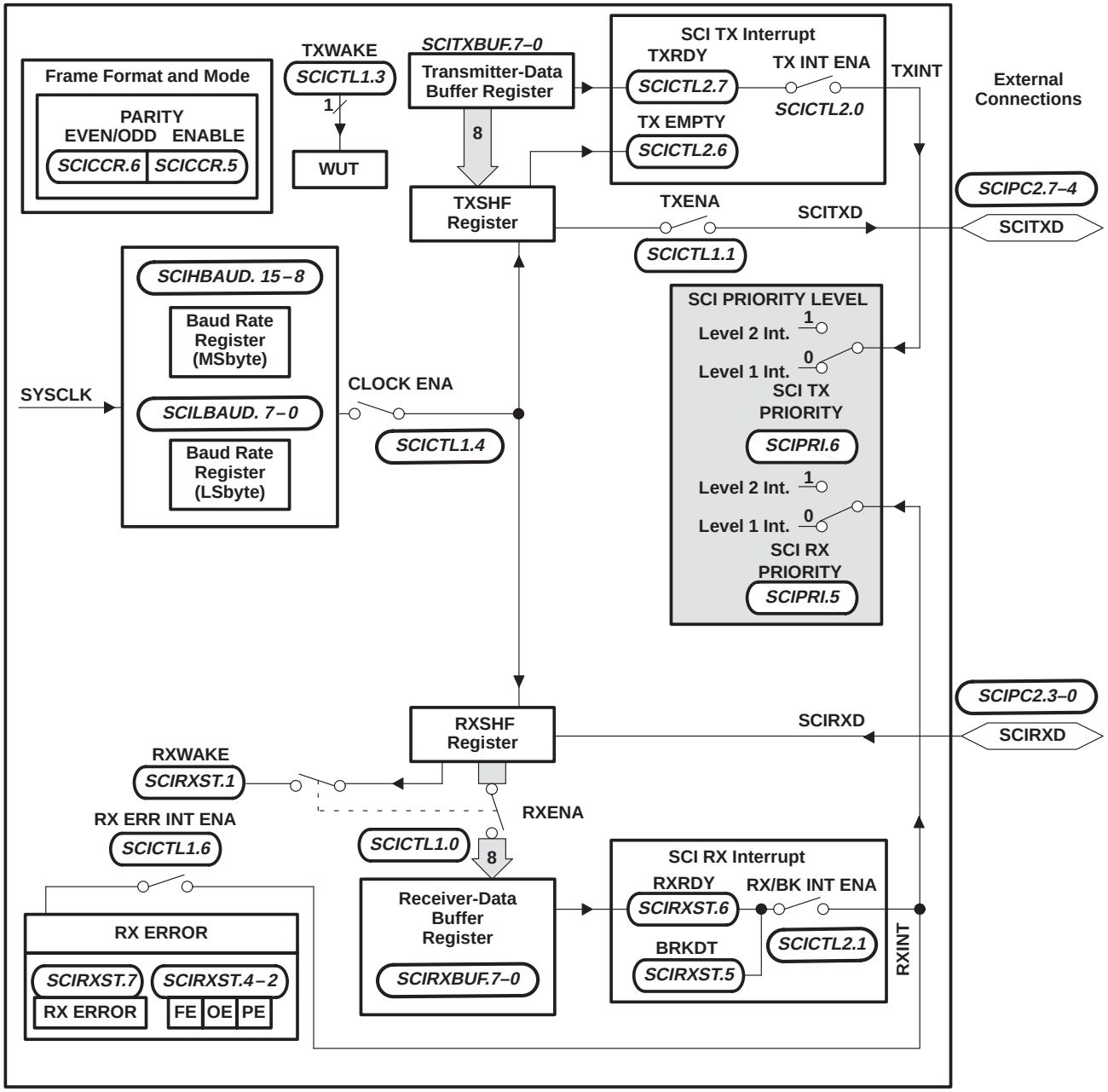


Figure 14. Serial Communications Interface (SCI) Module Block Diagram

watchdog (WD) and real-time interrupt (RTI) module

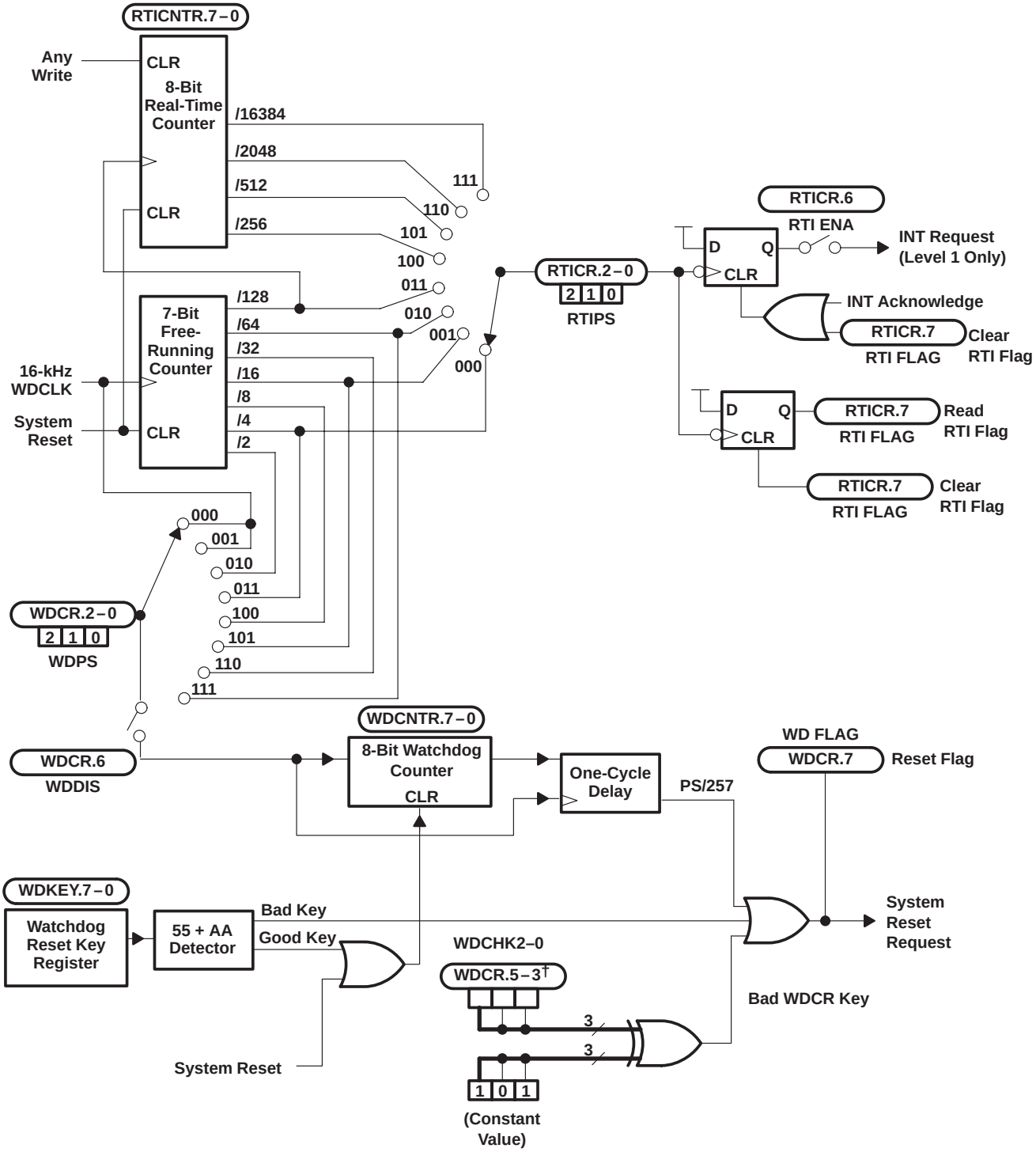
The SMJ320F240 device includes a watchdog (WD) timer and a real-time interrupt (RTI) module. The WD function of this module monitors software and hardware operation by generating a system reset if it is not periodically serviced by software by having the correct key written. The RTI function provides interrupts at programmable intervals. See Figure 15 for a block diagram of the WD/RTI module. The WD/RTI module features include the following:

- WD Timer
  - Seven different WD overflow rates ranging from 15.63 ms to 1 s
  - 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
  - A WD flag (WD FLAG) that indicates whether the WD timer initiated a system reset
  - 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.
- Real-time interrupt (RTI):
  - Interrupt generation at a programmable frequency of 1 to 4096 interrupts per second
  - Interrupt or polled operation
  - Two RTI control registers located in control register frame beginning at address 7020h.

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 15 shows the WD/RTI block diagram.

watchdog (WD) and real-time interrupt (RTI) module (continued)



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

Figure 15. WD/RTI Module Block Diagram

scan-based emulation

SMJ320F240 devices use scan-based emulation for code- and hardware-development support. Serial scan interface is provided by the test-access port. 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.

SMJ320F240 instruction set

The ’F240 microprocessor implements a comprehensive instruction set that supports both numeric-intensive signal-processing operations and general-purpose applications, such as multiprocessing and high-speed control. Source code for the ’C1x and ’C2x DSPs is upwardly compatible with the ’x2xx devices.

For maximum throughput, the next instruction is prefetched while the current one is being executed. Because the same data lines are used to communicate to external data, program, or I/O space, the number of cycles an instruction requires to execute varies, depending upon whether the next data operand fetch is from internal or external memory. Highest throughput is achieved by maintaining data memory on chip and using either internal or fast external program memory.

addressing modes

The SMJ320F240 instruction set provides four basic memory-addressing modes: direct, indirect, immediate, and register.

In direct addressing, the instruction word contains the lower 7 bits of the data memory address. This field is concatenated with the 9 bits of the data memory page pointer (DP) to form the 16-bit data memory address. Therefore, in the direct-addressing mode, data memory is paged effectively with a total of 512 pages, each page containing 128 words.

Indirect addressing accesses data memory through the auxiliary registers. In this addressing mode, the address of the instruction operand is contained in the currently selected auxiliary register. Eight auxiliary registers (AR0–AR7) provide flexible and powerful indirect addressing. To select a specific auxiliary register, the auxiliary register pointer (ARP) is loaded with a value from 0 to 7 for AR0 through AR7, respectively.

There are seven types of indirect addressing: autoincrement or autodecrement, postindexing by adding or subtracting the contents of AR0, single-indirect addressing with no increment or decrement, and bit-reversed addressing [used in fast fourier transforms (FFTs)] with increment or decrement. All operations are performed on the current auxiliary register in the same cycle as the original instruction, following which the current auxiliary register and ARP can be modified.

In immediate addressing, the actual operand data is provided in a portion of the instruction word or words. There are two types of immediate addressing: long and short. In short-immediate addressing, the data is contained in a portion of the bits in a single-word instruction. In long-immediate addressing, the data is contained in the second word of a two-word instruction. The immediate-addressing mode is useful for data that does not need to be stored or used more than once during the course of program execution (for example, initialization values or constants).

The register-addressing mode uses operands in CPU registers either explicitly, such as with a direct reference to a specific register, or implicitly, with instructions that intrinsically reference certain registers. In either case, operand reference is simplified because 16-bit values can be used without specifying a full 16-bit operand address or immediate value.

repeat feature

The repeat function can be used with instructions (as defined in Table 14) such as multiply/accumulates (MAC and MACD), block moves (BLDD and BLPD), I/O transfers (IN/OUT), and table read/writes (TBLR/TBLW). These instructions, although normally multicycle, are pipelined when the repeat feature is used, and they effectively become single-cycle instructions. For example, the table-read instruction can take three or more cycles to execute, but when the instruction is repeated, a table location can be read every cycle.

repeat feature (continued)

The repeat counter (RPTC) is loaded with the addressed data-memory location if direct or indirect addressing is used, and with an 8-bit immediate value if short-immediate addressing is used. The RPTC register is loaded by the RPT instruction. This results in a maximum of N + 1 executions of a given instruction. RPTC is cleared by reset. Once a repeat instruction (RPT) is decoded, all interrupts, including NMI (but excluding reset), are masked until the completion of the repeat loop.

instruction set summary

This section summarizes the operation codes (opcodes) of the instruction set for the ’F240 digital signal processors. This instruction set is a superset of the ’C1x and ’C2x instruction sets. The instructions are arranged according to function and are alphabetized by mnemonic within each category. The symbols in Table 13 are used in the instruction set summary table (Table 14). The TI ’C2xx assembler accepts ’C2x instructions.

The number of words that an instruction occupies in program memory is specified in column 3 of Table 14. Several instructions specify two values separated by a slash mark (/) for the number of words. In these cases, different forms of the instruction occupy a different number of words. For example, the ADD instruction occupies one word when the operand is a short-immediate value or two words if the operand is a long-immediate value.

The number of cycles that an instruction requires to execute is also in column 3 of Table 14. All instructions are assumed to be executed from internal program memory (RAM) and internal data dual-access memory. The cycle timings are for single-instruction execution, not for repeat mode.

instruction set summary (continued)

Table 13. SMJ320F240 Opcode Symbols

SYMBOL	DESCRIPTION
A	Address
ACC	Accumulator
ACCB	Accumulator buffer
ARx	Auxiliary register value (0–7)
BITx	4-bit field that specifies which bit to test for the BIT instruction
BMAR	Block-move address register
DBMR	Dynamic bit-manipulation register
I	Addressing-mode bit
II...II	Immediate operand value
INTM	Interrupt-mode flag bit
INTR#	Interrupt vector number
K	Constant
PREG	Product register
PROG	Program memory
RPTC	Repeat counter
SHF, SHFT	3/4-bit shift value
TC	Test-control bit
T P	Two bits used by the conditional execution instructions to represent the conditions TC, NTC, and BIO. T P Meaning 0 0 $\overline{\text{BIO}}$ low 0 1 TC = 1 1 0 TC = 0 1 1 None of the above conditions
TREGn	Temporary register n (n = 0, 1, or 2)
Z L V C	4-bit field representing the following conditions: Z: ACC = 0 L: ACC < 0 V: Overflow C: Carry A conditional instruction contains two of these 4-bit fields. The 4-LSB field of the instruction is a 4-bit mask field. A 1 in the corresponding mask bit indicates that the condition is being tested. The second 4-bit field (bits 4–7) indicates the state of the conditions designated by the mask bits as being tested. For example, to test for ACC ≥ 0, the Z and L fields are set while the V and C fields are not set. The next 4-bit field contains the state of the conditions to test. The Z field is set to indicate testing of the condition ACC = 0, and the L field is reset to indicate testing of the condition ACC ≥ 0. The conditions possible with these 8 bits are shown in the BCND and CC instructions. To determine if the conditions are met, the 4-LSB bit mask is ANDed with the conditions. If any bits are set, the conditions are met.

instruction set summary (continued)

Table 14. SMJ320F240 Instruction Set Summary

'F240 MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE MSB			LSB
ABS	Absolute value of accumulator	1/1	1011	1110	0000	0000
ADD	Add to accumulator with shift	1/1	0010	SHFT	IADD	RESS
	Add to high accumulator	1/1	0110	0001	IADD	RESS
	Add to accumulator short immediate	1/1	1011	1000	KKKK	KKKK
	Add to accumulator long immediate with shift	2/2	1011	1111	1001	SHFT
ADDC	Add to accumulator with carry	1/1	0110	0000	IADD	RESS
ADDS	Add to low accumulator with sign extension suppressed	1/1	0110	0010	IADD	RESS
ADDT	Add to accumulator with shift specified by T register	1/1	0110	0011	IADD	RESS
ADRK	Add to auxiliary register short immediate	1/1	0111	1000	KKKK	KKKK
AND	AND with accumulator	1/1	0110	1110	IADD	RESS
	AND immediate with accumulator with shift	2/2	1011	1111	1011	SHFT 16-Bit Constant
	AND immediate with accumulator with shift of 16	2/2	1011	1110	1000	0001 16-Bit Constant
APAC	Add P register to accumulator	1/1	1011	1110	0000	0100
B	Branch unconditionally	2/4	0111	1001	IADD	RESS Branch Address
BACC	Branch to address specified by accumulator	1/4	1011	1110	0010	0000
BANZ	Branch on auxiliary register not zero	2/4/2	0111	1011	IADD	RESS Branch Address
BCND	Branch if TC bit ≠ 0	2/4/2	1110	0001	0000	0000 Branch Address
	Branch if TC bit = 0	2/4/2	1110	0010	0000	0000 Branch Address
	Branch on carry	2/4/2	1110	0011	0001	0001 Branch Address
	Branch if accumulator ≥ 0	2/4/2	1110	0011	1000	1100 Branch Address
	Branch if accumulator > 0	2/4/2	1110	0011	0000	0100 Branch Address
	Branch on I/O status low	2/4/3	1110	0000	0000	0000 Branch Address
	Branch if accumulator ≤ 0	2/4/2	1110	0011	1100	1100 Branch Address
	Branch if accumulator < 0	2/4/2	1110	0011	0100	0100 Branch Address
	Branch on no carry	2/4/2	1110	0011	0000	0001 Branch Address
	Branch if no overflow	2/4/2	1110	0011	0000	0010 Branch Address

instruction set summary (continued)

Table 14. SMJ320F240 Instruction Set Summary (Continued)

'F240 MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
BCND	Branch if accumulator ≠ 0	2/4/2	1110	0011	0000	1000
			Branch Address			
	Branch on overflow	2/4/2	1110	0011	0010	0010
			Branch Address			
	Branch if accumulator = 0	2/4/2	1110	0011	1000	1000
			Branch Address			
BIT	Test bit	1/1	0100	BITx	IADD	RESS
BITT	Test bit specified by TREG	1/1	0110	1111	IADD	RESS
BLDD†	Block move from data memory to data memory source immediate	2/3	1010	1000	IADD	RESS
			Branch Address			
	Block move from data memory to data memory destination immediate	2/3	1010	1001	IADD	RESS
			Branch Address			
BLPD	Block move from program memory to data memory	2/3	1010	0101	IADD	RESS
			Branch Address			
CALA	Call subroutine indirect	1/4	1011	1110	0011	0000
CALL	Call subroutine	2/4	0111	1010	IADD	RESS
			Routine Address			
CC	Conditional call subroutine	2/4/2	1110	10TP	ZLVC	ZLVC
			Routine Address			
CLRC	Configure block as data memory	1/1	1011	1110	0100	0100
	Enable interrupt	1/1	1011	1110	0100	0000
	Reset carry bit	1/1	1011	1110	0100	1110
	Reset overflow mode	1/1	1011	1110	0100	0010
	Reset sign-extension mode	1/1	1011	1110	0100	0110
	Reset test/control flag	1/1	1011	1110	0100	1010
	Reset external flag	1/1	1011	1110	0100	1100
CMPL	Complement accumulator	1/1	1011	1110	0000	0001
CMPR	Compare auxiliary register with auxiliary register AR0	1/1	1011	1111	0100	01CM
DMOV	Data move in data memory	1/1	0111	0111	IADD	RESS
IDLE	Idle until interrupt	1/1	1011	1110	0010	0010
IN	Input data from port	2/2	1010	1111	IADD	RESS
			16BIT	I/O	PORT	ADRS
INTR	Software-interrupt	1/4	1011	1110	011K	KKKK
LACC	Load accumulator with shift	1/1	0001	SHFT	IADD	RESS
	Load accumulator long immediate with shift	2/2	1011	1111	1000	SHFT
			16-Bit Constant			
	Zero low accumulator and load high accumulator	1/1	0110	1010	IADD	RESS

† In 'F240 devices, the BLDD instruction does not work with memory-mapped registers IMR, IFR, and GREG.

instruction set summary (continued)

Table 14. SMJ320F240 Instruction Set Summary (Continued)

'F240 MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
LACL	Load accumulator immediate short	1/1	1011	1001	KKKK	KKKK
	Zero accumulator	1/1	1011	1001	0000	0000
	Zero low accumulator and load high accumulator	1/1	0110	1010	IADD	RESS
	Zero low accumulator and load low accumulator with no sign extension	1/1	0110	1001	IADD	RESS
LACT	Load accumulator with shift specified by T register	1/1	0110	1011	IADD	RESS
LAR	Load auxiliary register	1/2	0000	0ARx	IADD	RESS
	Load auxiliary register short immediate	1/2	1011	0ARx	KKKK	KKKK
	Load auxiliary register long immediate	2/2	1011	1111	0000	1ARx
			16-Bit Constant			
LDP	Load data-memory page pointer	1/2	0000	1101	IADD	RESS
	Load data-memory page pointer immediate	1/2	1011	110P	AGEP	OINT
LPH	Load high-P register	1/1	0111	0101	IADD	RESS
LST	Load status register ST0	1/2	0000	1110	IADD	RESS
	Load status register ST1	1/2	0000	1111	IADD	RESS
LT	Load TREG	1/1	0111	0011	IADD	RESS
LTA	Load TREG and accumulate previous product	1/1	0111	0000	IADD	RESS
LTD	Load TREG, accumulate previous product, and move data	1/1	0111	0010	IADD	RESS
LTP	Load TREG and store P register in accumulator	1/1	0111	0001	IADD	RESS
LTS	Load TREG and subtract previous product	1/1	0111	0100	IADD	RESS
MAC	Multiply and accumulate	2/3	1010	0010	IADD	RESS
			16-Bit Constant			
MACD	Multiply and accumulate with data move	2/3	1010	0011	IADD	RESS
			16-Bit Constant			
MAR	Load auxiliary register pointer	1/1	1000	1011	1000	1ARx
	Modify auxiliary register	1/1	1000	1011	IADD	RESS
MPY	Multiply (with TREG, store product in P register)	1/1	0101	0100	IADD	RESS
	Multiply immediate	1/1	110C	KKKK	KKKK	KKKK
MPYA	Multiply and accumulate previous product	1/1	0101	0000	IADD	RESS
MPYS	Multiply and subtract previous product	1/1	0101	0001	IADD	RESS
MPYU	Multiply unsigned	1/1	0101	0101	IADD	RESS
NEG	Negate accumulator	1/1	1011	1110	0000	0010
NMI	Nonmaskable interrupt	1/4	1011	1110	0101	0010
NOP	No operation	1/1	1000	1011	0000	0000
NORM	Normalize contents of accumulator	1/1	1010	0000	IADD	RESS
OR	OR with accumulator	1/1	0110	1101	IADD	RESS
	OR immediate with accumulator with shift	2/2	1011	1111	1100	SHFT
			16-Bit Constant			
	OR immediate with accumulator with shift of 16	2/2	1011	1110	1000	0010
			16-Bit Constant			
OUT	Output data to port	2/3	0000 16BIT	1100 I/O	IADD PORT	RESS ADRS
PAC	Load accumulator with P register	1/1	1011	1110	0000	0011

instruction set summary (continued)

Table 14. SMJ320F240 Instruction Set Summary (Continued)

'F240 MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
POP	Pop top of stack to low accumulator	1/1	1011	1110	0011	0010
POPD	Pop top of stack to data memory	1/1	1000	1010	IADD	RESS
PSHD	Push data-memory value onto stack	1/1	0111	0110	IADD	RESS
PUSH	Push low accumulator onto stack	1/1	1011	1110	0011	1100
RET	Return from subroutine	1/4	1110	1111	0000	0000
RETC	Conditional return from subroutine	1/4/2	1110	11TP	ZLVC	ZLVC
ROL	Rotate accumulator left	1/1	1011	1110	0000	1100
ROR	Rotate accumulator right	1/1	1011	1110	0000	1101
RPT	Repeat instruction as specified by data-memory value	1/1	0000	1011	IADD	RESS
	Repeat instruction as specified by immediate value	1/1	1011	1011	KKKK	KKKK
SACH	Store high accumulator with shift	1/1	1001	1SHF	IADD	RESS
SACL	Store low accumulator with shift	1/1	1001	0SHF	IADD	RESS
SAR	Store auxiliary register	1/1	1000	0ARx	IADD	RESS
SBRK	Subtract from auxiliary register short immediate	1/1	0111	1100	KKKK	KKKK
SETC	Set carry bit	1/1	1011	1110	0100	1111
	Configure block as program memory	1/1	1011	1110	0100	0101
	Disable interrupt	1/1	1011	1110	0100	0001
	Set overflow mode	1/1	1011	1110	0100	0011
	Set test/control flag	1/1	1011	1110	0100	1011
	Set external flag XF	1/1	1011	1110	0100	1101
	Set sign-extension mode	1/1	1011	1110	0100	0111
SFL	Shift accumulator left	1/1	1011	1110	0000	1001
SFR	Shift accumulator right	1/1	1011	1110	0000	1010
SPAC	Subtract P register from accumulator	1/1	1011	1110	0000	0101
SPH	Store high-P register	1/1	1000	1101	IADD	RESS
SPL	Store low-P register	1/1	1000	1100	IADD	RESS
SPM	Set P register output shift mode	1/1	1011	1111	IADD	RESS
SQRA	Square and accumulate	1/1	0101	0010	IADD	RESS
SQRS	Square and subtract previous product from accumulator	1/1	0101	0011	IADD	RESS
SST	Store status register ST0	1/1	1000	1110	IADD	RESS
	Store status register ST1	1/1	1000	1111	IADD	RESS
SPLK	Store long immediate to data memory	2/2	1010	1110	IADD	RESS
			16-Bit Constant			
SUB	Subtract from accumulator long immediate with shift	2/2	1011	1111	1010	SHFT
			16-Bit Constant			
	Subtract from accumulator with shift	1/1	0011	SHFT	IADD	RESS
	Subtract from high accumulator	1/1	0110	0101	IADD	RESS
	Subtract from accumulator short immediate	1/1	1011	1010	KKKK	KKKK

instruction set summary (continued)

Table 14. SMJ320F240 Instruction Set Summary (Continued)

'F240 MNEMONIC	DESCRIPTION	WORDS/ CYCLES	OPCODE			
			MSB			LSB
SUBB	Subtract from accumulator with borrow	1/1	0110	0100	IADD	RESS
SUBC	Conditional subtract	1/1	0000	1010	IADD	RESS
SUBS	Subtract from low accumulator with sign extension suppressed	1/1	0110	0110	IADD	RESS
SUBT	Subtract from accumulator with shift specified by TREG	1/1	0110	0111	IADD	RESS
TBLR	Table read	1/3	1010	0110	IADD	RESS
TBLW	Table write	1/3	1010	0111	IADD	RESS
TRAP	Software interrupt	1/4	1011	1110	0101	0001
XOR	Exclusive-OR with accumulator	1/1	0110	1100	IADD	RESS
	Exclusive-OR immediate with accumulator with shift	2/2	1011	1111	1101	SHFT
			16-Bit Constant			
	Exclusive-OR immediate with accumulator with shift of 16	2/2	1011	1110	1000	0011
			16-Bit Constant			
ZALR	Zero low accumulator and load high accumulator with rounding	1/1	0110	1000	IADD	RESS

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

Supply voltage range, V_{DD} (see Note 1) . . . . . –0.3 V to 7 V
Input voltage range . . . . . –0.3 V to 7 V
Output voltage range . . . . . –0.3 V to 7 V
Operating free-air temperature range, T_A . . . . . –55°C to125°C
Storage temperature range, T_{stg} . . . . . –55°C to 150°C
Resolution . . . . . 10-bit (1024 values)
Monotonic . . . . . Assured
Output conversion mode . . . . . 000h to 3FFh (000h for $V_I \le V_{SSA}$; 3FFh for $V_I \ge V_{CCA}$)
Analog supply reference source, V_{REFHI} and V_{REFLO} . . . . . –0.3 V to 7 V
Analog input voltage range, V_{AI} . . . . . –0.3 V to 7 V

† 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.

NOTES: 1 V_{DD} refers to supply voltage types CV_{DD} (digital core supply voltage), DVDD (digital I/O supply voltage), and V_{DDP} (programming voltage supply). All voltages are measured with respect to V_{SS}.
2. Measured with respect to CV_{SS}.

recommended operating conditions

			MIN	NOM	MAX	UNIT
DV_{DD}	Digital supply voltage		4.5	5	5.5	V
V_{SS}	Supply voltage return			0		V
V_{IH}	High-level input voltage	XTAL1/CLKIN	3		V_{DD} + 0.3	V
		$\overline{\text{PORESET}}$, NMI, $\overline{\text{RS}}$, and $\overline{\text{TRST}}$	2.2		V_{DD} + 0.3	
		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		0.8	
I_{OH}	High-level output current	$\overline{\text{RS}}$			–19	mA
		See complete listing of pin names‡			–16	
		All other outputs			–23	
I_{OL}	Low-level output current	$\overline{\text{RS}}$			8	mA
		See complete listing of pin names‡			7.5	
		All other outputs			14.5	
T_A	Operating free-air temperature		–55		125	°C
T_{FP}	Flash programming operating temperature		–40		85	°C
Θ_{JA}	Thermal resistance, junction-to-ambient				55.71	°C/W
Θ_{JC}	Thermal resistance, junction-to-case				2.57	°C/W

‡ IOPA[0:3], SCIRXD/IO, SCITXD/IO, XINT2/IO, XINT3/IO, ADCSOC/IOPC0, TMRDIR/IOPB6, TMRCLK/IOPB7 EMU0, EMU1/$\overline{\text{OFF}}$

output current variation with output voltage: SPICE simulation results (4.5 V, 150° C)

Table 15. Typical Output Source Current vs. Output Voltage High

	2.4 V	3.0 V	3.5 V	4.0 V
$\overline{RS}$	–19 mA	–16 mA	–12 mA	–6 mA
See complete listing of pin names†	–16 mA	–13.5 mA	–9.5 mA	–5.0 mA
All other inputs	–23 mA	–18.5 mA	–13 mA	–6.5 mA

† IOPA[0:3], SCIRXD/IO, SCITXD/IO, XINT2/IO, XINT3/IO, ADCSOC/IOPC0, TMRDIR/IOPB6, TMRCLK/IOPB7 EMU0, EMU1/$\overline{OFF}$

Table 16. Typical Output Sink Current vs. Output Voltage Low

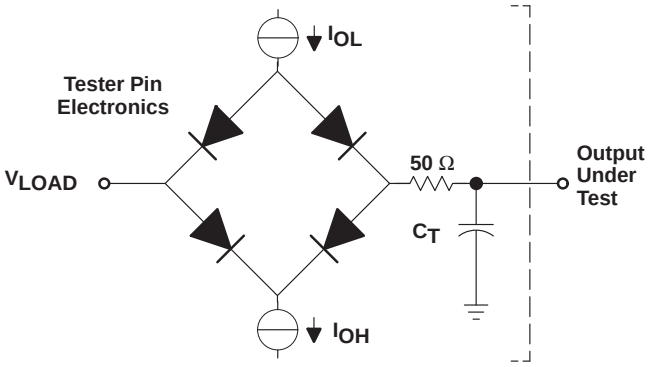
	0.6 V	0.4 V	0.2 V
$\overline{RS}$	8 mA	6 mA	3 mA
See complete listing of pin names†	7.5 mA	5 mA	2.5 mA
All other inputs	14.5 mA	10 mA	5.0 mA

† IOPA[0:3], SCIRXD/IO, SCITXD/IO, XINT2/IO, XINT3/IO, ADCSOC/IOPC0, TMRDIR/IOPB6, TMRCLK/IOPB7 EMU0, EMU1/$\overline{OFF}$

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

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage	I_{OH} = MAX		2.4			V
V_{OL}	Low-level output voltage	I_{OL} = MAX				0.6	V
I_I	Input current	$\overline{TRST}$ with internal pulldown		–10		500	µA
		EMU0, EMU1/$\overline{OFF}$, TMS, TCK, and TDI, with internal pullup	$V_I = V_{SS}$ or V_{DD}	–500		10	
		All other input pins		–10		10	
I_{OZ}	Output current, high-impedance state (off-state)	$V_O = V_{DD}$ or 0 V		–5		5	µA
I_{DD}	Supply current, operating mode	$t_{c(CO)}$ = 50 ns			80		mA
	Supply current, Idle 1 low-power mode	$t_{c(CO)}$ = 50 ns			50		
	Supply current, Idle 2 low-power mode	$t_{c(CO)}$ = 50 ns			7		
	Supply current, PLL power-down mode	$t_{c(CO)}$ = 50 ns			1		
	Supply current, OSC power-down mode	$t_{c(CO)}$ = 50 ns			400		µA
C_i	Input capacitance				15		pF
C_o	Output capacitance				15		pF
I_{DDP}	Flash programming supply current	$t_{c(CO)}$ = 50 ns			10		mA

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 16. Test Load Circuit

signal transition levels

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 17 shows the TTL-level outputs.

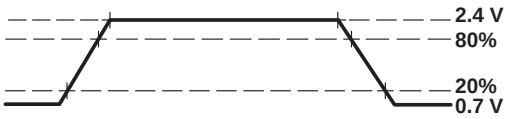


Figure 17. TTL-Level Outputs

TTL-compatible 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.

PARAMETER MEASUREMENT INFORMATION

Figure 18 shows the TTL-level inputs.

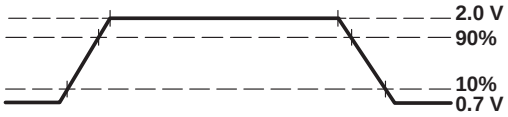


Figure 18. TTL-Level Inputs

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-A. To shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

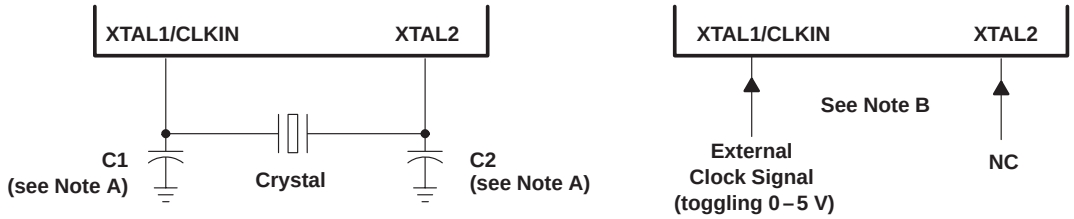
A	A[15:0]	MS	Memory strobe pins $\overline{IS}$, $\overline{DS}$, or $\overline{PS}$
CI	XTAL1/CLKIN	R	READY
CO	CLKOUT/IOPC1	RD	Read cycle or W/$\overline{R}$
D	D[15:0]	RS	$\overline{RS}$ or $\overline{PORESET}$
INT	NMI, XINT1, XINT2/IO, and XINT3/IO	W	Write cycle or $\overline{WE}$

Lowercase subscripts and their meanings:		Letters and symbols and their meanings:	
a	access time	H	High
c	cycle time (period)	L	Low
d	delay time	V	Valid
f	fall time	X	Unknown, changing, or don't care level
h	hold time	Z	High impedance
r	rise time		
su	setup time		
t	transition time		
v	valid time		
w	pulse duration (width)		

general notes on timing parameters

All output signals from the SMJ320F240 devices (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, see the appropriate cycle description section of this data sheet.



NOTES: A. For the values of C1 and C2, see the crystal manufacturer's specification.
B. Use this configuration in conjunction with $\overline{OSCBYP}$ pin pulled low.
C. Texas Instruments encourages customers to submit samples of the device to the resonator/crystal vendor for full characterization.

Figure 19. Recommended Crystal/Clock Connection

CLOCK OPTIONS

clock options

PARAMETER	CLKMD[1:0]
Clock-in mode, divide-by-2	00
Clock-in mode, divide-by-1	01
PLL enabled, divide-by-2 before PLL lock	10
PLL enabled, divide-by-1 before PLL lock	11

input clock frequency over operating free-air temperature range (PLL circuit disabled)

PARAMETER		MIN	MAX	UNIT
f_x	Divide-by-2 mode	0†	40	MHz
	Divide-by-1 mode	0†	20	

† 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.

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$] (see Note 3 and Figure 20)

PARAMETER		CLOCK MODE	MIN	TYP	MAX	UNIT
$t_{c(CPU)}$	Cycle time, CPUCLK	CLKIN divide by 2		$2t_{c(CI)}$	†	ns
		CLKIN divide by 1		$t_{c(CI)}$		
$t_{c(SYS)}$	Cycle time, SYSCLK	CPUCLK divide by 2		$2t_{c(CPU)}$	†	ns
		CPUCLK divide by 4‡		$4t_{c(CPU)}$		
$t_{c(CO)}$	Cycle time, CLKOUT	CLKIN divide by 2		$2t_{c(CI)}$	†	ns
		CLKIN divide by 1		$t_{c(CI)}$	†	
$t_{d(CIH-CO)}$	Delay time, XTAL1/CLKIN high to CLKOUT high/low		3	18	32	ns
$t_{f(CO)}$	Fall time, CLKOUT			5		ns
$t_{r(CO)}$	Rise time, CLKOUT			5		ns
$t_{w(COL)}$	Pulse duration, CLKOUT low		H–10	H–6	H–1	ns
$t_{w(COH)}$	Pulse duration, CLKOUT high		H+0	H+4	H+8	ns

† 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.
‡ SYSCLK is initialized to divide-by-4 mode by any device reset.
NOTE 3: Timings assume CLKOUT is set to output CPUCLK. CLKOUT is initialized to CPUCLK by power-on reset.

timing requirements (see Figure 20)

		CLOCK-IN MODE	MIN	MAX	UNIT
$t_{c(CI)}$	Cycle time, XTAL1/CLKIN	Divide by 2	25	†	ns
		Divide by 1	50	†	
$t_{f(CI)}$	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 $t_{c(CI)}$		45	55	%
$t_{w(CIH)}$	Pulse duration, XTAL1/CLKIN high as a percentage of $t_{c(CI)}$		45	55	%

† 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.

CLOCK OPTIONS (CONTINUED)

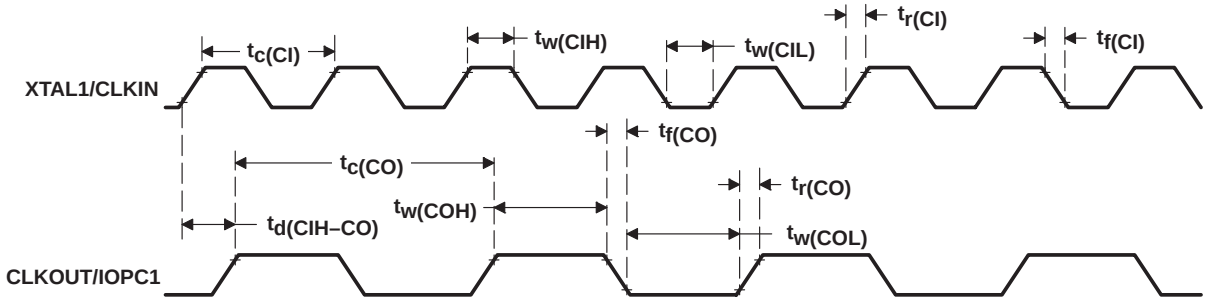


Figure 20. External Divide-by-Two Clock Timings

external reference crystal with PLL-circuit-enabled clock option

The internal oscillator is enabled by connecting $\overline{\text{OSCBYP}}$ to V_{DD} and connecting a crystal across XTAL1/CLKIN and XTAL2 pins as shown in Figure 19. The crystal should be in either fundamental or overtone operation and parallel resonant, with an effective series resistance of 30 Ω and a power dissipation of 1 mW; it should be specified at a load capacitance of 20 pF. Note that overtone crystals require an additional tuned-LC circuit.

input characteristics with the PLL circuit enabled

PARAMETER		EXTERNAL REFERENCE CRYSTAL	MIN	TYP	MAX	UNIT
f_x	Input clock frequency	4 MHz		4		MHz
		6 MHz		6		
		8 MHz		8		
C1, C2	Load capacitance			10		pF

switching characteristics over recommended operating conditions, H = 0.5 $t_{c(CO)}$ (see Figure 21)

PARAMETER		CLOCK MODE	MIN	TYP	MAX	UNIT
$t_{c(CPU)}$	Cycle time, CPUCLK	before PLL lock, CLKIN divide by 2		$2t_{c(CI)}$	†	ns
		before PLL lock, CLKIN divide by 1		$t_{c(CI)}$		
		after PLL lock		50		
$t_{c(SYS)}$	Cycle time, SYSCLK	CPUCLK divide by 2		$2t_{c(CPU)}$	†	ns
		CPUCLK divide by 4‡		$4t_{c(CPU)}$		
$t_{c(CO)}$	Cycle time, CLKOUT		50		†	ns
$t_{f(CO)}$	Fall time, CLKOUT			5		ns
$t_{r(CO)}$	Rise time, CLKOUT			5		ns
$t_{w(COL)}$	Pulse duration, CLKOUT low		H–10	H–6	H–1	ns
$t_{w(COH)}$	Pulse duration, CLKOUT high		H+0	H+4	H+8	ns
t_p	Transition time, PLL synchronized after PLL enabled	before PLL lock, CLKIN divide by 2			$2000t_{c(CI)}$	ns
		before PLL lock, CLKIN divide by 1			$1000t_{c(CI)}$	

† 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.
‡ SYSCLK is initialized to divide-by-4 mode by any device reset.

timing requirements (see Note 3 and Figure 21)

		EXTERNAL REFERENCE CRYSTAL	MIN	MAX	UNIT
$t_{c(CI)}$	Cycle time, XTAL1/CLKIN	4 MHz	250	†	ns
		6 MHz	167		
		8 MHz	125		
$t_{f(CI)}$	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.

NOTE 3: Timings assume CLKOUT is set to output CPUCLK. CLKOUT is initialized to CPUCLK by power-on reset.

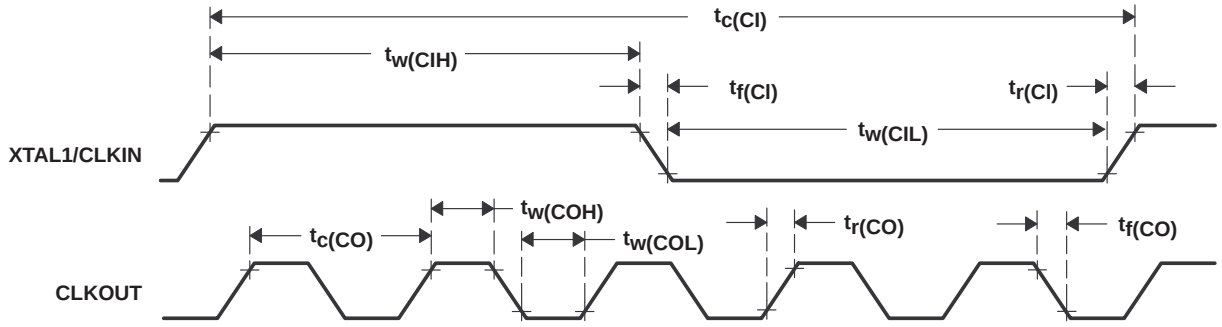


Figure 21. CLKIN-to-CLKOUT Timings for PLL Oscillator Mode, Multiply-by-5 Option With 4-MHz Crystal

low-power mode timings

switching characteristics over recommended operating conditions) (see Figure 22, Figure 23, Figure 24, and Figure 25)

PARAMETER		LOW-POWER MODES	MIN	TYP	MAX	UNIT
$t_{d(WAKE-A)}$	Delay time, CLKOUT switching to program execution resume (see Note 3)	Idle 1 and Idle 2		15 X $t_{c(CO)}$		ns
		PLL or OSC power down		15 X $t_{c(CI)}$		
$t_{d(IDLE-COH)}$	Delay time, Idle instruction executed to CLKOUT high (see Note 3)	Idle 2, PLL power down, OSC power down		500		ns
$t_{d(WAKE-LOCK)}$	Delay time, CLKOUT switching to PLL synchronized (see Note 3)	PLL or OSC power down		100		µs
$t_{d(WAKE-OSC)}$	Delay time, wakeup interrupt asserted to oscillator running	OSC power down		10		ms
$t_{d(IDLE-OSC)}$	Delay time, Idle instruction executed to oscillator power off	OSC power down		60		µs

NOTE 3: Timings assume CLKOUT is set to output CPUCLK. CLKOUT is initialized to CPUCLK by power-on reset.

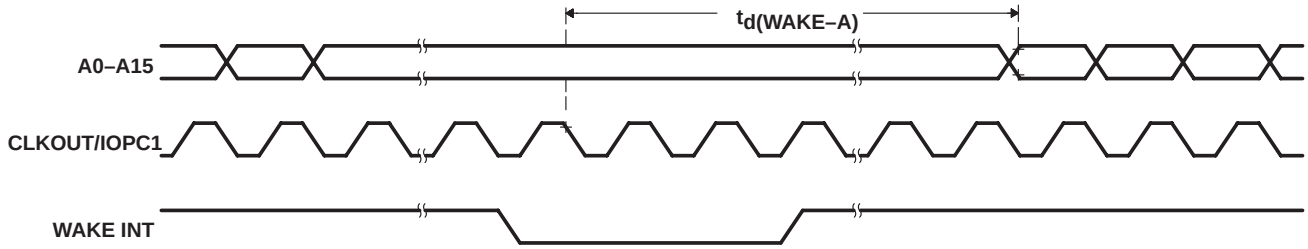


Figure 22. IDLE1 Entry and Exit Timings

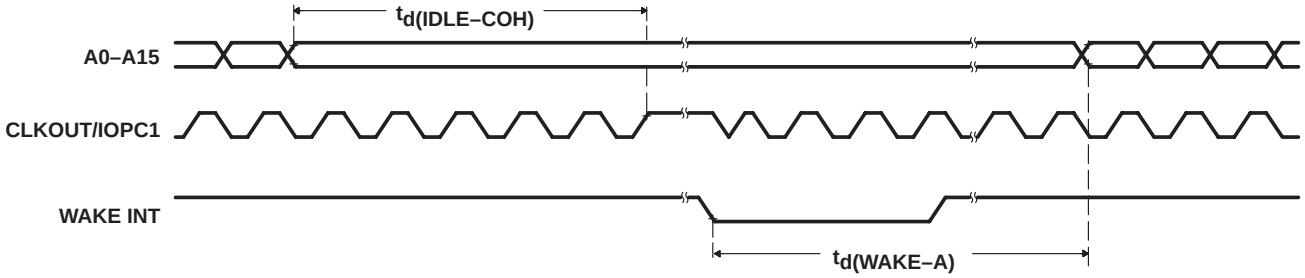


Figure 23. IDLE2 Entry and Exit Timings

low-power mode timings (continued)

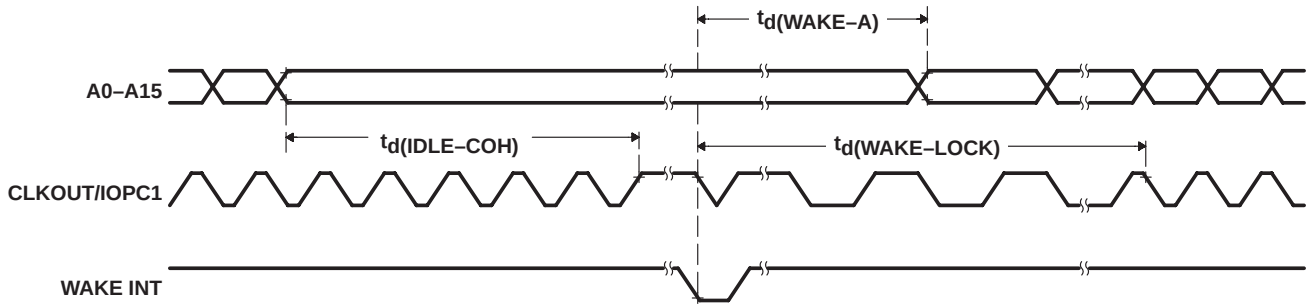


Figure 24. PLL Power-Down Entry and Exit Timings

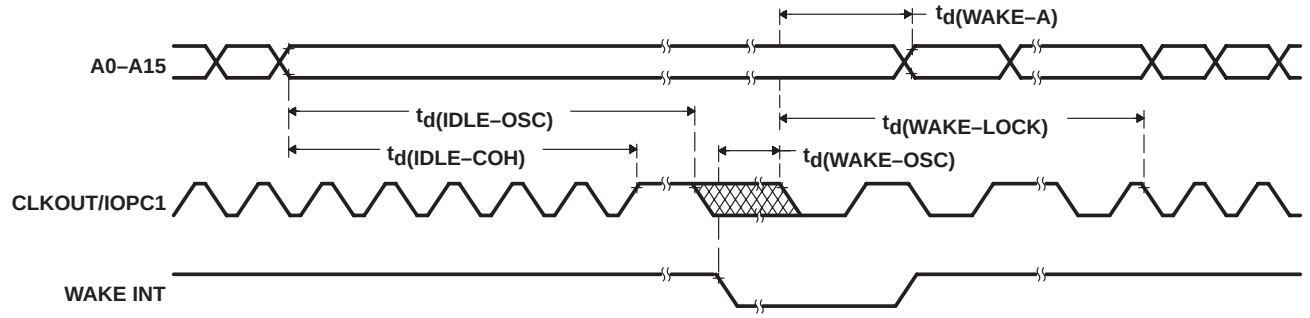


Figure 25. OSC Power-Down Entry and Exit Timings

memory and parallel I/O interface *read* timings

switching characteristics over recommended operating conditions for a memory read (see Figure 26)

PARAMETER		MIN	MAX	UNIT
$t_{d(CO-A)RD}$	Delay time, CLKOUT/IOPC1 low to address valid		17	ns
$t_{d(CO-SL)RD}$	Delay time, CLKOUT/IOPC1 low to $\overline{STRB}$ low		10	ns
$t_{d(CO-SH)RD}$	Delay time, CLKOUT/IOPC1 low to $\overline{STRB}$ high		6	ns
$t_{d(CO-ACTL)RD}$	Delay time, CLKOUT/IOPC1 low to $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, and $\overline{BR}$ low		10	ns
$t_{d(CO-ACTH)RD}$	Delay time, CLKOUT/IOPC1 low to $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, and $\overline{BR}$ high		10	ns

timing requirements for a memory read, H = $0.5t_{c(CO)}$† (see Figure 26)

			MIN	MAX	UNIT
$t_{a(A)}$	Access time, from address valid to read data	0 wait state		2H – 32	ns
		1 wait state		4H – 32	
$t_{su(D-COL)RD}$	Setup time, data read before CLKOUT/IOPC1 low		15		ns
$t_{h(COL-D)RD}$	Hold time, data read after CLKOUT/IOPC1 low		2		ns

† All timings with respect to CLKOUT/IOPC1 assume CLKSRC[1:0] bits are set to select CPUCLK for output.

memory and parallel I/O interface *read* timings (continued)

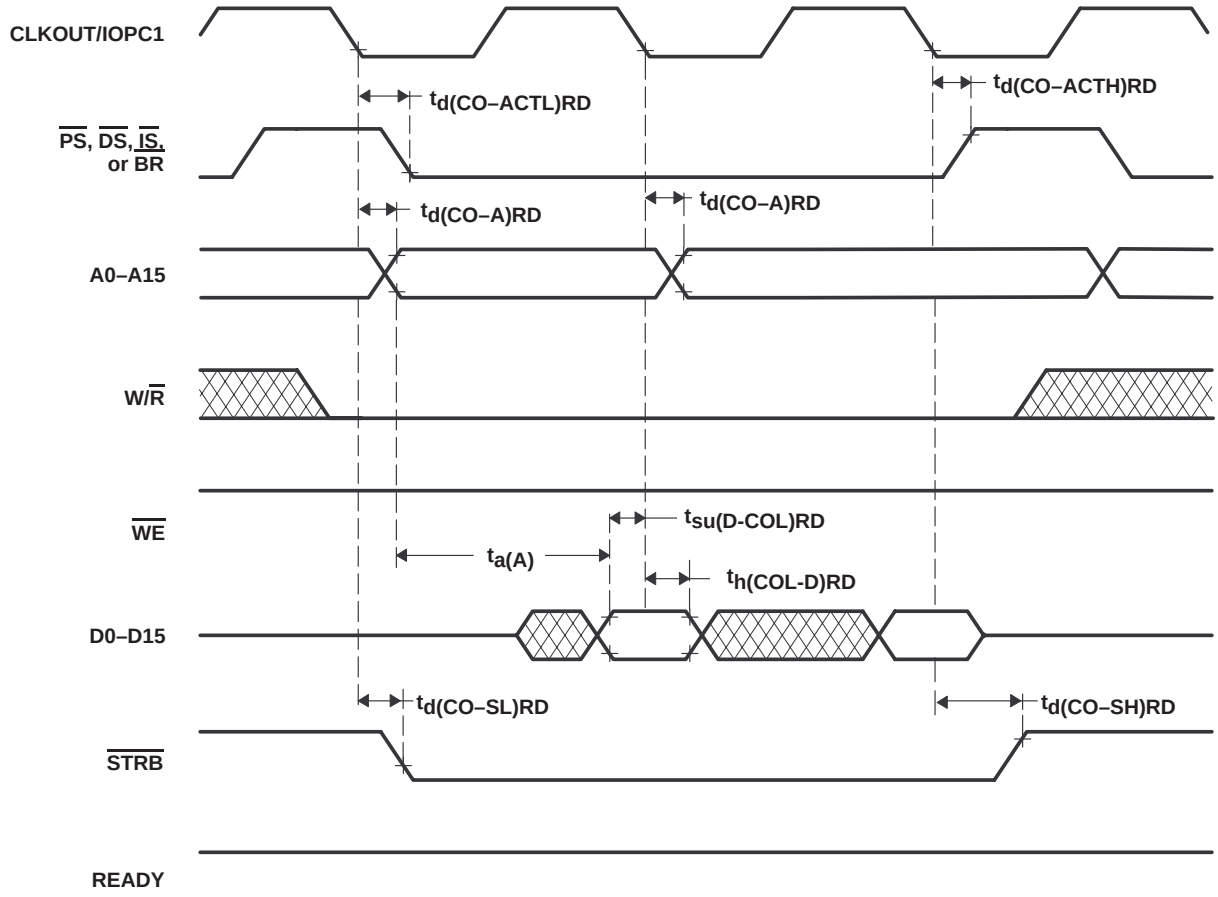


Figure 26. Memory Interface *Read* Timings

memory and parallel I/O interface *write* timings

switching characteristics over recommended operating conditions for a memory write H = $0.5t_{c(CO)}$† (see Figure 27)

PARAMETER		MIN	MAX	UNIT
$t_{d(CO-A)W}$	Delay time, CLKOUT/IOPC1 high to address valid		17	ns
$t_{d(CO-D)}$	Delay time, CLKOUT/IOPC1 low to data bus driven		15	ns
$t_{d(D-WH)}$	Delay time, address valid after $\overline{WE}$ high	H – 8		ns
$t_{w(WH)}$	Pulse duration, $\overline{WE}$ high	2H – 11		ns
$t_{w(WL)}$	Pulse duration, $\overline{WE}$ low	2H – 11		
$t_{d(CO-WL)}$	Delay time, CLKOUT/IOPC1 low to $\overline{WE}$ low		9	ns
$t_{d(CO-WH)}$	Delay time, CLKOUT/IOPC1 low to $\overline{WE}$ high		9	ns
$t_{d(WH-D)}$	Delay time, write data valid before $\overline{WE}$ high	2H – 8		ns
$t_{d(D-WHZ)}$	Delay time, $\overline{WE}$ high to data bus Hi-Z	0	5	ns
$t_{d(CO-SL)W}$	Delay time, CLKOUT/IOPC1 low to $\overline{STRB}$ low		10	ns
$t_{d(CO-SH)W}$	Delay time, CLKOUT/IOPC1 low to $\overline{STRB}$ high		6	ns
$t_{d(CO-ACTL)W}$	Delay time, CLKOUT/IOPC1 high to $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, and $\overline{BR}$ low		10	ns
$t_{d(CO-ACTH)W}$	Delay time, CLKOUT/IOPC1 high to $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, and $\overline{BR}$ high		10	ns
$t_{d(CO-RWL)}$	Delay time, CLKOUT/IOPC1 high to R/$\overline{W}$ low		10	ns
$t_{d(CO-RWH)}$	Delay time, CLKOUT/IOPC1 high to R/$\overline{W}$ high		10	ns

† All timings with respect to CLKOUT/IOPC1 assume CLKSRC[1:0] bits are set to select CPUCLK for output.

memory and parallel I/O interface *write* timings (continued)

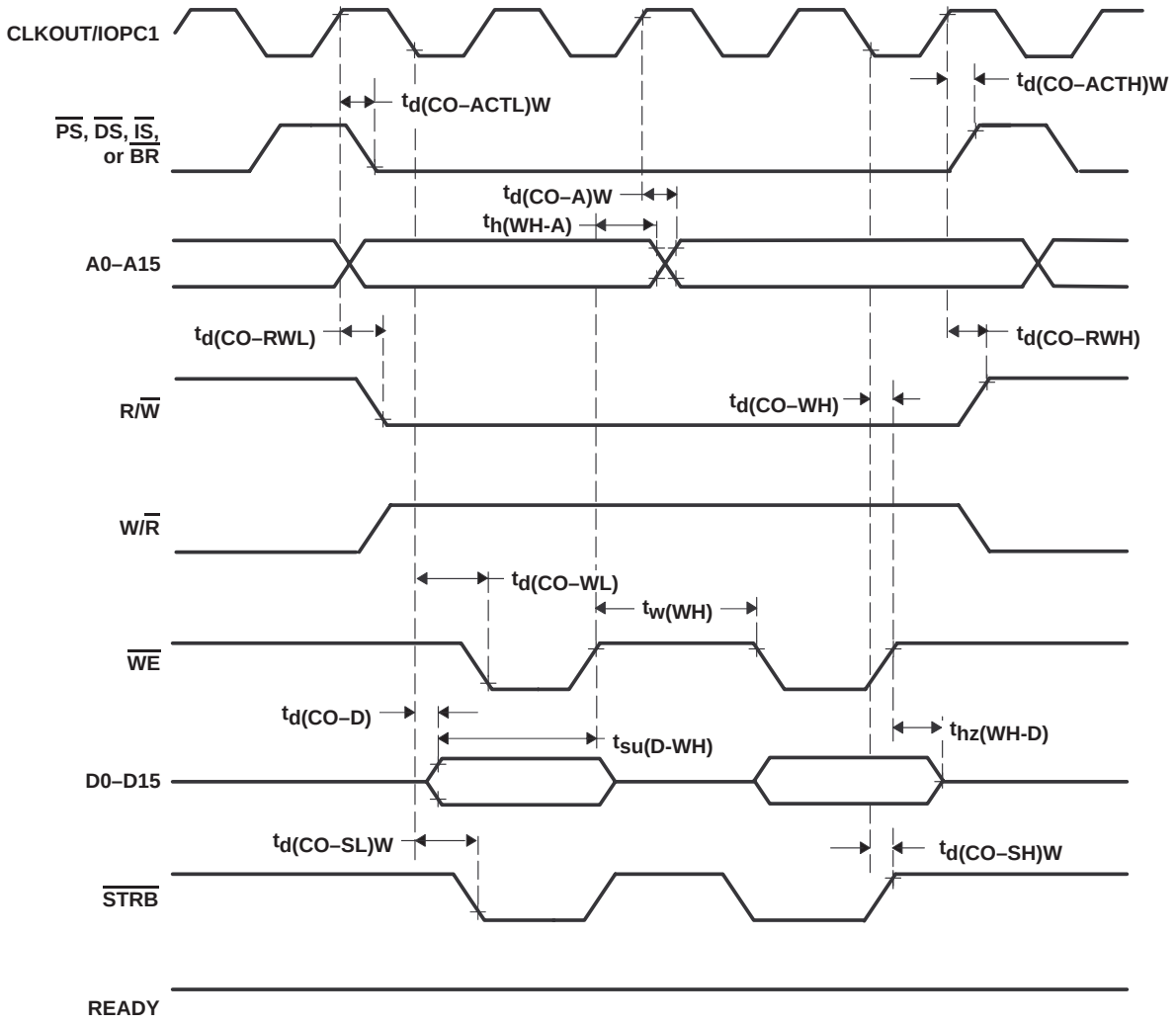


Figure 27. Memory Interface *Write* Timings

I/O timing variation with load capacitance: SPICE simulation results

Condition: **Temperature : – 40 to 150° C**
Capacitance : 5 – 125pF
Voltage : 5.0 V

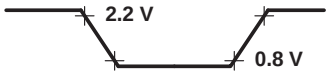


Figure 28. Rise and Fall Time Diagram

Table 17. Timing Variation With Load Capacitance, V_{DD} = 5 V, V_{OH} = 2.2 V, V_{OL} = 0.8 V

	– 40°C		27°C		150°C	
	RISE	FALL	RISE	FALL	RISE	FALL
5 pF	2.5 ns	3.6 ns	3.1 ns	4.5 ns	4.3 ns	6.2 ns
25 pF	3.1 ns	4.6 ns	4.0 ns	5.7 ns	5.6 ns	7.8 ns
50 pF	3.9 ns	5.9 ns	5.0 ns	7.3 ns	7.2 ns	9.9 ns
75 pF	4.7 ns	7.3 ns	6.1 ns	8.9 ns	8.8 ns	11.7 ns
100 pF	5.4 ns	8.9 ns	7.2 ns	10.6 ns	10.5 ns	13.8 ns
125 pF	6.2 ns	10.4 ns	8.3 ns	12.2 ns	12.1 ns	15.8 ns

READY timings

timing requirements† (see Figure 29)

		MIN	MAX	UNIT
$t_{su(R-CO)}$	Setup time, READY low before CLKOUT/IOPC1 high	14 16‡		ns
$t_{h(CO-R)}$	Hold time, READY low after CLKOUT/IOPC1 high	0		ns
$t_{v(R)ARD}$	Valid time, READY after address valid on read		3H – 31 3H – 33§	ns
$t_{v(R)AW}$	Valid time, READY after address valid on write		4H – 31 4H – 33§	ns

† The READY timings are based on one software wait state. At full speed operation, the 'F240 does not allow for single READY-based wait states.
‡ MIN value for 'C240 only
§ MAX values for 'C240 only

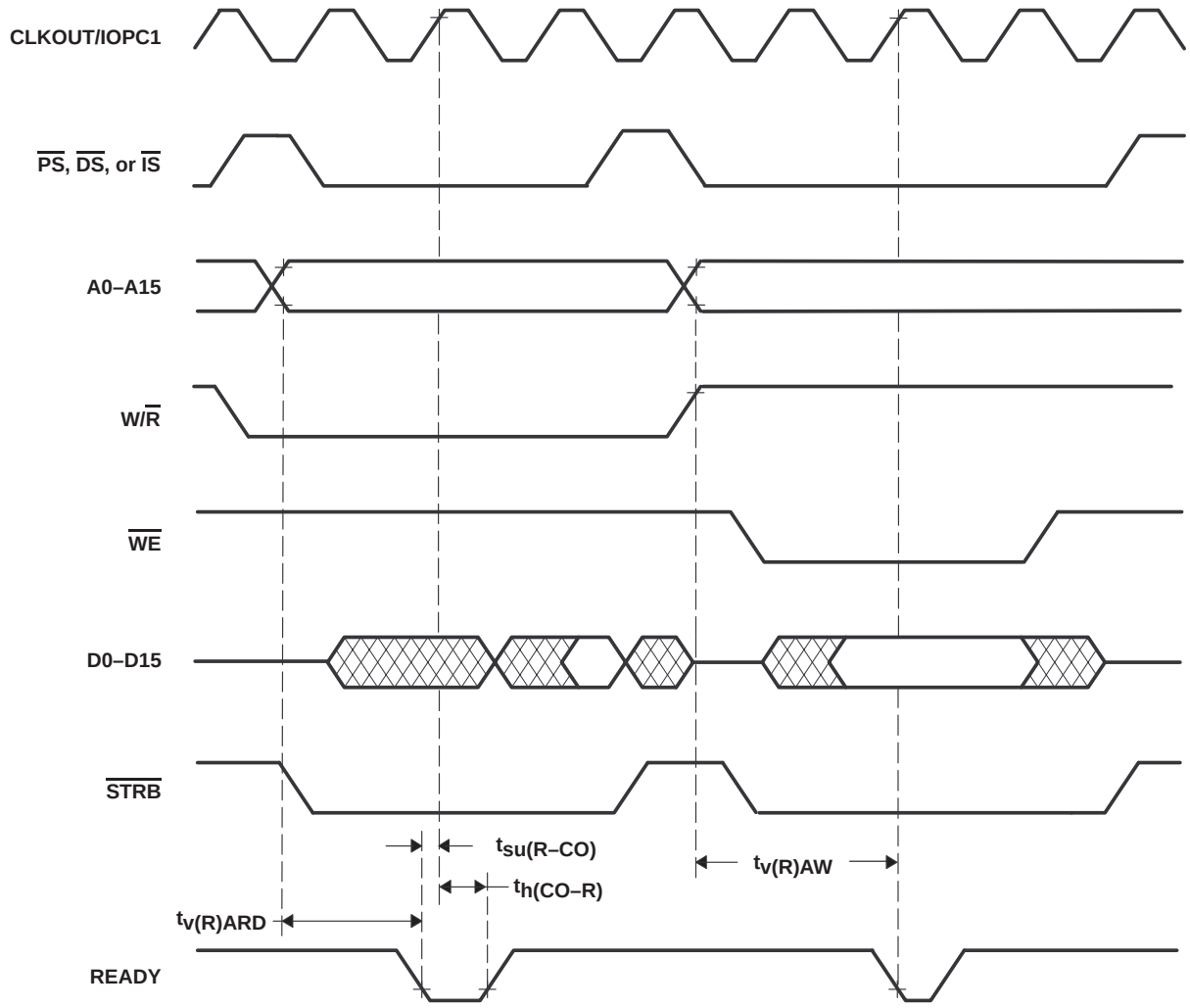


Figure 29. READY Timings

$\overline{RS}$ and $\overline{PORESET}$ timings

switching characteristics over recommended operating conditions for a reset, H = 0.5$t_{c(CO)}$ (see Figure 30 and Figure 31)

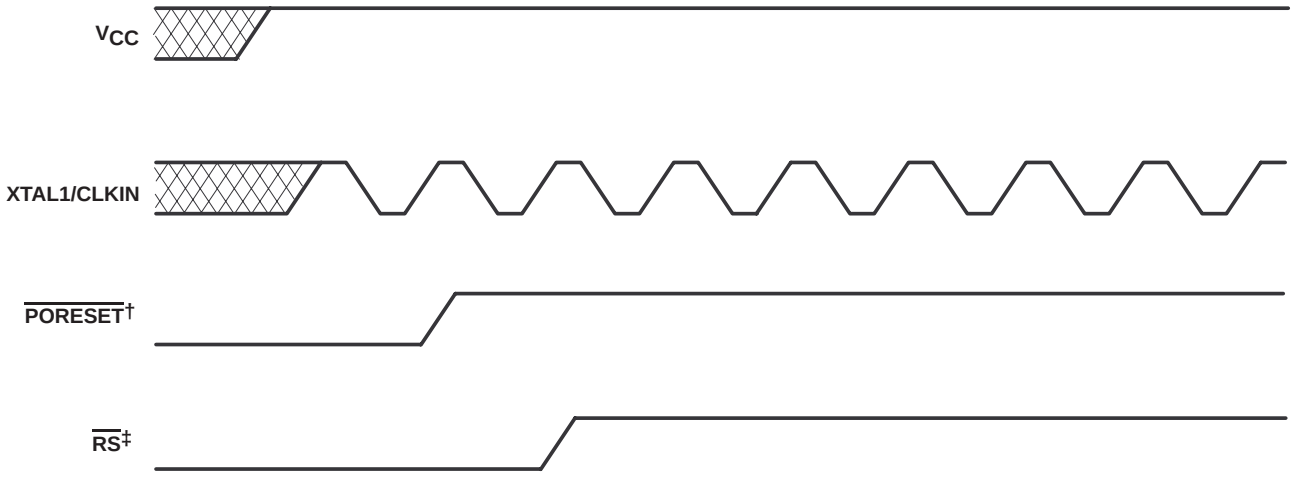
PARAMETER		MIN	MAX	UNIT
$t_{w(RSL1)}$	Pulse duration, $\overline{RS}$ low†	$8t_{c(SYS)}$		ns
$t_{d(RS)}$	Delay time, $\overline{RS}$ low to program address at reset vector	4H		ns
$t_{d(EX)}$	Delay time, $\overline{RS}$ high to reset vector executed	32H		ns

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

timing requirements for reset (see Figure 30 and Figure 31)

		MIN	MAX	UNIT
$t_{w(RSL)}$	Pulse duration, $\overline{RS}$ or $\overline{PORESET}$ low‡	5		ns

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

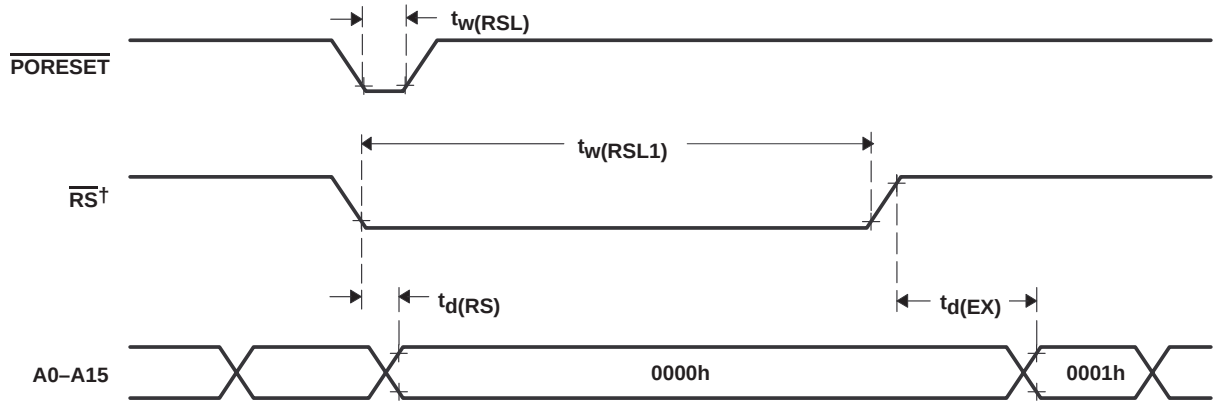


† $\overline{PORESET}$ is required to be driven low during power up to ensure all clock/PLL registers are reset to a known state.

‡ $\overline{RS}$ is a bidirectional (open-drain output) pin and can be optionally pulled low through an open-drain or open-collector drive circuit, or through a 2.7-kΩ resistor in series with a totem pole drive circuit. If $\overline{RS}$ is left undriven, then a 20-kΩ pullup resistor should be used.

Figure 30. Reset Timings

$\overline{RS}$ and $\overline{PORESET}$ timings (continued)



† $\overline{RS}$ is driven low by any device reset, which includes asserting $\overline{PORESET}$, $\overline{RS}$, access to an illegal address, execution of a software reset, or a watchdog timer reset.

Figure 31. Power-On Reset Timings

XF, $\overline{BIO}$, and MP/$\overline{MC}$ timings

switching characteristics over recommended operating conditions (see Figure 32)

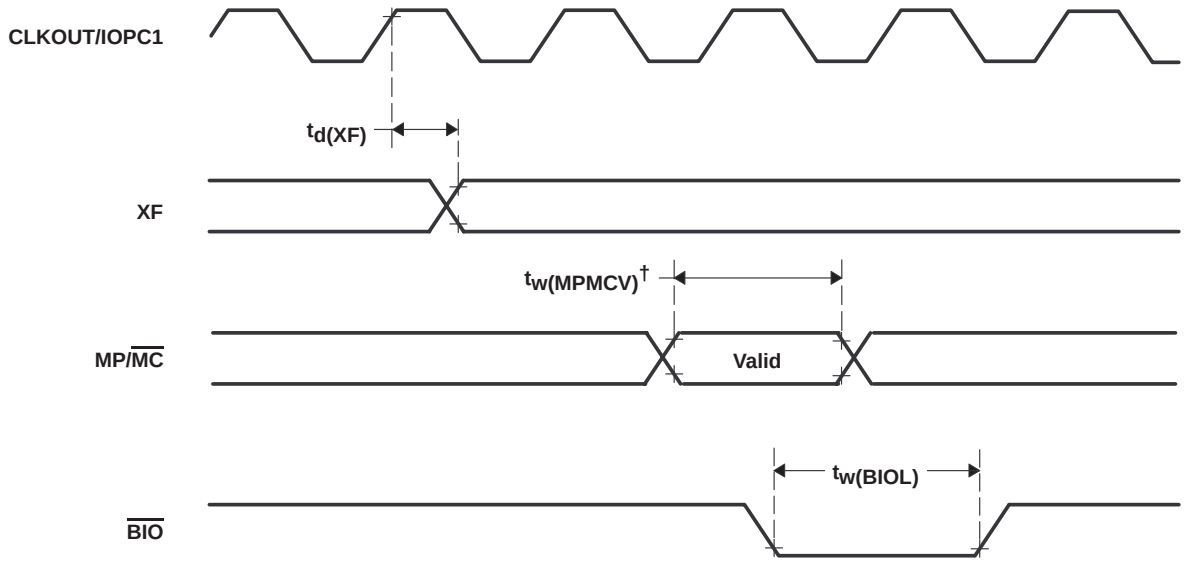
PARAMETER		MIN	MAX	UNIT
$t_{d(XF)}$	Delay time, CLKOUT high to XF high/low		11	ns

timing requirements, H = 0.5$t_{c(CO)}$ (see Figure 32)

		MIN	MAX	UNIT
$t_{w(BIOL)}$	Pulse duration, $\overline{BIO}$ low	2H + 16		ns
$t_{w(MPMCV)}$	Pulse duration, MP/$\overline{MC}$ valid†	2H + 24		ns

† This is the minimum time the MP/$\overline{MC}$ pin needs to be stable in order to be recognized by internal logic; however, for proper operation, the user must maintain a valid level for the duration of the entire memory access (or accesses) on- or off-chip.

XF, $\overline{\text{BIO}}$, and MP/$\overline{\text{MC}}$ timings (continued)



† This is the minimum time the MP/$\overline{\text{MC}}$ pin needs to be stable in order to be recognized by internal logic; however, for proper operation, the user must maintain a valid level for the duration of the entire memory access (or accesses) on- or off-chip.

Figure 32. XF, $\overline{\text{BIO}}$, and MP/$\overline{\text{MC}}$ Timings

PWM/CMP timings

PWM refers to PWM1/CMP1, PWM2/CMP2, PWM3/CMP3, PWM4/CMP4, PWM5/CMP5, PWM6/CMP6, T1PWM/T1CMP, T2PWM/T2CMP, T3PWM/T3CMP, PWM7/CMP7, PWM8/CMP8, and PWM9/CMP9.

switching characteristics over recommended operating conditions for PWM timing (see Figure 33)

PARAMETER		MIN	MAX	UNIT
$t_{d(PWM)CO}$	Delay time, CLKOUT high to PWM output switching		12	ns

timing requirements, [H = 0.5$t_{c(CO)}$] (see Figure 34 and Figure 35)

		MIN	MAX	UNIT
$t_{w(TMRDIR)}$	Pulse duration, TMRDIR low/high	4H + 12 4H + 14†		ns
$t_{w(TMRCLKL)}$	Pulse duration, TMRCLK low as a percentage of TMRCLK cycle time	40	60	%
$t_{w(TMRCLKH)}$	Pulse duration, TMRCLK high as a percentage of TMRCLK cycle time	40	60	%
$t_{c(TMRCLK)}$	Cycle time, TMRCLK	4 × $t_{c(CPU)}$		ns

† MIN value for ’C240 only

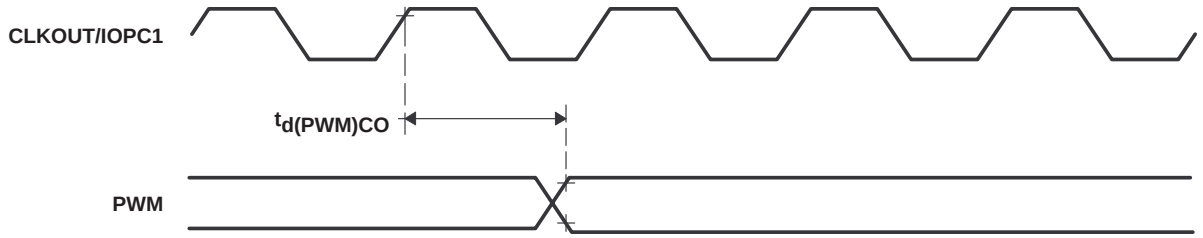


Figure 33. PWM and Compare Output Timings

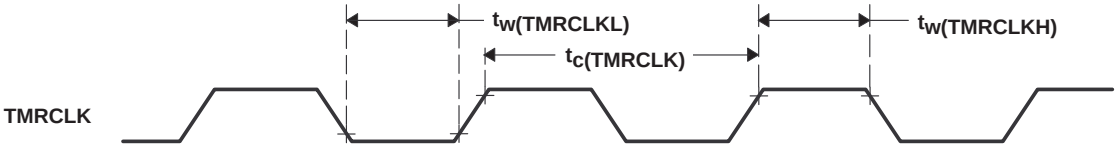


Figure 34. External Timer Clock Input Timings

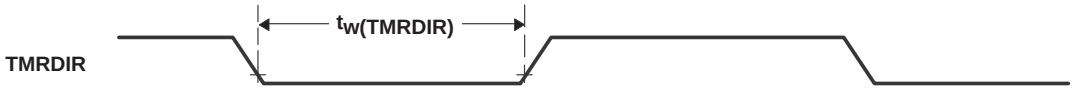


Figure 35. External Timer Direction Input Timings

capture and QEP timings

CAP refers to CAP1/QEP1/IOPC4, CAP2/QEP2/IOPC5, CAP3/IOPC6, and CAP4/IOPC7.

timing requirements, [H = 0.5$t_{c(CO)}$] (see Figure 36)

		MIN	MAX	UNIT
$t_{w(CAP)}$	Pulse duration, CAP input low/high	4H + 12 4H + 15†		ns

† MIN value for ’C240 only

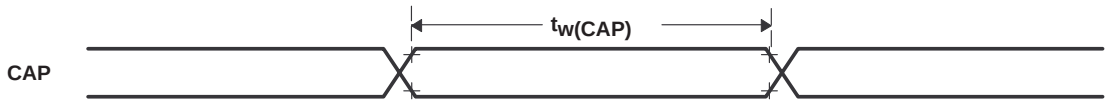


Figure 36. Capture and QEP Input Timings

interrupt timings

PWM refers to PWM1/CMP1, PWM2/CMP2, PWM3/CMP3, PWM4/CMP4, PWM5/CMP5, PWM6/CMP6, T1PWM/T1CMP, T2PWM/T2CMP, T3PWM/T3CMP, PWM7/CMP7, PWM8/CMP8, and PWM9/CMP9.

INT refers to NMI, XINT1, XINT2/IO, and XINT3/IO. PDP refers to $\overline{\text{PDPINT}}$.

switching characteristics over recommended operating conditions for interrupts (see Figure 38)

PARAMETER		MIN	MAX	UNIT
$t_{d(PWM)PDP}$	Delay time, $\overline{\text{PDPINT}}$ low to PWM to high-impedance state	0	15	ns

timing requirements, [H = 0.5$t_{c(CO)}$] (see Figure 37 and Figure 38)

		MIN	MAX	UNIT
$t_{w(INT)}$	Pulse duration, INT input low/high	$t_{c(SYS)}$ + 12		ns
$t_{w(PDP)}$	Pulse duration, $\overline{\text{PDPINT}}$ input low	2H + 18		ns
$t_{d(INT)}$	Delay time, INT low/high to interrupt-vector fetch	$2t_{c(SYS)}$ + $4t_{c(CPU)}$		ns

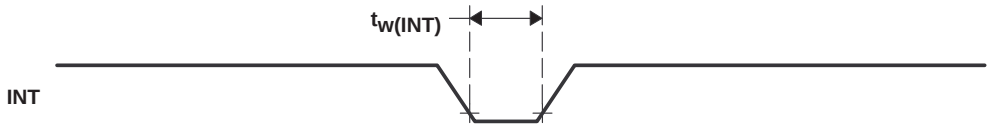


Figure 37. External Interrupt Timings

interrupt timings (continued)

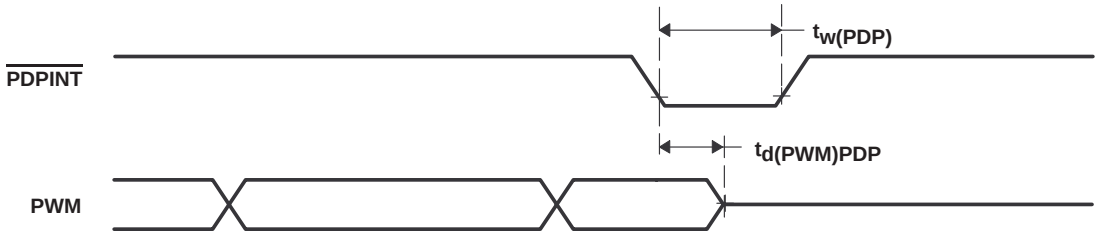


Figure 38. Power-Drive Protection Interrupt Timings

general-purpose input/output timings

GPO refers to the digital output function of shared pins IOPA0–3, IOPB0–7, IOPC0–7, XINT2/IO, XINT3/IO.

GPI refers to the digital input function of shared pins IOPA0–3, IOPB0–7, IOPC0–7, XINT2/IO, XINT3/IO.

switching characteristics over recommended operating conditions for a GPI/O (see Figure 39)

PARAMETER			MIN	MAX	UNIT
$t_{d(GPO)CO}$	Delay time, CLKOUT low to GPO low/high	XINT2/IO, XINT3/IO, IOPB6, IOPB7, and IOPC0		33	ns
		All other GPOs		25	

timing requirements (see Figure 40)

		MIN	MAX	UNIT
$t_{w(GPI)}$	Pulse duration, GPI high/low	$t_{c(SYS)}$ + 12		ns

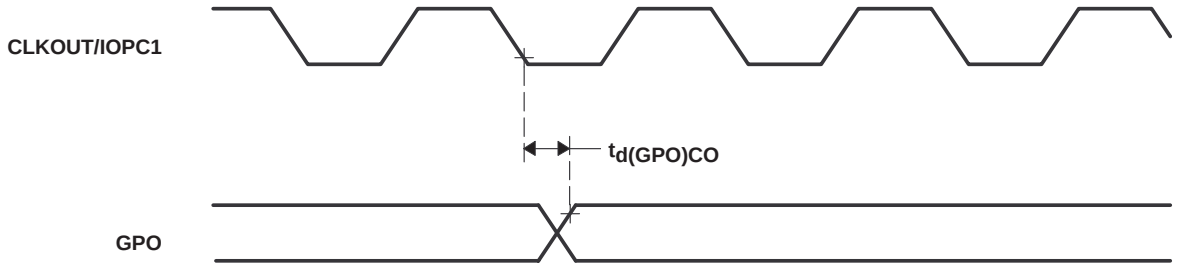


Figure 39. General-Purpose Output Timings

Figure 40. General-Purpose Input Timings

serial communications interface (SCI) I/O timings

timing characteristics for SCI (see Note 4 and Figure 41)

PARAMETER		(BRR + 1) IS EVEN AND BRR = 0		(BRR + 1) IS ODD AND BRR ≠ 0		UNIT
		MIN	MAX	MIN	MAX	
$t_{c(SCC)}$	Cycle time, SCICLK	$16t_c$	$65536t_c$	$24t_c$	$65535t_c$	ns
$t_{v(TXD)}$	Valid time, SCITXD data	$t_{c(SCC)}-70$	$t_{c(SCC)}+70$	$t_{c(SCC)}-70$	$t_{c(SCC)}+70$	ns
$t_{v(RXD)}$	Valid time, SCIRXD data	$16t_c$		$24t_c$		ns

NOTE 4: t_c = system clock cycle time = 1/SYSCLK = $t_{c(SYS)}$

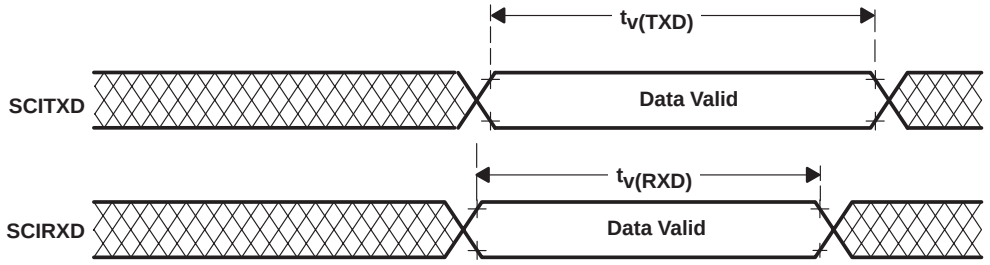


Figure 41. SCI Timings

SPI master mode timing parameters

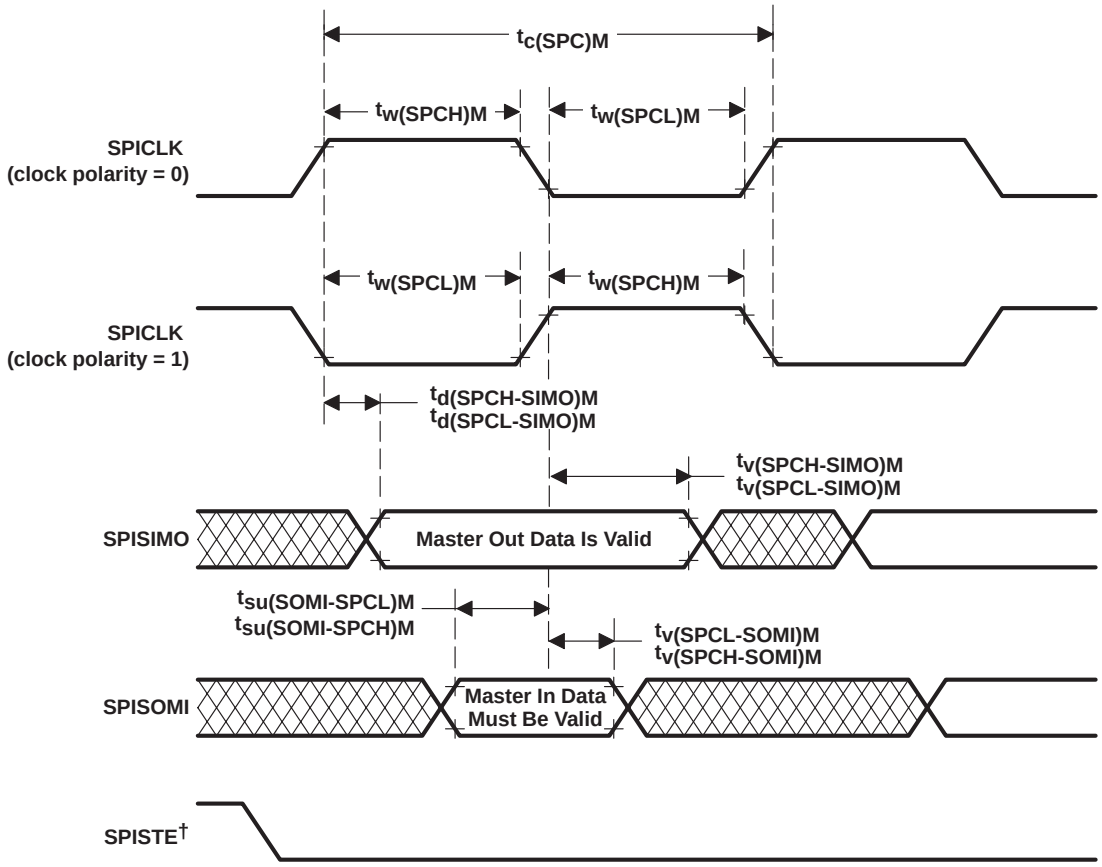
SPI master mode timing information is listed in the following table.

SPI master mode external timing parameters (clock phase = 0)† (see Figure 42)

		WHEN (SPIBRR + 1) IS EVEN OR SPIBRR = 0 OR 2		WHEN (SPIBRR + 1) IS ODD AND SPIBRR > 3		UNIT
		MIN	MAX	MIN	MAX	
$t_{c(SPC)M}$	Cycle time, SPICLK	$4t_c$‡	$128t_c$‡	$5t_c$‡	$127t_c$‡	ns
$t_{w(SPCH)M}$§	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_c-70$	$0.5t_{c(SPC)M}-0.5t_c$	ns
$t_{w(SPCL)M}$§	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_c-70$	$0.5t_{c(SPC)M}-0.5t_c$	
$t_{w(SPCL)M}$§	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_c-70$	$0.5t_{c(SPC)M}+0.5t_c$	ns
$t_{w(SPCH)M}$§	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_c-70$	$0.5t_{c(SPC)M}+0.5t_c$	
$t_{d(SPCH-SIMO)M}$§	Delay time, SPICLK high (clock polarity = 0) to SPISIMO valid	– 10	10	– 10	10	ns
$t_{d(SPCL-SIMO)M}$§	Delay time, SPICLK low (clock polarity = 1) to SPISIMO valid	– 10	10	– 10	10	
$t_{h(SPCL-SIMO)M}$§	Hold time, SPISIMO data valid after SPICLK low (clock polarity =0)	$0.5t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}+0.5t_c-70$		ns
$t_{h(SPCH-SIMO)M}$§	Hold time, SPISIMO data valid after SPICLK high (clock polarity =1)	$0.5t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}+0.5t_c-70$		
$t_{su(SOMI-SPCL)M}$§	Setup time, SPISOMI before SPICLK low (clock polarity = 0)	0		0		ns
$t_{su(SOMI-SPCH)M}$§	Setup time, SPISOMI before SPICLK high (clock polarity = 1)	0		0		
$t_{h(SPCL-SOMI)M}$§	Hold time, SPISOMI data valid after SPICLK low (clock polarity = 0)	$0.25t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-0.5t_c-70$		ns
$t_{h(SPCH-SOMI)M}$§	Hold time, SPISOMI data valid after SPICLK high (clock polarity = 1)	$0.25t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-0.5t_c-70$		

† The MASTER / SLAVE bit (SPICTL.2) is set and the CLOCK PHASE bit (SPICTL.3) is cleared.
‡ t_c = system clock cycle time = 1 / SYSCLK = $t_{c(SYS)}$
§ The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).

PARAMETER MEASUREMENT INFORMATION



† The SPISTE signal must be active before the SPI communication stream starts; the SPISTE signal must remain active until the SPI communication stream is complete.

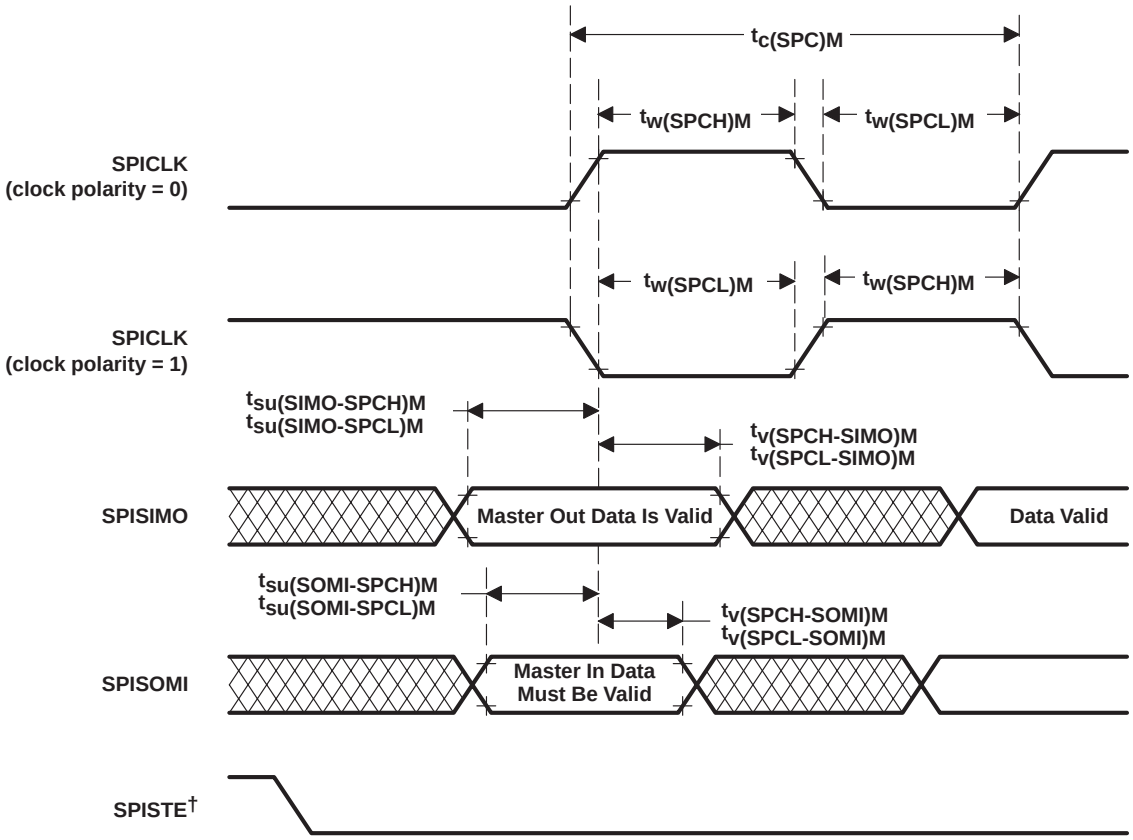
Figure 42. SPI Master Mode External Timings (Clock Phase = 0)

SPI master mode external timing (clock phase = 1)† (see Figure 43)

		WHEN (SPIBRR + 1) IS EVEN OR SPIBRR = 0 OR 2		WHEN (SPIBRR + 1) IS ODD AND SPIBRR > 3		UNIT
		MIN	MAX	MIN	MAX	
$t_{c(SPC)M}$	Cycle time, SPICLK	$4t_c$‡	$128t_c$‡	$5t_c$‡	$127t_c$‡	ns
$t_{w(SPCH)M}$§	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_c-70$	$0.5t_{c(SPC)M}-0.5t_c$	ns
$t_{w(SPCL)M}$§	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}-0.5t_c-70$	$0.5t_{c(SPC)M}-0.5t_c$	
$t_{w(SPCL)M}$§	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_c-70$	$0.5t_{c(SPC)M}+0.5t_c$	ns
$t_{w(SPCH)M}$§	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)M}-70$	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}+0.5t_c-70$	$0.5t_{c(SPC)M}+0.5t_c$	
$t_{su(SIMO-SPCH)M}$§	Setup time, SPISIMO data valid before SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-70$		ns
$t_{su(SIMO-SPCL)M}$§	Setup time, SPISIMO data valid before SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-70$		
$t_{h(SPCH-SIMO)M}$§	Hold time, SPISIMO data valid after SPICLK high (clock polarity =0)	$0.5t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-70$		ns
$t_{h(SPCL-SIMO)M}$§	Hold time, SPISIMO data valid after SPICLK low (clock polarity =1)	$0.5t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-70$		
$t_{su(SOMI-SPCH)M}$§	Setup time, SPISOMI before SPICLK high (clock polarity = 0)	0		0		ns
$t_{su(SOMI-SPCL)M}$§	Setup time, SPISOMI before SPICLK low (clock polarity = 1)	0		0		
$t_{h(SPCH-SOMI)M}$§	Hold time, SPISOMI data valid after SPICLK high (clock polarity = 0)	$0.25t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-70$		ns
$t_{h(SPCL-SOMI)M}$§	Hold time, SPISOMI data valid after SPICLK low (clock polarity = 1)	$0.25t_{c(SPC)M}-70$		$0.5t_{c(SPC)M}-70$		

† The MASTER / SLAVE bit (SPICTL.2) is set and the CLOCK PHASE bit (SPICTL.3) is set.
‡ t_c = system clock cycle time = 1 / SYSCLK = $t_{c(SYS)}$
§ The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).

PARAMETER MEASUREMENT INFORMATION



† The SPISTE signal must be active before the SPI communication stream starts; the SPISTE signal must remain active until the SPI communication stream is complete.

Figure 43. SPI Master Mode External Timings (Clock Phase = 1)

SPI slave mode timing parameters

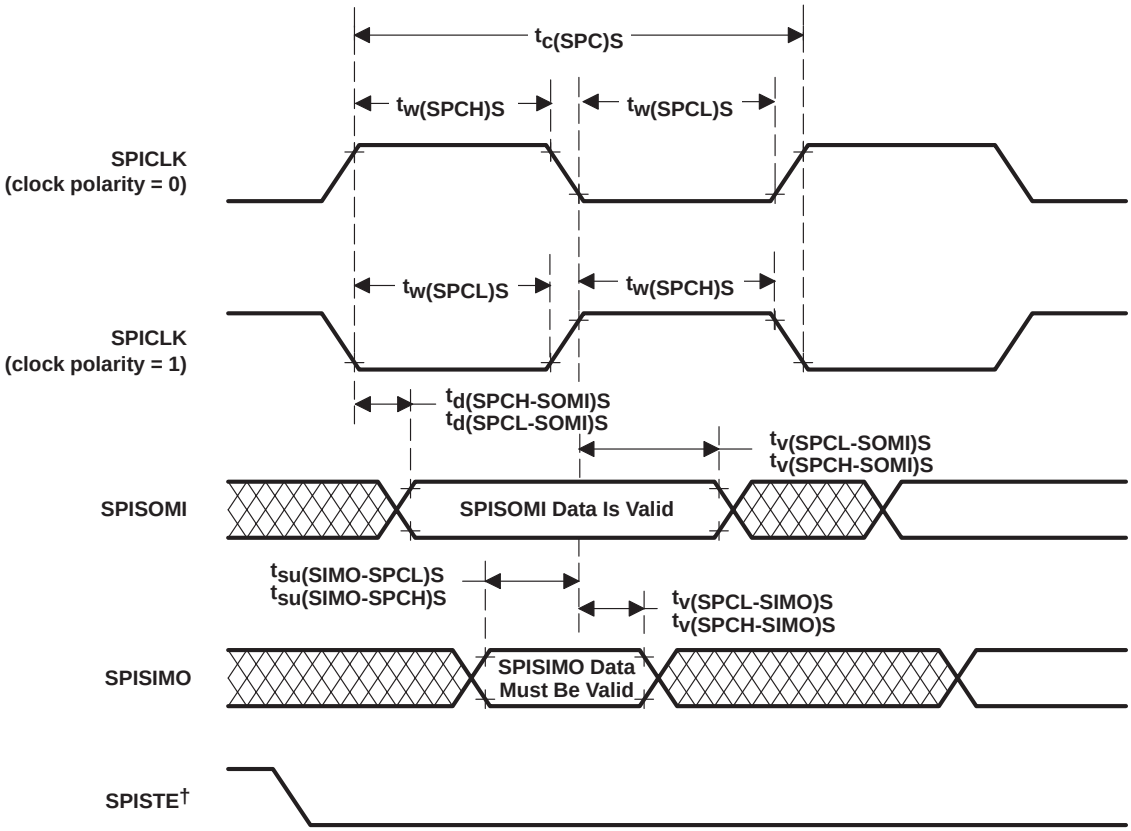
Slave mode timing information is listed in the following tables.

SPI slave mode external timing requirements (clock phase = 0)† (see Figure 44)

		MIN	MAX	UNIT
$t_{c(SPC)S}$	Cycle time, SPICLK	$8t_c$‡		ns
$t_{w(SPCH)S}$§	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	ns
$t_{w(SPCL)S}$§	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	
$t_{w(SPCL)S}$§	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	ns
$t_{w(SPCH)S}$§	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	
$t_{d(SPCH-SOMI)S}$§	Delay time, SPICLK high (clock polarity = 0) to SPISOMI valid	$0.375t_{c(SPC)S}-70$		ns
$t_{d(SPCL-SOMI)S}$§	Delay time, SPICLK low (clock polarity = 1) to SPISOMI valid	$0.375t_{c(SPC)S}-70$		
$t_{v(SPCL-SOMI)S}$§	Valid time, SPISOMI data valid after SPICLK low (clock polarity =0)	$0.75t_{c(SPC)S}$		ns
$t_{v(SPCH-SOMI)S}$§	Valid time, SPISOMI data valid after SPICLK high (clock polarity =1)	$0.75t_{c(SPC)S}$		
$t_{su(SIMO-SPCL)S}$§	Setup time, SPISIMO before SPICLK low (clock polarity = 0)	0		ns
$t_{su(SIMO-SPCH)S}$§	Setup time, SPISIMO before SPICLK high (clock polarity = 1)	0		
$t_{v(SPCL-SIMO)S}$§	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)S}$		ns
$t_{v(SPCH-SIMO)S}$§	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)S}$		

† The MASTER / SLAVE bit (SPICTL.2) is cleared and the CLOCK PHASE bit (SPICTL.3) is cleared.
‡ t_c = system clock cycle time = 1 / SYSCLK = $t_{c(SYS)}$
§ The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).

PARAMETER MEASUREMENT INFORMATION



† The SPISTE signal must be active before the SPI communication stream starts; the SPISTE signal must remain active until the SPI communication stream is complete.

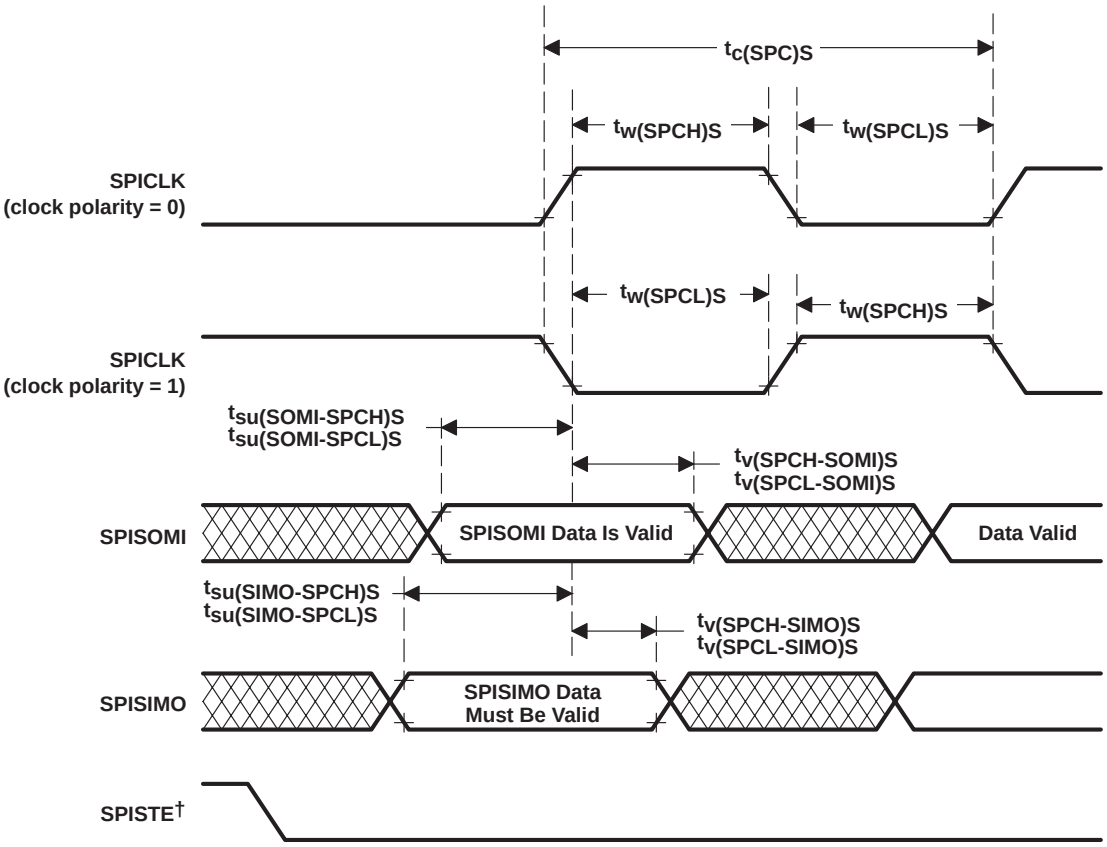
Figure 44. SPI Slave Mode External Timing (Clock Phase = 0)

SPI slave mode external timing requirements (clock phase = 1)† (see Figure 45)

		MIN	MAX	UNIT
$t_{c(SPC)S}$	Cycle time, SPICLK	$8t_c$‡		ns
$t_{w(SPCH)S}$§	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	ns
$t_{w(SPCL)S}$§	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	
$t_{w(SPCL)S}$§	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	ns
$t_{w(SPCH)S}$§	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)S}-70$	$0.5t_{c(SPC)S}$	
$t_{su(SOMI-SPCH)S}$§	Setup time, SPISOMI before SPICLK high (clock polarity = 0)	$0.125t_{c(SPC)S}$		ns
$t_{su(SOMI-SPCL)S}$§	Setup time, SPISOMI before SPICLK low (clock polarity = 1)	$0.125t_{c(SPC)S}$		
$t_{v(SPCH-SOMI)S}$§	Valid time, SPISOMI data valid after SPICLK high (clock polarity =0)	$0.75t_{c(SPC)S}$		ns
$t_{v(SPCL-SOMI)S}$§	Valid time, SPISOMI data valid after SPICLK low (clock polarity =1)	$0.75t_{c(SPC)S}$		
$t_{su(SIMO-SPCH)S}$§	Setup time, SPISIMO before SPICLK high (clock polarity = 0)	0		ns
$t_{su(SIMO-SPCL)S}$§	Setup time, SPISIMO before SPICLK low (clock polarity = 1)	0		
$t_{v(SPCH-SIMO)S}$§	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)S}$		ns
$t_{v(SPCL-SIMO)S}$§	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)S}$		

† The MASTER / SLAVE bit (SPICTL.2) is cleared and the CLOCK PHASE bit (SPICTL.3) is set.
‡ t_c = system clock cycle time = 1 / SYSCLK = $t_{c(SYS)}$
§ The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).

PARAMETER MEASUREMENT INFORMATION



† The SPISTE signal must be active before the SPI communication stream starts; the SPISTE signal must remain active until the SPI communication stream is complete.

Figure 45. SPI Slave Mode External Timing (Clock Phase = 1)

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

The 10-bit dual ADC has a separate power bus for its analog circuitry. These pins are referred to as V_{CCA} and V_{SSA}. The purpose 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.

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, ADCIN0–ADCIN15	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.

electrical characteristics (see Note 5)

PARAMETER		DESCRIPTION		MIN	MAX	UNIT
I_{CCA}	Analog supply current	V_{CCA} = 5.5 V	Converting		5	mA
			Non-converting		2	
		V_{CCA} = V_{REFHI} = 5.5 V	PLL or OSC power down		1	
I_{ref}	Input charge current, V_{REFHI} or V_{REFLO}	V_{CCA} = V_{CCD} = V_{REFHI} = 5.5 V, V_{REFLO} = 0 V			5	mA
C_{ai}	Analog input capacitance	Typical capacitive load on analog input pin	Non-sampling		6	pF
			Sampling		8	
Z_{AI}	Analog input source impedance	Analog input source impedance for conversions to remain within specifications.			9	kΩ
E_{DNL}	Differential nonlinearity error	Difference between the actual step width and the ideal value		– 1	1.5	LSB
E_{INL}	Integral nonlinearity error	Maximum deviation from the best straight line through the ADC transfer characteristics, excluding the quantization error			± 1.5	LSB
$t_{d(PU)}$	Delay time, power-up to ADC valid	Time to stabilize analog stage after power-up			10	μs

NOTE 5: 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 sizes decrease. Therefore, the absolute accuracy and differential/integral linearity errors in terms of LSBs increase.

The ADC module allows complete freedom in the design of the sources for the analog inputs. The period of the sample time is independent of the source impedance. The sample-and-hold period occurs in the first half-period of the ADC clock after the ADCIMSTART bit or the ADCSOC bit of the ADC control register 1 (ADCTRL1, bits 13 and 0, respectively) is set to 1. The conversion then occurs during the next six ADC clock cycles. The digital result registers are updated on the next ADC clock cycle once the conversion is completed.

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 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 46.

ADC input pin circuit (continued)

Figure 46. Typical ADC Input Pin Circuit

ADC timing requirements (see Figure 47)

		MIN	MAX	UNIT
$t_{c(AD)}$	Cycle time, ADC prescaled clock	1		μs
$t_{w(SHC)}$	Pulse duration, total sample/hold and conversion time (see Note 6)	6.1		μs
$t_{w(SH)}$	Pulse duration, sample and hold time	$t_{c(AD)}$		μs
$t_{su(SH)}$	Setup time, analog input stable before sample/hold start	0		ns
$t_{h(SH)}$	Hold time, analog input stable after sample/hold complete	0		ns
$t_{w(C)}$	Pulse duration, total conversion time	$4.5t_{c(AD)}$		μs
$t_{d(SOC-SH)}$	Delay time, start of conversion† to beginning of sample and hold	$3t_{c(SYS)}$		ns
$t_{d(EOC-FIFO)}$	Delay time, end of conversion to data loaded into result FIFO	$3t_{c(SYS)}$		ns

† Start of conversion is signaled by the ADCIMSTART bit or the ADCSOC bit set in software, the external start signal active (ADCSOC), or internal EVSOC signal active.

NOTE 6: The total sample/hold and conversion time is determined by the summation of $t_{d(SOC-SH)}$, $t_{w(SH)}$, $t_{w(C)}$, and $t_{d(EOC-FIFO)}$.

ADC timing requirements (continued)

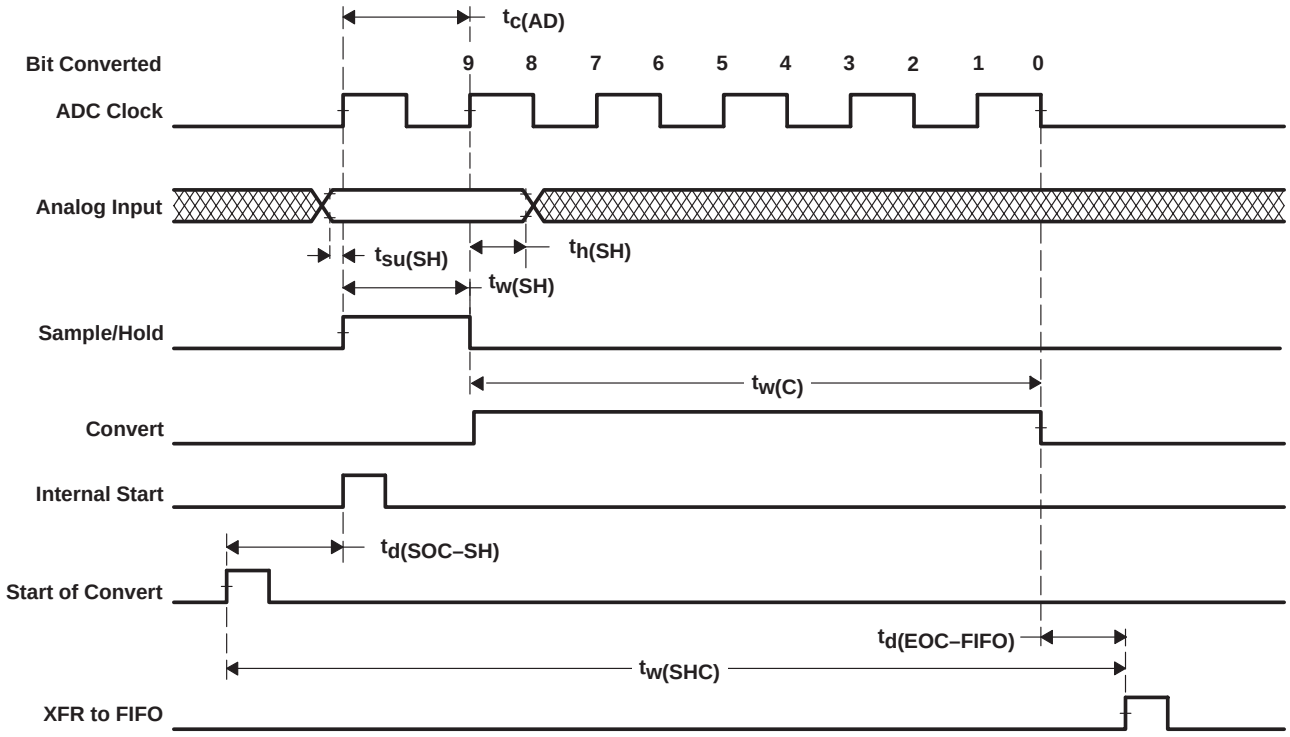


Figure 47. Analog-to-Digital Timing

flash EEPROM

switching characteristics over recommended operating conditions (see page 57)

PARAMETER		’320F240			UNIT
		MIN	TYP	MAX	
	Program-erase endurance	10K			Cycles
	Program pulses per word†	1	10	150	Pulses
	Erase pulses per array†	1	20	1000	Pulses
	Flash-write pulses per array†	1	20	6000	Pulses

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, see the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282).

timing requirements

		’320F240		UNIT
		MIN	MAX	
$t_{d(BUSY)}$	Delay time, after mode deselect to stabilization†	10		µs
$t_{d(RD-VERIFY)}$	Delay time, verify read mode select to stabilization†	10		µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, see the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282).

programming operation (maximum programming temperature 85°C for flash memory)

PARAMETER		’320F240			UNIT
		MIN	NOM	MAX	
$t_{w(PGM)}$	Pulse duration, programming algorithm†	95	100	105	µs
$t_{d(PGM-MODE)}$	Delay time, program mode select to stabilization†	10			µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, see the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282).

erase operation

PARAMETER		’320F240			UNIT
		MIN	NOM	MAX	
$t_{w(ERASE)}$	Pulse duration, erase algorithm†	6.65	7	7.35	ms
$t_{d(ERASE-MODE)}$	Delay time, erase mode select to stabilization†	10			µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, see the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282).

flash-write operation

PARAMETER		’320F240			UNIT
		MIN	NOM	MAX	
$t_{w(FLW)}$	Pulse duration, flash-write algorithm†‡	13.3	14	14.7	ms
$t_{d(FLW-MODE)}$	Delay time, flash-write mode select to stabilization†‡	10			µs

† These parameters are used in the flash programming algorithms. For a detailed description of the algorithms, see the *TMS320F20x/F24x DSPs Embedded Flash Memory Technical Reference* (literature number SPRU282).
‡ Refer to the **recommended operating conditions** section for the flash programming operating temperature range when programming flash.

register file compilation

Table 18 is a collection of all the programmable registers of the SMJ320F240 (provided for a quick reference).

Table 18. Register File Compilation

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	
	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 CONFIGURATION REGISTERS								
07018h	RESET1	RESET0	—	—	—	—	—	—	SYSCR
	CLKSRC1	CLKSRC0	—	—	—	—	—	—	
07019h	Reserved								
0701Ah	PORST	—	—	ILLADR	—	SWRST	WDRST	—	SYSSR
	—	—	HPO	—	VCCAOR	—	—	VECRD	
0701Bh to 0701Dh	Reserved								
0701Eh	0	0	0	0	0	0	0	0	SYSIVR
	D7	D6	D5	D4	D3	D2	D1	D0	
0701Fh	Reserved								
	WD/RTI CONTROL REGISTERS								
07020h	Reserved								
07021h	D7	D6	D5	D4	D3	D2	D1	D0	RTICNTR
07022h	Reserved								
07023h	D7	D6	D5	D4	D3	D2	D1	D0	WDCNTR
07024h	Reserved								
07025h	D7	D6	D5	D4	D3	D2	D1	D0	WDKEY
07026h	Reserved								
07027h	RTI FLAG	RTI ENA	—	—	—	RTIPS2	RTIPS1	RTIPS0	RTICR
07028h	Reserved								
07029h	WD FLAG	WDDIS	WDCHK2	WDCHK1	WDCHK0	WDPS2	WDPS1	WDPS0	WDCR
	PLL CLOCK CONTROL REGISTERS								
0702Ah	Reserved								
0702Bh	CLKMD(1)	CLKMD(0)	PLLOCK(1)	PLLOCK(0)	PLLPM(1)	PLLPM(0)	ACLKENA	PLLPS	CKCR0
0702Ch	Reserved								

register file compilation (continued)

Table 18. 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	
	PLL CLOCK CONTROL REGISTERS (CONTINUED)								
0702Dh	CKINF(3)	CKINF(2)	CKINF(1)	CKINF(0)	PLLDIV(2)	PLLFB(2)	PLLFB(1)	PLLFB(0)	CKCR1
0702Eh to 07031h	Reserved								
	A-to-D MODULE CONTROL REGISTERS								
07032h	SUSPEND-SOFT	SUSPEND-FREE	ADCIM-START	ADC1EN	ADC2EN	ADCCON-RUN	ADCINTEN	ADCINTFLAG	ADCTRL1
	ADCEOC	ADC2CHSEL			ADC1CHSEL			ADCSOC	
07033h	Reserved								
07034h	—	—	—	—	—	ADCEVSOC	ADCEXTSOC	—	ADCTRL2
	ADCFIFO1		—	ADCFIFO2		ADCPSCALE			
07035h	Reserved								
07036h	D9	D8	D7	D6	D5	D4	D3	D2	ADCFIFO1
	D1	D0	0	0	0	0	0	0	
07037h	Reserved								
07038h	D9	D8	D7	D6	D5	D4	D3	D2	ADCFIFO2
	D1	D0	0	0	0	0	0	0	
07039h to 0703Fh	Reserved								
	SERIAL PERIPHERAL INTERFACE (SPI) CONFIGURATION CONTROL REGISTERS								
07040h	SPI SW RESET	CLOCK POLARITY	—	—	—	SPI CHAR2	SPI CHAR1	SPI CHAR0	SPICCR
07041h	—	—	—	OVERRUN INT ENA	CLOCK PHASE	MASTER/ SLAVE	TALK	SPI INT ENA	SPICTL
07042h	RECEIVER OVERRUN	SPI INT FLAG	—	—	—	—	—	—	SPISTS
07043h	Reserved								
07044h	—	SPI BIT RATE 6	SPI BIT RATE 5	SPI BIT RATE 4	SPI BIT RATE 3	SPI BIT RATE 2	SPI BIT RATE 1	SPI BIT RATE 0	SPIBRR
07045h	Reserved								
07046h	ERCVD7	ERCVD6	ERCVD5	ERCVD4	ERCVD3	ERCVD2	ERCVD1	ERCVD0	SPIEMU
07047h	RCVD7	RCVD6	RCVD5	RCVD4	RCVD3	RCVD2	RCVD1	RCVD0	SPIBUF
07048h	Reserved								
07049h	SDAT7	SDAT6	SDAT5	SDAT4	SDAT3	SDAT2	SDAT1	SDAT0	SPIDAT
0704Ah to 0704Ch	Reserved								
0704Dh	SPISTE DATA IN	SPISTE DATA OUT	SPISTE FUNCTION	SPISTE DATA DIR	SPICLK DATA IN	SPICLK DATA OUT	SPICLK FUNCTION	SPICLK DATA DIR	SPIPC1
0704Eh	SPISIMO DATA IN	SPISIMO DATA OUT	SPISIMO FUNCTION	SPISIMO DATA DIR	SPISOMI DATA IN	SPISOMI DATA OUT	SPISOMI FUNCTION	SPISOMI DATA DIR	SPIPC2
0704Fh	—	SPI PRIORITY	SPI ESPEN	—	—	—	—	—	SPIPRI

register file compilation (continued)

Table 18. 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	
	SERIAL COMMUNICATIONS INTERFACE (SCI) CONFIGURATION CONTROL REGISTERS								
07050h	STOP BITS	EVEN/ODD PARITY	PARITY ENABLE	SCI ENA	ADDR/IDLE MODE	SCI CHAR2	SCI CHAR1	SCI CHAR0	SCICCR
07051h	—	RX ERR INT ENA	SW RESET	CLOCK ENA	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
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	Reserved								
07059h	TXDT7	TXDT6	TXDT5	TXDT4	TXDT3	TXDT2	TXDT1	TXDT0	SCITXBUF
0705Ah to 0705Dh	Reserved								
0705Eh	SCITXD DATA IN	SCITXD DATA OUT	SCITXD FUNCTION	SCITXD DATA DIR	SCIRXD DATA IN	SCIRXD DATA OUT	SCIRXD FUNCTION	SCIRXD DATA DIR	SCIPC2
0705Fh	—	SCITX PRIORITY	SCIRX PRIORITY	SCI ESPEN	—	—	—	—	SCIPRI
07060h to 0706Fh	Reserved								
	EXTERNAL INTERRUPT CONTROL REGISTERS								
07070h	XINT1 FLAG	—	—	—	—	—	—	—	XINT1CR
	—	XINT1 PIN DATA	0	—	—	XINT1 POLARITY	XINT1 PRIORITY	XINT1 ENA	
07071h	Reserved								
07072h	NMI FLAG	—	—	—	—	—	—	—	NMICR
	—	NMI PIN DATA	1	—	—	NMI POLARITY	—	—	
07073h to 07077h	Reserved								
07078h	XINT2 FLAG	—	—	—	—	—	—	—	XINT2CR
	—	XINT2 PIN DATA	—	XINT2 DATA DIR	XINT2 DATA OUT	XINT2 POLARITY	XINT2 PRIORITY	XINT2 ENA	
07079h	Reserved								

register file compilation (continued)

Table 18. 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	
	EXTERNAL INTERRUPT CONTROL REGISTERS (CONTINUED)								
0707Ah	XINT3 FLAG	—	—	—	—	—	—	—	XINT3CR
	—	XINT3 PIN DATA	—	XINT3 DATA DIR	XINT3 DATA OUT	XINT3 POLARITY	XINT3 PRIORITY	XINT3 ENA	
0707Bh to 0708Fh	Reserved								
	DIGITAL I/O CONTROL REGISTERS								
07090h	CRA.15	CRA.14	CRA.13	CRA.12	CRA.11	CRA.10	CRA.9	CRA.8	OCRA
	—	—	—	—	CRA.3	CRA.2	CRA.1	CRA.0	
07091h	Reserved								
07092h	—	—	—	—	—	—	—	—	OCRB
	CRB.7	CRB.6	CRB.5	CRB.4	CRB.3	CRB.2	CRB.1	CRB.0	
07093h to 07097h	Reserved								
07098h	—	—	—	—	A3DIR	A2DIR	A1DIR	A0DIR	PADATDIR
	—	—	—	—	IOPA3	IOPA2	IOPA1	IOPA0	
07099h	Reserved								
0709Ah	B7DIR	B6DIR	B5DIR	B4DIR	B3DIR	B2DIR	B1DIR	B0DIR	PBDATDIR
	IOPB7	IOPB6	IOPB5	IOPB4	IOPB3	IOPB2	IOPB1	IOPB0	
0709Bh	Reserved								
0709Ch	C7DIR	C6DIR	C5DIR	C4DIR	C3DIR	C2DIR	C1DIR	C0DIR	PCDATDIR
	IOPC7	IOPC6	IOPC5	IOPC4	IOPC3	IOPC2	IOPC1	IOPC0	
0709Dh to 073FFh	Reserved								
	GENERAL-PURPOSE (GP) TIMER CONFIGURATION CONTROL REGISTERS								
07400h	T3STAT	T2STAT	T1STAT	T3TOADC		T2TOADC		T1TOADC(1)	GPTCON
	T1TOADC(0)	TCOMPOE	T3PIN		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	TMODE2	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	

register file compilation (continued)

Table 18. 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	
	GENERAL-PURPOSE (GP) TIMER CONFIGURATION CONTROL REGISTERS (CONTINUED)								
07407h	D15	D14	D13	D12	D11	D10	D9	D8	T2PR
	D7	D6	D5	D4	D3	D2	D1	D0	
07408h	FREE	SOFT	TMODE2	TMODE1	TMODE0	TPS2	TPS1	TPS0	T2CON
	TSWT1	TENABLE	TCLKS1	TCLKS0	TCLD1	TCLD0	TECMPR	SELT1PR	
07409h	D15	D14	D13	D12	D11	D10	D9	D8	T3CNT
	D7	D6	D5	D4	D3	D2	D1	D0	
0740Ah	D15	D14	D13	D12	D11	D10	D9	D8	T3CMPR
	D7	D6	D5	D4	D3	D2	D1	D0	
0740Bh	D15	D14	D13	D12	D11	D10	D9	D8	T3PR
	D7	D6	D5	D4	D3	D2	D1	D0	
0740Ch	FREE	SOFT	TMODE2	TMODE1	TMODE0	TPS2	TPS1	TPS0	T3CON
	TSWT1	TENABLE	TCLKS1	TCLKS0	TCLD1	TCLD0	TECMPR	SELT1PR	
0740Dh to 07410h	Reserved								
	FULL AND SIMPLE COMPARE UNIT REGISTERS								
07411h	CENABLE	CLD1	CLD0	SVENABLE	ACTRLD1	ACTRLD0	FCOMPOE	SCOMPOE	COMCON
	SELTMR	SCLD1	SCLD0	SACTRLD1	SACTRLD0	SELCMP3	SELCMP2	SELCMP1	
07412h	Reserved								
07413h	SVRDIR	D2	D1	D0	CMP6ACT1	CMP6ACT0	CMP5ACT1	CMP5ACT0	ACTR
	CMP4ACT1	CMP4ACT0	CMP3ACT1	CMP3ACT0	CMP2ACT1	CMP2ACT0	CMP1ACT1	CMP1ACT0	
07414h	—	—	—	—	—	—	—	—	SACTR
	—	—	SCMP3-ACT1	SCMP3-ACT0	SCMP2-ACT1	SCMP2-ACT0	SCMP1-ACT1	SCMP1-ACT0	
07415h	DBT7	DBT6	DBT5	DBT4	DBT3	DBT2	DBT1	DBT0	DBTCON
	EDBT3	EDBT2	EDBT1	DBTPS1	DBTPS0	—	—	—	
07416h	Reserved								
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	D15	D14	D13	D12	D11	D10	D9	D8	SCMPR1
	D7	D6	D5	D4	D3	D2	D1	D0	
0741Bh	D15	D14	D13	D12	D11	D10	D9	D8	SCMPR2
	D7	D6	D5	D4	D3	D2	D1	D0	
0741Ch	D15	D14	D13	D12	D11	D10	D9	D8	SCMPR3
	D7	D6	D5	D4	D3	D2	D1	D0	
0741Dh to 0741Fh	Reserved								

register file compilation (continued)

Table 18. 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**	
	CAPTURE UNIT REGISTERS								
07420h	CAPRES	CAPQEPN		CAP3EN	CAP4EN	CAP34TSEL	CAP12TSEL	CAP4TOADC	CAPCON
	CAP1EDGE		CAP2EDGE		CAP3EDGE		CAP4EDGE		
07421h	Reserved								
07422h	CAP4FIFO		CAP3FIFO		CAP2FIFO		CAP1FIFO		CAPFIFO
	CAPFIFO15	CAPFIFO14	CAPFIFO13	CAPFIFO12	CAPFIFO11	CAPFIFO10	CAPFIFO9	CAPFIFO8	
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	D15	D14	D13	D12	D11	D10	D9	D8	CAP4FIFO
	D7	D6	D5	D4	D3	D2	D1	D0	
07427h to 0742Bh	Reserved								
	EVENT MANAGER (EV) INTERRUPT CONTROL REGISTERS								
0742Ch	—	—	—	—	—	T1OFINT ENA	T1UFINT ENA	T1CINT ENA	EVIMRA
	T1PINT ENA	SCMP3INT ENA	SCMP2INT ENA	SCMP1INT ENA	CMP3INT ENA	CMP2INT ENA	CMP1INT ENA	PDPINT ENA	
0742Dh	—	—	—	—	—	—	—	—	EVIMRB
	T3OFINT ENA	T3UFINT ENA	T3CINT ENA	T3PINT ENA	T2OFINT ENA	T2UFINT ENA	T2CINT ENA	T2PINT ENA	
0742Eh	—	—	—	—	—	—	—	—	EVIMRC
	—	—	—	—	CAP4INT ENA	CAP3INT ENA	CAP2INT ENA	CAP1INT ENA	
0742Fh	—	—	—	—	—	T1OFINT FLAG	T1UFINT FLAG	T1CINT FLAG	EVIFRA
	T1PINT FLAG	SCMP3INT FLAG	SCMP2INT FLAG	SCMP1INT FLAG	CMP3INT FLAG	CMP2INT FLAG	CMP1INT FLAG	PDPINT FLAG	
07430h	—	—	—	—	—	—	—	—	EVIFRB
	T3OFINT FLAG	T3UFINT FLAG	T3CINT FLAG	T3PINT FLAG	T2OFINT FLAG	T2UFINT FLAG	T2CINT FLAG	T2PINT FLAG	
07431h	—	—	—	—	—	—	—	—	EVIFRC
	—	—	—	—	CAP4INT FLAG	CAP3INT FLAG	CAP2INT FLAG	CAP1INT FLAG	
07432h	0	0	0	0	0	0	0	0	EVIVRA
	0	0	D5	D4	D3	D2	D1	D0	
07433h	0	0	0	0	0	0	0	0	EVIVRB
	0	0	D5	D4	D3	D2	D1	D0	
07434h	0	0	0	0	0	0	0	0	EVIVRC
	0	0	D5	D4	D3	D2	D1	D0	
07435h to 0743Fh	Reserved								

register file compilation (continued)

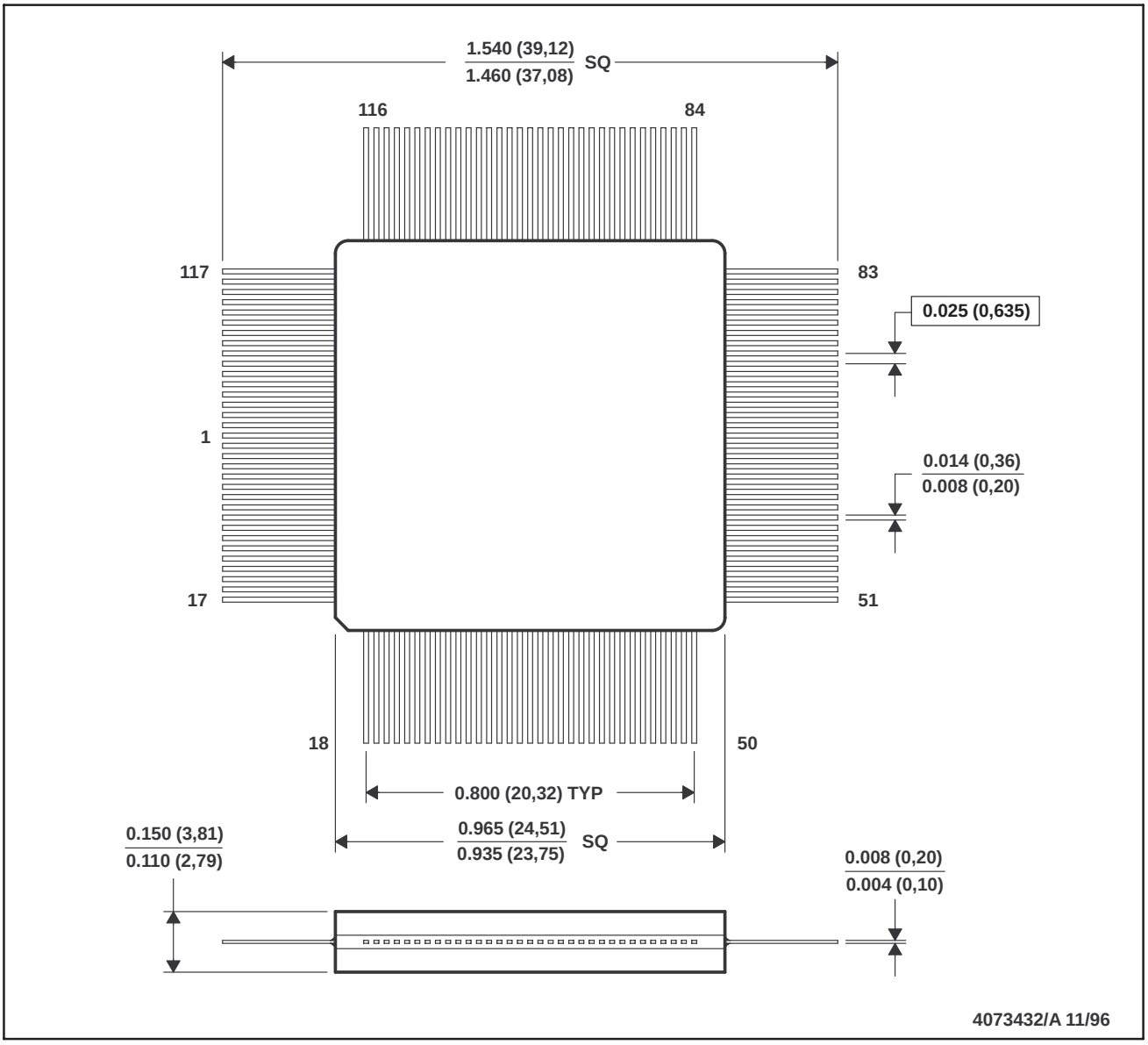
Table 18. Register File Compilation (Continued)

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	
	I/O MEMORY SPACE								
	FLASH CONTROL MODE REGISTER†								
0FF0Fh	—	—	—	—	—	—	—	—	FCMR
	—	—	—	—	—	—	—	—	
	WAIT-STATE GENERATOR CONTROL REGISTER								
0FFFFh	—	—	—	—	—	—	—	—	WSGR
	—	—	—	—	AVIS	ISWS	DSWS	PSWS	

† See the **flash control mode register** section.

MECHANICAL DATA

HFP (S-GQFP-F132) **CERAMIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. This package can be hermetically sealed with a ceramic lid using glass frit.

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

TI warrants performance of its semiconductor products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.