TMS320F28335, TMS320F28334, TMS320F28332
Digital Signal Controllers (DSCs)

Data Manual

Literature Number: SPRS439B
June 2007–Revised October 2007

ADVANCE INFORMATION concerns new products in the sampling or preproduction phase of development. Characteristic data and other specifications are subject to change without notice.

TEXAS INSTRUMENTS

Contents

List of Figures

List of Tables

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

The table lists the technical changes made for this revision.

Changes Made in Revision B

Location	Additions, Deletions, Modifications
Global	Changed 1.8 V to 1.9 V
Section 2.2	Modified type on the MCLKXB option of GPIO26 in the Signal Descriptions table
Section 3.1	Added bullets at the beginning of the Memory Maps section
Figure 3-2 – Figure 3-4	Modified all three memory map figures
Table 3-5	Modified Wait-states table
Section 4.7.3	Modified the ADC Calibration section
Section 4.8	Modified clock rate equation in the McBSP Module section
Figure 4-18	Added note to GPIO MUX Block Diagram
Table 4-16	Modified GPIO-B Mux Peripheral Selection Matrix
Figure 5-1	Modified device nomenclature example figure
Section 6.2	Modified clock frequency in Recommended Operating Conditions table
Table 6-5	Modified the LSPCLK values in the Clocking and Nomenclature (150-MHz devices) table
Table 6-6	Modified the HSPCLK value in the Clocking and Nomenclature (100-MHz devices) table
Table 6-53	Modified the Current Consumption for Different ADC Configurations table
Table 6-54	Modified the values in Sequential Sampling Mode Timing table
Table 6-55	Modified the values in Simultaneous Sampling Mode Timing table

1 TMS320F28335, TMS320F28334, TMS320F28332 DSCs

1.1 Features

- **High-Performance Static CMOS Technology**
  - **Up to 150 MHz (6.67-ns Cycle Time)**
  - **1.9-V Core, 3.3-V I/O Design**
- **High-Performance 32-Bit CPU (TMS320C28x)**
  - **IEEE-754 Single-Precision Floating-Point Unit (FPU)**
  - **16 x 16 and 32 x 32 MAC Operations**
  - **16 x 16 Dual MAC**
  - **Harvard Bus Architecture**
  - **Fast Interrupt Response and Processing**
  - **Unified Memory Programming Model**
  - **Code-Efficient (in C/C++ and Assembly)**
- **Six Channel DMA Controller (for ADC, McBSP, XINTF, and SARAM)**
- **16-bit or 32-bit External Interface (XINTF)**
  - **Over 2M x 16 Address Reach**
- **On-Chip Memory**
  - **F28335: 256K x 16 Flash, 34K x 16 SARAM**
  - **F28334:128K x 16 Flash, 34K x 16 SARAM**
  - **F28332: 64K x 16 Flash, 26K x 16 SARAM**
  - **1K x 16 OTP ROM**
- **Boot ROM (8K x 16)**
  - **With Software Boot Modes (via SCI, SPI, CAN, I2C, McBSP, XINTF, and Parallel I/O)**
  - **Standard Math Tables**
- **Clock and System Control**
  - **Dynamic PLL Ratio Changes Supported**
  - **On-Chip Oscillator**
  - **Watchdog Timer Module**
- **GPIO0 to GPIO63 Pins Can Be Connected to One of the Eight External Core Interrupts**
- **Peripheral Interrupt Expansion (PIE) Block That Supports All 58 Peripheral Interrupts**
- **128-Bit Security Key/Lock**
  - **Protects Flash/OTP/RAM Blocks**
  - **Prevents Firmware Reverse Engineering**
- **Enhanced Control Peripherals**
  - **Up to 18 PWM Outputs**
  - **Up to 6 HRPWM Outputs With 150 ps MEP Resolution**
  - **Up to 6 Event Capture Inputs**
  - **Up to 2 Quadrature Encoder Interfaces**
  - **Up to 8 32-bit/Six 16-bit Timers**
- **Three 32-Bit CPU Timers**
- **Serial Port Peripherals**
  - **Up to 2 CAN Modules**
  - **Up to 3 SCI (UART) Modules**
  - **Up to 2 McBSP Modules (Configurable as SPI)**
  - **One SPI Module**
  - **One Inter-Integrated-Circuit (I2C) Bus**
- **12-Bit ADC, 16 Channels**
  - **80-ns Conversion Rate**
  - **2 x 8 Channel Input Multiplexer**
  - **Two Sample-and-Hold**
  - **Single/Simultaneous Conversions**
  - **Internal or External Reference**
- **Up to 88 Individually Programmable, Multiplexed GPIO Pins With Input Filtering**
- **JTAG Boundary Scan Support** (1)
- **Advanced Emulation Features**
  - **Analysis and Breakpoint Functions**
  - **Real-Time Debug via Hardware**
- **Development Support Includes**
  - **ANSI C/C++ Compiler/Assembler/Linker**
  - **Code Composer Studio™ IDE**
  - **DSP/BIOS™**
  - **Digital Motor Control and Digital Power Software Libraries**
- **Low-Power Modes and Power Savings**
  - **IDLE, STANDBY, HALT Modes Supported**
  - **Disable Individual Peripheral Clocks**
- **Package Options**
  - **Lead-free Green Packaging**
  - **Thin Quad Flatpack (PGF)**
  - **MicroStar BGA™ (ZHH)**
  - **Plastic BGA (ZJZ)**
- **Temperature Options:**
  - **A: –40°C to 85°C (PGF, ZHH, ZJZ)**
  - **S: –40°C to 125°C (ZJZ)**

(1) IEEE Standard 1149.1-1990 Standard Test Access Port and Boundary Scan Architecture

ADVANCE INFORMATION

1.2 Getting Started

This section gives a brief overview of the steps to take when first developing for a C28x device. For more detail on each of these steps, see the following:

- *Getting Started With TMS320C28x™ Digital Signal Controllers* (literature number SPRAAM0).
- C2000 Getting Started Website (http://www.ti.com/c2000getstarted)

2 Introduction

The TMS320F28335, TMS320F28334, and TMS320F28332, devices, members of the TMS320C28x™ DSC generation, are highly integrated, high-performance solutions for demanding control applications.

Throughout this document, TMS320F28335, TMS320F28334, and TMS320F28332, are abbreviated as F28335, F28334, and F28332, respectively. Table 2-1 provides a summary of features for each device.

Table 2-1. Hardware Features

FEATURE		F28335 (150 MHz)	F28334 (150 MHz)	F28332 (100 MHz)
Instruction cycle		6.67 ns	6.67 ns	10 ns
Floating-point Unit		Yes	Yes	Yes
3.3-V on-chip flash (16-bit word)		256K	128K	64K
Single-access RAM (SARAM) (16-bit word)		34K	34K	26K
One-time programmable (OTP) ROM (16-bit word)		1K	1K	1K
Code security for on-chip flash/SARAM/OTP blocks		Yes	Yes	Yes
Boot ROM (8K X16)		Yes	Yes	Yes
16/32-bit External Interface (XINTF)		Yes	Yes	Yes
6-channel Direct Memory Access (DMA)		Yes	Yes	Yes
PWM outputs		ePWM1/2/3/4/5/6	ePWM1/2/3/4/5/6	ePWM1/2/3/4/5/6
HRPWM channels		ePWM1A/2A/3A/4A/5A/6A	ePWM1A/2A/3A/4A/5A/6A	ePWM1A/2A/3A/4A
32-bit Capture inputs or auxiliary PWM outputs		6	6	4
32-bit QEP channels (four inputs/channel)		2	2	2
Watchdog timer		Yes	Yes	Yes
12-Bit ADC	No. of channels	16	16	16
	MSPS	12.5	12.5	12.5
	Conversion time	80 ns	80 ns	80 ns
32-Bit CPU timers		3	3	3
Multichannel Buffered Serial Port (McBSP)/SPI		2	2	1
Serial Peripheral Interface (SPI)		1	1	1
Serial Communications Interface (SCI)		3	3	2
Enhanced Controller Area Network (eCAN)		2	2	2
Inter-Integrated Circuit (I2C)		1	1	1
General Purpose I/O pins (shared)		88	88	88
External interrupts		8	8	8
Packaging	176-Pin PGF	Yes	Yes	Yes
	179-Ball ZHH	Yes	Yes	Yes
	176-Ball ZJZ	Yes	Yes	Yes
Temperature options	A: –40°C to 85°C	(PGF, ZHH, ZJZ)	(PGF, ZHH, ZJZ)	(PGF, ZHH, ZJZ)
	S: –40°C to 125°C	(ZJZ)	(ZJZ)	(ZJZ)
Product status		TMX	TMX	TMX

2.1 Pin Assignments

The 176-pin PZ low-profile quad flatpack (LQFP) pin assignments are shown in Figure 2-1. The 179-ball ZHH ball grid array (BGA) terminal assignments are shown in Figure 2-2 through Figure 2-5. The 176-ball ZJZ plastic ball grid array (PBGA) terminal assignments are shown in Figure 2-6 through Figure 2-9.Table 2-2 describes the function(s) of each pin.

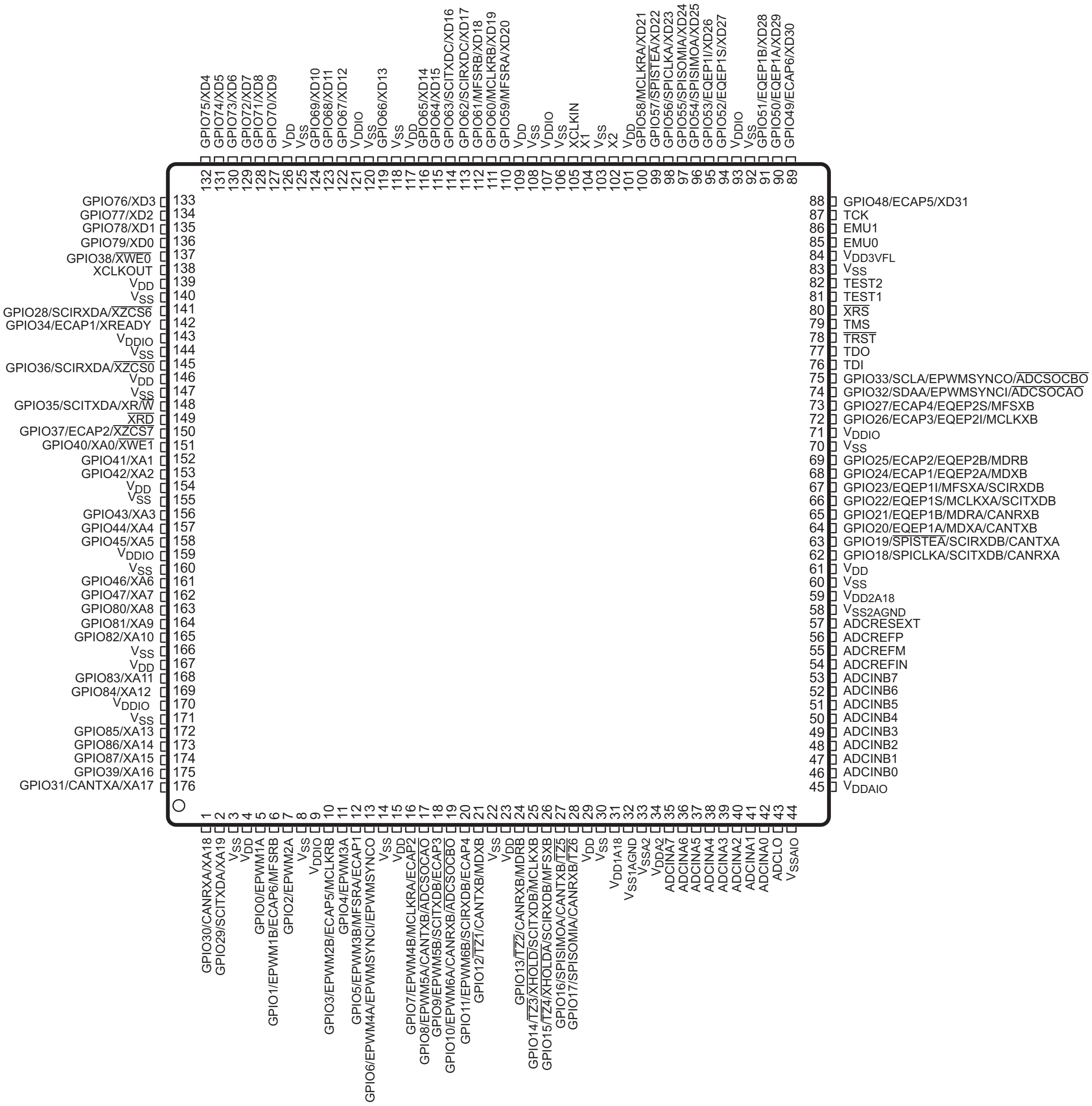


Figure 2-1. F28335, F28334, F28332 176-Pin PGF LQFP (Top View)

	1	2	3	4	5	6	7	
P	V_{SSAIO}	ADCINB0	ADCINB2	ADCINB6	ADCREFP	V_{SS}	GPIO21/ EQEP1B/ MDRA/ CANRXB	P
N	ADCINA1	V_{DDAIO}	ADCINB1	ADCINB5	ADCREFM	V_{DD}	GPIO22/ EQEP1S/ MCLKXA/ SCITXDB	N
M	ADCINA2	ADCLO	ADCINA0	ADCINB4	ADCRESEXT	V_{DD2A18}	GPIO23/ EQEP1I/ MFSXA/ SCIRXDB	M
L	ADCINA5	ADCINA4	ADCINA3	ADCINB3	ADCREFIN	GPIO18/ SPICLKA/ SCITXDB/ CANRXA	GPIO20/ EQEP1A/ MDXA/ CANTXB	L
K	$V_{SS1AGND}$	V_{DDA2}	V_{SSA2}	ADCINA7	ADCINB7	$V_{SS2AGND}$	GPIO19/ $\overline{\text{SPISTEA}}$/ SCIRXDB/ CANTXA	K
						6	7	
J	GPIO17/ SPISOMIA/ CANRXB/ $\overline{\text{TZ6}}$	V_{DD}	V_{SS}	V_{DD1A18}	ADCINA6	J		
H	V_{DD}	GPIO14/ $\overline{\text{TZ3}}$/$\overline{\text{XHOLD}}$/ SCITXDB/ MCLKXB	GPIO13/ $\overline{\text{TZ2}}$/ CANRXB/ MDRB	GPIO15/ $\overline{\text{TZ4}}$/$\overline{\text{XHOLDA}}$/ SCIRXDB/ MFSXB	GPIO16/ SPISIMOA/ CANTXB/ $\overline{\text{TZ5}}$	H		
	1	2	3	4	5			

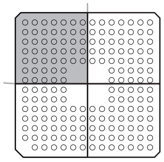


Figure 2-2. F28335, F28334, F28332 179-Ball ZHH MicroStar BGA™ (Upper Left Quadrant) (Bottom View)

ADVANCE INFORMATION

	8	9	10	11	12	13	14	
P	V_{SS}	GPIO33/ SCLA/ EPWMSYNCO/ $\overline{\text{ADCSOCBO}}$	TMS	TEST2	EMU1	GPIO48/ ECAP5/ XD31	GPIO50/ EQEP1A/ XD29	P
N	GPIO25/ ECAP2/ EQEP2B/ MDRB	GPIO32/ SDAA/ EPWMSYNCI/ $\overline{\text{ADCSOCAO}}$	V_{SS}	V_{SS}	TCK	GPIO49/ ECAP6/ XD30	V_{DDIO}	N
M	GPIO24/ ECAP1/ EQEP2A/ MDXB	TDI	$\overline{\text{TRST}}$	V_{DD3VFL}	V_{SS}	GPIO51/ EQEP1B/ XD28	GPIO52/ EQEP1S/ XD27	M
L	V_{DDIO}	GPIO27/ ECAP4/ EQEP2S/ MFSXB	$\overline{\text{XRS}}$	EMU0	GPIO53/ EQEP1I/ XD26	GPIO54/ SPISIMOA/ XD25	GPIO55/ SPISOMIA/ XD24	L
K	GPIO26/ ECAP3/ EQEP2I/ MCLKXB	TDO	TEST1	GPIO56/ SPICLKA/ XD23	GPIO58/ MCLKRA/ XD21	GPIO57/ $\overline{\text{SPISTEA}}$/ XD22	V_{DD}	K
	8	9						
J			V_{SS}	X2	V_{SS}	X1	XCLKIN	J
H			V_{SS}	V_{DDIO}	V_{DD}	V_{SS}	GPIO59/ MFSRA/ XD20	H
			10	11	12	13	14	

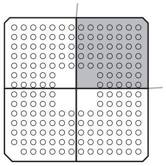
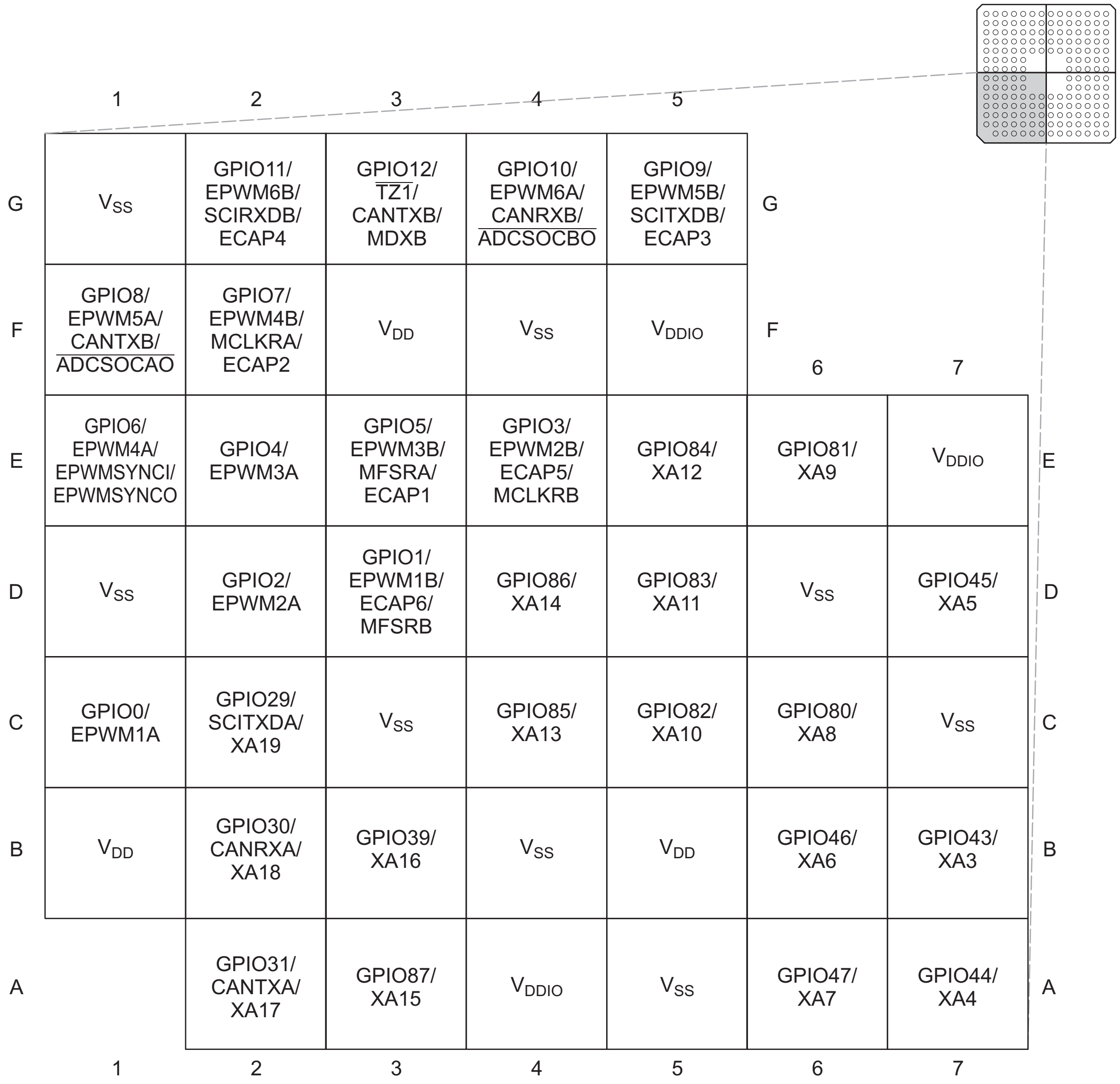


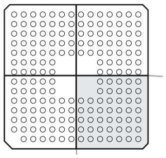
Figure 2-3. F28335, F28334, F28332 179-Ball ZHH MicroStar BGA™ (Upper Right Quadrant) (Bottom View)



	1	2	3	4	5	6	7
G	V_{SS}	GPIO11/ EPWM6B/ SCIRXDB/ ECAP4	GPIO12/ $\overline{TZ1}$/ CANTXB/ MDXB	GPIO10/ EPWM6A/ CANRXB/ $\overline{ADCSOCBO}$	GPIO9/ EPWM5B/ SCITXDB/ ECAP3		
F	GPIO8/ EPWM5A/ CANTXB/ $\overline{ADCSOCAO}$	GPIO7/ EPWM4B/ MCLKRA/ ECAP2	V_{DD}	V_{SS}	V_{DDIO}		
E	GPIO6/ EPWM4A/ EPWMSYNCI/ EPWMSYNCO	GPIO4/ EPWM3A	GPIO5/ EPWM3B/ MFSRA/ ECAP1	GPIO3/ EPWM2B/ ECAP5/ MCLKRB	GPIO84/ XA12	GPIO81/ XA9	V_{DDIO}
D	V_{SS}	GPIO2/ EPWM2A	GPIO1/ EPWM1B/ ECAP6/ MFSRB	GPIO86/ XA14	GPIO83/ XA11	V_{SS}	GPIO45/ XA5
C	GPIO0/ EPWM1A	GPIO29/ SCITXDA/ XA19	V_{SS}	GPIO85/ XA13	GPIO82/ XA10	GPIO80/ XA8	V_{SS}
B	V_{DD}	GPIO30/ CANRXA/ XA18	GPIO39/ XA16	V_{SS}	V_{DD}	GPIO46/ XA6	GPIO43/ XA3
A		GPIO31/ CANTXA/ XA17	GPIO87/ XA15	V_{DDIO}	V_{SS}	GPIO47/ XA7	GPIO44/ XA4

Figure 2-4. F28335, F28334, F28332 179-Ball ZHH MicroStar BGA™ (Lower Left Quadrant) (Bottom View)

ADVANCE INFORMATION



	8	9	10	11	12	13	14	
G			GPIO64/ XD15	GPIO63/ SCITXDC/ XD16	GPIO61/ MFSRB/ XD18	GPIO62/ SCIRXDC XD17	GPIO60/ MCLKRB/ XD19	G
F			GPIO69/ XD10	GPIO66/ XD13	V_{SS}	V_{DD}	GPIO65/ XD14	F
E	V_{SS}	V_{DD}	GPIO28/ SCIRXDA/ $\overline{\text{XZCS6}}$	GPIO68/ XD11	V_{DDIO}	GPIO67/ XD12	V_{SS}	E
D	GPIO40/ XA0/ $\overline{\text{XWE1}}$	GPIO37/ ECAP2/ $\overline{\text{XZCS7}}$	GPIO34/ ECAP1/ XREADY	GPIO38/ $\overline{\text{XWE0}}$	GPIO70/ XD9	V_{DD}	V_{SS}	D
C	V_{DD}	V_{SS}	GPIO36/ SCIRXDA/ $\overline{\text{XZCS0}}$	XCLKOUT	GPIO73/ XD6	GPIO74/ XD5	GPIO71/ XD8	C
B	GPIO42/ XA2	$\overline{\text{XRD}}$	V_{DDIO}	V_{DD}	GPIO78/ XD1	GPIO76/ XD3	GPIO72/ XD7	B
A	GPIO41/ XA1	GPIO35/ SCITXDA/ XR/$\overline{\text{W}}$	V_{SS}	V_{SS}	GPIO79/ XD0	GPIO77/ XD2	GPIO75/ XD4	A
	8	9	10	11	12	13	14	

Figure 2-5. F28335, F28334, F28332 179-Ball ZHH MicroStar BGA ™(Lower Right Quadrant) (Bottom View)

	1	2	3	4	5	6	7
P	V_{SSA2}	$V_{SS2AGND}$	ADCINB0	ADCREFM	ADCREFP	ADCRESEXT	ADCREFIN
N	V_{SSAIO}	ADCLO	ADCINB1	ADCINB3	ADCINB5	ADCINB7	EMU0
M	ADCINA2	ADCINA1	ADCINA0	ADCINB2	ADCINB4	ADCINB6	TEST1
L	ADCINA5	ADCINA4	ADCINA3	$V_{SS1AGND}$	V_{DDAIO}	V_{DD2A18}	TEST2
K	ADCINA7	ADCINA6	V_{DD1A18}	V_{DDA2}			
J	GPIO15/ $\overline{\text{TZ4}}$/$\overline{\text{XHOLDA}}$/ SCIRXDB/ MFSXB	GPIO16/ SPISIMOA/ CANTXB/ $\overline{\text{TZ5}}$	GPIO17/ SPISOMIA/ CANRXB/ $\overline{\text{TZ6}}$	V_{DD}		V_{SS}	V_{SS}
H	GPIO12/ $\overline{\text{TZ1}}$/ CANTXB/ MDXB	GPIO13/ $\overline{\text{TZ2}}$/ CANRXB/ MDRB	GPIO14/ $\overline{\text{TZ3}}$/$\overline{\text{XHOLD}}$/ SCITXDB/ MCLKXB	V_{DD}		V_{SS}	V_{SS}

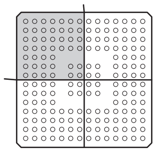
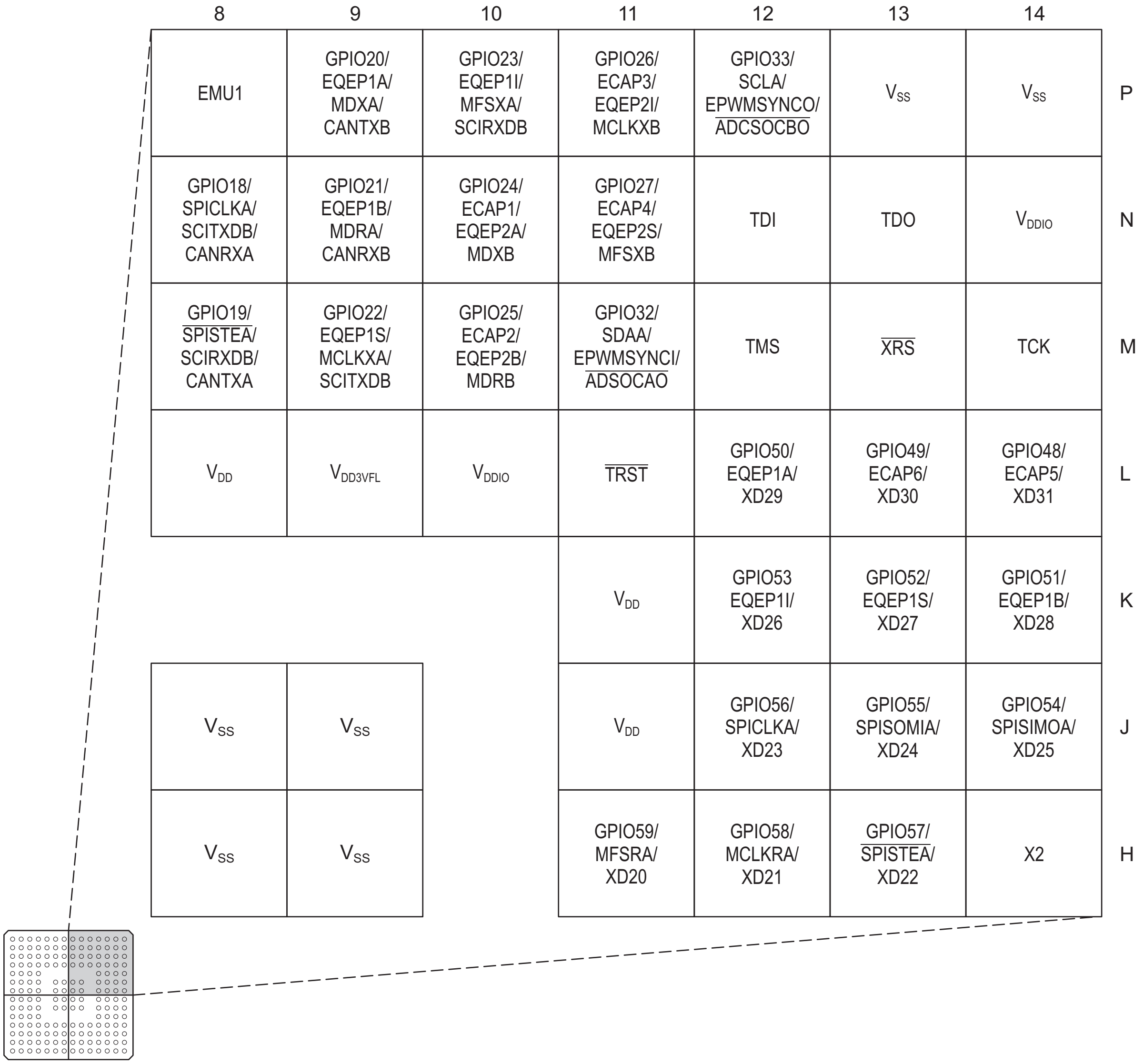
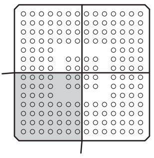


Figure 2-6. F28335, F28334, F28332 176-Ball ZJZ Plastic BGA (Upper Left Quadrant) (Bottom View)



8	9	10	11	12	13	14	
EMU1	GPIO20/ EQEP1A/ MDXA/ CANTXB	GPIO23/ EQEP1I/ MFSXA/ SCIRXDB	GPIO26/ ECAP3/ EQEP2I/ MCLKXB	GPIO33/ SCLA/ EPWMSYNCO/ $\overline{\text{ADCSOCBO}}$	V_{SS}	V_{SS}	P
GPIO18/ SPICLKA/ SCITXDB/ CANRXA	GPIO21/ EQEP1B/ MDRA/ CANRXB	GPIO24/ ECAP1/ EQEP2A/ MDXB	GPIO27/ ECAP4/ EQEP2S/ MFSXB	TDI	TDO	V_{DDIO}	N
GPIO19/ $\overline{\text{SPISTEA}}$/ SCIRXDB/ CANTXA	GPIO22/ EQEP1S/ MCLKXA/ SCITXDB	GPIO25/ ECAP2/ EQEP2B/ MDRB	GPIO32/ SDAA/ EPWMSYNCI/ $\overline{\text{ADSOCAO}}$	TMS	$\overline{\text{XRS}}$	TCK	M
V_{DD}	V_{DD3VFL}	V_{DDIO}	$\overline{\text{TRST}}$	GPIO50/ EQEP1A/ XD29	GPIO49/ ECAP6/ XD30	GPIO48/ ECAP5/ XD31	L
			V_{DD}	GPIO53 EQEP1I/ XD26	GPIO52/ EQEP1S/ XD27	GPIO51/ EQEP1B/ XD28	K
V_{SS}	V_{SS}		V_{DD}	GPIO56/ SPICLKA/ XD23	GPIO55/ SPISOMIA/ XD24	GPIO54/ SPISIMOA/ XD25	J
V_{SS}	V_{SS}		GPIO59/ MFSRA/ XD20	GPIO58/ MCLKRA/ XD21	GPIO57/ $\overline{\text{SPISTEA}}$/ XD22	X2	H

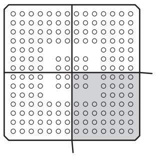
Figure 2-7. F28335, F28334, F28332 176-Ball ZJZ Plastic BGA (Upper Right Quadrant) (Bottom View)



G	GPIO9/ EPWM5B/ SCITXDB/ ECAP3	GPIO10/ EPWM6A/ CANRXB/ $\overline{\text{ADCSOCBO}}$	GPIO11/ EPWM6B/ SCIRXDB/ ECAP4	V_{DDIO}		V_{SS}	V_{SS}
F	GPIO6/ EPWM4A/ EPWMSYNCI/ EPWMSYNCO	GPIO7/ EPWM4B/ MCLKRA/ ECAP2	GPIO8/ EPWM5A/ CANTXB/ $\overline{\text{ADCSOCAO}}$	V_{DD}		V_{SS}	V_{SS}
E	GPIO3/ EPWM2B/ ECAP5/ MCLKRB	GPIO4/ EPWM3A	GPIO5/ EPWM3B/ MFSRA/ ECAP1	V_{DDIO}			
D	GPIO0/ EPWM1A	GPIO1/ EPWM1B/ ECAP6/ MFSRB	GPIO2/ EPWM2A	V_{DD}	V_{DD}	GPIO47/ XA7	V_{DDIO}
C	GPIO29/ SCITXDA/ XA19	GPIO30/ CANRXA/ XA18	GPIO39/ XA16	GPIO85/ XA13	GPIO82/ XA10	GPIO46/ XA6	GPIO43/ XA3
B	V_{DDIO}	GPIO31/ CANTXA/ XA17	GPIO87/ XA15	GPIO84/ XA12	GPIO81/ XA9	GPIO45/ XA5	GPIO42/ XA2
A	V_{SS}	V_{SS}	GPIO86/ XA14	GPIO83/ XA11	GPIO80/ XA8	GPIO44/ XA4	GPIO41/ XA1
	1	2	3	4	5	6	7

Figure 2-8. F28335, F28334, F28332 176-Ball ZJZ Plastic BGA (Lower Left Quadrant) (Bottom View)

ADVANCE INFORMATION



	8	9	10	11	12	13	14
G	V_{SS}	V_{SS}		V_{DDIO}	GPIO60/ MCLKRB/ XD19	XCLKIN	X1
F	V_{SS}	V_{SS}		V_{DD}	GPIO63/ SCITXDC/ XD16	GPIO62/ SCIRXDC/ XD17	GPIO61/ MFSRB/ XD18
E				V_{DD}	GPIO66/ XD13	GPIO65/ XD14	GPIO64/ XD15
D	V_{DD}	V_{DD}	GPIO28/ SCIRXDA/ $\overline{\text{XZCS6}}$	V_{DDIO}	GPIO69/ XD10	GPIO68/ XD11	GPIO67/ XD12
C	GPIO40/ XA0/$\overline{\text{XWE1}}$	GPIO36/ SCIRXDA/ $\overline{\text{XZCS0}}$	GPIO38/ $\overline{\text{XWE0}}$	GPIO78/ XD1	GPIO75/ XD4	GPIO71/ XD8	GPIO70/ XD9
B	GPIO37/ ECAP2/ $\overline{\text{XZCS7}}$	GPIO35/ SCITXDA/ XR/$\overline{\text{W}}$	GPIO79/ XD0	GPIO77/ XD2	GPIO74/ XD5	GPIO72 XD7	V_{SS}
A	$\overline{\text{XRD}}$	GPIO34/ ECAP1/ XREADY	XCLKOUT	GPIO76/ XD3	GPIO73/ XD6	V_{DDIO}	V_{SS}

Figure 2-9. F28335, F28334, F28332 176-Ball ZJZ Plastic BGA (Lower Right Quadrant) (Bottom View)

2.2 Signal Descriptions

Table 2-2 describes the signals on the F2833x devices. All digital inputs are TTL-compatible. All outputs are 3.3 V with CMOS levels. Inputs are not 5-V tolerant.

Table 2-2. Signal Descriptions

NAME	PIN NO.			DESCRIPTION (1)
	PGF PIN #	ZHH BAL L #	ZJZ BAL L #	
JTAG				
$\overline{\text{TRST}}$	78	M10	L11	JTAG test reset with internal pulldown. $\overline{\text{TRST}}$, when driven high, gives the scan system control of the operations of the device. If this signal is not connected or driven low, the device operates in its functional mode, and the test reset signals are ignored. NOTE: $\overline{\text{TRST}}$ is an active high test pin and must be maintained low at all times during normal device operation. An external pulldown resistor is recommended on this pin. The value of this resistor should be based on drive strength of the debugger pods applicable to the design. A 2.2-kΩ resistor generally offers adequate protection. Since this is application-specific, it is recommended that each target board be validated for proper operation of the debugger and the application. (I, ↓)
TCK	87	N12	M14	JTAG test clock with internal pullup (I, ↑)
TMS	79	P10	M12	JTAG test-mode select (TMS) with internal pullup. This serial control input is clocked into the TAP controller on the rising edge of TCK. (I, ↑)
TDI	76	M9	N12	JTAG test data input (TDI) with internal pullup. TDI is clocked into the selected register (instruction or data) on a rising edge of TCK. (I, ↑)
TDO	77	K9	N13	JTAG scan out, test data output (TDO). The contents of the selected register (instruction or data) are shifted out of TDO on the falling edge of TCK. (O/Z 8 mA drive)
EMU0	85	L11	N7	Emulator pin 0. When $\overline{\text{TRST}}$ is driven high, this pin is used as an interrupt to or from the emulator system and is defined as input/output through the JTAG scan. This pin is also used to put the device into boundary-scan mode. With the EMU0 pin at a logic-high state and the EMU1 pin at a logic-low state, a rising edge on the $\overline{\text{TRST}}$ pin would latch the device into boundary-scan mode. (I/O/Z, 8 mA drive ↑) **NOTE:** An external pullup resistor is recommended on this pin. The value of this resistor should be based on the drive strength of the debugger pods applicable to the design. A 2.2-kΩ to 4.7-kΩ resistor is generally adequate. Since this is application-specific, it is recommended that each target board be validated for proper operation of the debugger and the application.
EMU1	86	P12	P8	Emulator pin 1. When $\overline{\text{TRST}}$ is driven high, this pin is used as an interrupt to or from the emulator system and is defined as input/output through the JTAG scan. This pin is also used to put the device into boundary-scan mode. With the EMU0 pin at a logic-high state and the EMU1 pin at a logic-low state, a rising edge on the $\overline{\text{TRST}}$ pin would latch the device into boundary-scan mode. (I/O/Z, 8 mA drive ↑) **NOTE:** An external pullup resistor is recommended on this pin. The value of this resistor should be based on the drive strength of the debugger pods applicable to the design. A 2.2-kΩ to 4.7-kΩ resistor is generally adequate. Since this is application-specific, it is recommended that each target board be validated for proper operation of the debugger and the application.
FLASH				
V_{DD3VFL}	84	M11	L9	3.3-V Flash Core Power Pin. This pin should be connected to 3.3 V at all times.
TEST1	81	K10	M7	Test Pin. Reserved for TI. Must be left unconnected. (I/O)
TEST2	82	P11	L7	Test Pin. Reserved for TI. Must be left unconnected. (I/O)
CLOCK				
XCLKOUT	138	C11	A10	Output clock derived from SYSCLKOUT. XCLKOUT is either the same frequency, one-half the frequency, or one-fourth the frequency of SYSCLKOUT. This is controlled by bits 18:16 (XTIMCLK) and bit 2 (CLKMODE) in the XINTCNF2 register. At reset, XCLKOUT = SYSCLKOUT/4. The XCLKOUT signal can be turned off by setting XINTCNF2[CLKOFF] to 1. Unlike other GPIO pins, the XCLKOUT pin is not placed in high-impedance state during a reset. (O/Z, 8 mA drive).
XCLKIN	105	J14	G13	External Oscillator Input. This pin is to feed a clock from an external 3.3-V oscillator. In this case, the X1 pin must be tied to GND. If a crystal/resonator is used (or if an external 1.9-V oscillator is used to feed clock to X1 pin), this pin must be tied to GND. (I)

---

(1) I = Input, O = Output, Z = High impedance, OD = Open drain, ↑ = Pullup, ↓ = Pulldown

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION (1)
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
X1	104	J13	G14	Internal/External Oscillator Input. To use the internal oscillator, a quartz crystal or a ceramic resonator may be connected across X1 and X2. The X1 pin is referenced to the 1.9-V core digital power supply. A 1.9-V external oscillator may be connected to the X1 pin. In this case, the XCLKIN pin must be connected to ground. If a 3.3-V external oscillator is used with the XCLKIN pin, X1 must be tied to GND. (I)
X2	102	J11	H14	Internal Oscillator Output. A quartz crystal or a ceramic resonator may be connected across X1 and X2. If X2 is not used it must be left unconnected. (O)
RESET				
$\overline{XRS}$	80	L10	M13	Device Reset (in) and Watchdog Reset (out). Device reset. $\overline{XRS}$ causes the device to terminate execution. The PC will point to the address contained at the location 0x3FFFC0. When $\overline{XRS}$ is brought to a high level, execution begins at the location pointed to by the PC. This pin is driven low by the DSC when a watchdog reset occurs. During watchdog reset, the $\overline{XRS}$ pin is driven low for the watchdog reset duration of 512 OSCCLK cycles. (I/OD, ↑) The output buffer of this pin is an open-drain with an internal pullup. It is recommended that this pin be driven by an open-drain device.
ADC SIGNALS				
ADCINA7	35	K4	K1	ADC Group A, Channel 7 input (I)
ADCINA6	36	J5	K2	ADC Group A, Channel 6 input (I)
ADCINA5	37	L1	L1	ADC Group A, Channel 5 input (I)
ADCINA4	38	L2	L2	ADC Group A, Channel 4 input (I)
ADCINA3	39	L3	L3	ADC Group A, Channel 3 input (I)
ADCINA2	40	M1	M1	ADC Group A, Channel 2 input (I)
ADCINA1	41	N1	M2	ADC Group A, Channel 1 input (I)
ADCINA0	42	M3	M3	ADC Group A, Channel 0 input (I)
ADCINB7	53	K5	N6	ADC Group B, Channel 7 input (I)
ADCINB6	52	P4	M6	ADC Group B, Channel 6 input (I)
ADCINB5	51	N4	N5	ADC Group B, Channel 5 input (I)
ADCINB4	50	M4	M5	ADC Group B, Channel 4 input (I)
ADCINB3	49	L4	N4	ADC Group B, Channel 3 input (I)
ADCINB2	48	P3	M4	ADC Group B, Channel 2 input (I)
ADCINB1	47	N3	N3	ADC Group B, Channel 1 input (I)
ADCINB0	46	P2	P3	ADC Group B, Channel 0 input (I)
ADCLO	43	M2	N2	Low Reference (connect to analog ground) (I)
ADCRESEXT	57	M5	P6	ADC External Current Bias Resistor. Connect a 22-kΩ resistor to analog ground.
ADCREFIN	54	L5	P7	External reference input (I)
ADCREFP	56	P5	P5	Internal Reference Positive Output. Requires a low ESR (50 mΩ - 1.5 Ω) ceramic bypass capacitor of 2.2 µF to analog ground. (O)
ADCREFM	55	N5	P4	Internal Reference Medium Output. Requires a low ESR (50 mΩ - 1.5 Ω) ceramic bypass capacitor of 2.2 µF to analog ground. (O)
CPU AND I/O POWER PINS				
V_{DDA2}	34	K2	K4	ADC Analog Power Pin
V_{SSA2}	33	K3	P1	ADC Analog Ground Pin
V_{DDAIO}	45	N2	L5	ADC Analog I/O Power Pin
V_{SSAIO}	44	P1	N1	ADC Analog I/O Ground Pin
V_{DD1A18}	31	J4	K3	ADC Analog Power Pin
$V_{SS1AGND}$	32	K1	L4	ADC Analog Ground Pin
V_{DD2A18}	59	M6	L6	ADC Analog Power Pin
$V_{SS2AGND}$	58	K6	P2	ADC Analog Ground Pin

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION [(1)]
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
V_{DD}	4	B1	D4	CPU and Logic Digital Power Pins
V_{DD}	15	B5	D5	
V_{DD}	23	B11	D8	
V_{DD}	29	C8	D9	
V_{DD}	61	D13	E11	
V_{DD}	101	E9	F4	
V_{DD}	109	F3	F11	
V_{DD}	117	F13	H4	
V_{DD}	126	H1	J4	
V_{DD}	139	H12	J11	
V_{DD}	146	J2	K11	
V_{DD}	154	K14	L8	
V_{DD}	167	N6		
V_{DDIO}	9	A4	A13	Digital I/O Power Pin
V_{DDIO}	71	B10	B1	
V_{DDIO}	93	E7	D7	
V_{DDIO}	107	E12	D11	
V_{DDIO}	121	F5	E4	
V_{DDIO}	143	L8	G4	
V_{DDIO}	159	H11	G11	
V_{DDIO}	170	N14	L10	
V_{DDIO}			N14	
V_{SS}	3	A5	A1	Digital Ground Pins
V_{SS}	8	A10	A2	
V_{SS}	14	A11	A14	
V_{SS}	22	B4	B14	
V_{SS}	30	C3	F6	
V_{SS}	60	C7	F7	
V_{SS}	70	C9	F8	
V_{SS}	83	D1	F9	
V_{SS}	92	D6	G6	
V_{SS}	103	D14	G7	
V_{SS}	106	E8	G8	
V_{SS}	108	E14	G9	
V_{SS}	118	F4	H6	
V_{SS}	120	F12	H7	
V_{SS}	125	G1	H8	
V_{SS}	140	H10	H9	
V_{SS}	144	H13	J6	
V_{SS}	147	J3	J7	
V_{SS}	155	J10	J8	
V_{SS}	160	J12	J9	

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION (1)
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
V_{SS}	166	M12	P13	Digital Ground Pins
V_{SS}	171	N10	P14	
V_{SS}		N11		
V_{SS}		P6		
V_{SS}		P8		
GPIOA AND PERIPHERAL SIGNALS (2) (3)				
GPIO0 EPWM1A - -	5	C1	D1	General purpose input/output 0 (I/O/Z) (4) Enhanced PWM1 Output A and HRPWM channel (O) - -
GPIO1 EPWM1B ECAP6 MFSRB	6	D3	D2	General purpose input/output 1 (I/O/Z) (4) Enhanced PWM1 Output B (O) Enhanced Capture 6 input/output (I/O) McBSP-B receive frame synch (I/O)
GPIO2 EPWM2A - -	7	D2	D3	General purpose input/output 2 (I/O/Z) (4) Enhanced PWM2 Output A and HRPWM channel (O) - -
GPIO3 EPWM2B ECAP5 MCLKRB	10	E4	E1	General purpose input/output 3 (I/O/Z) (4) Enhanced PWM2 Output B (O) Enhanced Capture 5 input/output (I/O) McBSP-B receive clock (I/O)
GPIO4 EPWM3A - -	11	E2	E2	General purpose input/output 4 (I/O/Z) (4) Enhanced PWM3 output A and HRPWM channel (O) - -
GPIO5 EPWM3B MFSRA ECAP1	12	E3	E3	General purpose input/output 5 (I/O/Z) (4) Enhanced PWM3 output B (O) McBSP-A receive frame synch (I/O) Enhanced Capture input/output 1 (I/O)
GPIO6 EPWM4A EPWMSYNCI EPWMSYNCO	13	E1	F1	General purpose input/output 6 (I/O/Z) (4) Enhanced PWM4 output A and HRPWM channel (O) External ePWM sync pulse input (I) External ePWM sync pulse output (O)
GPIO7 EPWM4B MCLKRA ECAP2	16	F2	F2	General purpose input/output 7 (I/O/Z) (4) Enhanced PWM4 output B (O) McBSP-A receive clock (I/O) Enhanced capture input/output 2 (I/O)
GPIO8 EPWM5A CANTXB $\overline{\text{ADCSOCAO}}$	17	F1	F3	General Purpose Input/Output 8 (I/O/Z) (4) Enhanced PWM5 output A and HRPWM channel (O) Enhanced CAN-B transmit (O) ADC start-of-conversion A (O)
GPIO9 EPWM5B SCITXDB ECAP3	18	G5	G1	General purpose input/output 9 (I/O/Z) (4) Enhanced PWM5 output B (O) SCI-B transmit data(O) Enhanced capture input/output 3 (I/O)
GPIO10 EPWM6A CANRXB $\overline{\text{ADCSOCBO}}$	19	G4	G2	General purpose input/output 10 (I/O/Z) (4) Enhanced PWM6 output A and HRPWM channel (O) Enhanced CAN-B receive (I) ADC start-of-conversion B (O)

---

(2) Some peripheral functions may not be available in all devices. See Table 2-1 for details.
(3) All GPIO pins are I/O/Z, 4-mA drive typical (unless otherwise indicated), and have an internal pullup, which can be selectively enabled/disabled on a per-pin basis. This feature only applies to the GPIO pins. The GPIO function (shown in Italics) is the default at reset. The peripheral signals that are listed under them are alternate functions.
(4) The pullups on GPIO0-GPIO11 pins are not enabled at reset.

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION (1)
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
GPIO11 EPWM6B SCIRXDB ECAP4	20	G2	G3	General purpose input/output 11 (I/O/Z)(4) Enhanced PWM6 output B (O) SCI-B receive data (I) Enhanced CAP Input/Output 4 (I/O)
GPIO12 $\overline{\text{TZ1}}$ CANTXB MDXB	21	G3	H1	General purpose input/output 12 (I/O/Z)(5) Trip Zone input 1 (I) Enhanced CAN-B transmit (O) McBSP-B transmit serial data (O)
GPIO13 $\overline{\text{TZ2}}$ CANRXB MDRB	24	H3	H2	General purpose input/output 13 (I/O/Z)(5) Trip Zone input 2 (I) Enhanced CAN-B receive (I) McBSP-B receive serial data (I)
GPIO14 $\overline{\text{TZ3}}$/$\overline{\text{XHOLD}}$ SCITXDB MCLKXB	25	H2	H3	General purpose input/output 14 (I/O/Z)(5) Trip Zone input 3/External Hold Request. $\overline{\text{XHOLD}}$, when active (low), requests the external interface (XINTF) to release the external bus and place all buses and strobes into a high-impedance state. To prevent this from happening when $\overline{\text{TZ3}}$ signal goes active, disable this function by writing XINTCNF2[HOLD] = 1. If this is not done, the XINTF bus will go into high impedance anytime $\overline{\text{TZ3}}$ goes low. On the ePWM side, $\overline{\text{TZn}}$ signals are ignored by default, unless they are enabled by the code. The XINTF will release the bus when any current access is complete and there are no pending accesses on the XINTF. (I) SCI-B Transmit (I) McBSP-B transmit clock (I/O)
GPIO15 $\overline{\text{TZ4}}$/$\overline{\text{XHOLDA}}$ SCIRXDB MFSXB	26	H4	J1	General purpose input/output 15 (I/O/Z)(5) Trip Zone input 4/External Hold Acknowledge. The pin function for this option is based on the direction chosen in the GPADIR register. If the pin is configured as an input, then $\overline{\text{TZ4}}$ function is chosen. If the pin is configured as an output, then $\overline{\text{XHOLDA}}$ function is chosen. $\overline{\text{XHOLDA}}$ is driven active (low) when the XINTF has granted an $\overline{\text{XHOLD}}$ request. All XINTF buses and strobe signals will be in a high-impedance state. $\overline{\text{XHOLDA}}$ is released when the $\overline{\text{XHOLD}}$ signal is released. External devices should only drive the external bus when $\overline{\text{XHOLDA}}$ is active (low). (I/0) SCI-B receive (I) McBSP-B transmit frame synch (I/O)
GPIO16 SPISIMOA CANTXB $\overline{\text{TZ5}}$	27	H5	J2	General purpose input/output 16 (I/O/Z)(5) SPI slave in, master out (I/O) Enhanced CAN-B transmit (O) Trip Zone input 5 (I)
GPIO17 SPISOMIA CANRXB $\overline{\text{TZ6}}$	28	J1	J3	General purpose input/output 17 (I/O/Z)(5) SPI-A slave out, master in (I/O) Enhanced CAN-B receive (I) Trip zone input 6 (I)
GPIO18 SPICLKA SCITXDB CANRXA	62	L6	N8	General purpose input/output 18 (I/O/Z)(5) SPI-A clock input/output (I/O) SCI-B transmit (O) Enhanced CAN-A receive (I)
GPIO19 $\overline{\text{SPISTEA}}$ SCIRXDB CANTXA	63	K7	M8	General purpose input/output 19 (I/O/Z)(5) SPI-A slave transmit enable input/output (I/O) SCI-B receive (I) Enhanced CAN-A transmit (O)
GPIO20 EQEP1A MDXA CANTXB	64	L7	P9	General purpose input/output 20 (I/O/Z)(5) Enhanced QEP1 input A (I) McBSP-A transmit serial data (O) Enhanced CAN-B transmit (O)
GPIO21 EQEP1B MDRA CANRXB	65	P7	N9	General purpose input/output 21 (I/O/Z)(5) Enhanced QEP1 input B (I) McBSP-A receive serial data (I) Enhanced CAN-B receive (I)

---

(5) The pullups on GPIO12-GPIO34 are enabled upon reset.

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION [1]
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
GPIO22 EQEP1S MCLKXA SCITXDB	66	N7	M9	General purpose input/output 22 (I/O/Z)[5] Enhanced QEP1 strobe (I/O) McBSP-A transmit clock (I/O) SCI-B transmit (O)
GPIO23 EQEP1I MFSXA SCIRXDB	67	M7	P10	General purpose input/output 23 (I/O/Z)[5] Enhanced QEP1 index (I/O) McBSP-A transmit frame synch (I/O) SCI-B receive (I)
GPIO24 ECAP1 EQEP2A MDXB	68	M8	N10	General purpose input/output 24 (I/O/Z)[5] Enhanced capture 1 (I/O) Enhanced QEP2 input A (I) McBSP-B transmit serial data (O)
GPIO25 ECAP2 EQEP2B MDRB	69	N8	M10	General purpose input/output 25 (I/O/Z)[5] Enhanced capture 2 (I/O) Enhanced QEP2 input B (I) McBSP-B receive serial data (I)
GPIO26 ECAP3 EQEP2I MCLKXB	72	K8	P11	General purpose input/output 26 (I/O/Z)[5] Enhanced capture 3 (I/O) Enhanced QEP2 index (I/O) McBSP-B transmit clock (I/O)
GPIO27 ECAP4 EQEP2S MFSXB	73	L9	N11	General purpose input/output 27 (I/O/Z)[5] Enhanced capture 4 (I/O) Enhanced QEP2 strobe (I/O) McBSP-B transmit frame synch (I/O)
GPIO28 SCIRXDA $\overline{\text{XZCS6}}$	141	E10	D10	General purpose input/output 28 (I/O/Z)[5] SCI receive data (I) External Interface zone 6 chip select (O)
GPIO29 SCITXDA XA19	2	C2	C1	General purpose input/output 29. (I/O/Z)[5] SCI transmit data (O) External Interface Address Line 19 (O)
GPIO30 CANRXA XA18	1	B2	C2	General purpose input/output 30 (I/O/Z)[5] Enhanced CAN-A receive (I) External Interface Address Line 18 (O)
GPIO31 CANTXA XA17	176	A2	B2	General purpose input/output 31 (I/O/Z)[5] Enhanced CAN-A transmit (O) External Interface Address Line 17 (O)
GPIO32 SDAA EPWMSYNCI $\overline{\text{ADCSOCAO}}$	74	N9	M11	General purpose input/output 32 (I/O/Z)[5] I2C data open-drain bidirectional port (I/OD) Enhanced PWM external sync pulse input (I) ADC start-of-conversion A (O)
GPIO33 SCLA EPWMSYNCO $\overline{\text{ADCSOCBO}}$	75	P9	P12	General-Purpose Input/Output 33 (I/O/Z)[5] I2C clock open-drain bidirectional port (I/OD) Enhanced PWM external synch pulse output (O) ADC start-of-conversion B (O)
GPIO34 ECAP1 XREADY	142	D10	A9	General-Purpose Input/Output 34 (I/O/Z)[5] Enhanced Capture input/output 1 (I/O) External Interface Ready signal
GPIO35 SCITXDA XR/$\overline{\text{W}}$	148	A9	B9	General-Purpose Input/Output 35 (I/O/Z) SCI-A transmit data (O) External Interface read, not write strobe
GPIO36 SCIRXDA $\overline{\text{XZCS0}}$	145	C10	C9	General-Purpose Input/Output 36 (I/O/Z) SCI receive data (I) External Interface zone 0 chip select (O)
GPIO37 ECAP2 $\overline{\text{XZCS7}}$	150	D9	B8	General-Purpose Input/Output 37 (I/O/Z) Enhanced Capture input/output 2 (I/O) External Interface zone 7 chip select (O)

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION [1]
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
GPIO38 - $\overline{\text{XWE0}}$	137	D11	C10	General-Purpose Input/Output 38 (I/O/Z) - External Interface Write Enable 0 (O)
GPIO39 - XA16	175	B3	C3	General-Purpose Input/Output 39 (I/O/Z) - External Interface Address Line 16 (O)
GPIO40 - XA0/$\overline{\text{XWE1}}$	151	D8	C8	General-Purpose Input/Output 40 (I/O/Z) - External Interface Address Line 0/External Interface Write Enable 1 (O)
GPIO41 - XA1	152	A8	A7	General-Purpose Input/Output 41 (I/O/Z) - External Interface Address Line 1 (O)
GPIO42 - XA2	153	B8	B7	General-Purpose Input/Output 42 (I/O/Z) - External Interface Address Line 2 (O)
GPIO43 - XA3	156	B7	C7	General-Purpose Input/Output 43 (I/O/Z) - External Interface Address Line 3 (O)
GPIO44 - XA4	157	A7	A6	General-Purpose Input/Output 44 (I/O/Z) - External Interface Address Line 4 (O)
GPIO45 - XA5	158	D7	B6	General-Purpose Input/Output 45 (I/O/Z) - External Interface Address Line 5 (O)
GPIO46 - XA6	161	B6	C6	General-Purpose Input/Output 46 (I/O/Z) - External Interface Address Line 6 (O)
GPIO47 - XA7	162	A6	D6	General-Purpose Input/Output 47 (I/O/Z) - External Interface Address Line 7 (O)
GPIO48 ECAP5 XD31	88	P13	L14	General-Purpose Input/Output 48 (I/O/Z) Enhanced Capture input/output 5 (I/O) External Interface Data Line 31 (O)
GPIO49 ECAP6 XD30	89	N13	L13	General-Purpose Input/Output 49 (I/O/Z) Enhanced Capture input/output 6 (I/O) External Interface Data Line 30 (O)
GPIO50 EQEP1A XD29	90	P14	L12	General-Purpose Input/Output 50 (I/O/Z) Enhanced QEP 1input A (I) External Interface Data Line 29 (O)
GPIO51 EQEP1B XD28	91	M13	K14	General-Purpose Input/Output 51 (I/O/Z) Enhanced QEP 1input B (I) External Interface Data Line 28 (O)
GPIO52 EQEP1S XD27	94	M14	K13	General-Purpose Input/Output 52 (I/O/Z) Enhanced QEP 1Strobe (I/O) External Interface Data Line 27 (O)
GPIO53 EQEP1I XD26	95	L12	K12	General-Purpose Input/Output 53 (I/O/Z) Enhanced CAP1 Index (I/O) External Interface Data Line 26 (O)
GPIO54 SPISIMOA XD25	96	L13	J14	General-Purpose Input/Output 54 (I/O/Z) SPI-A slave in, master out (I/O) External Interface Data Line 25 (O)
GPIO55 SPISOMIA XD24	97	L14	J13	General-Purpose Input/Output 55 (I/O/Z) SPI-A slave out, master in (I/O) External Interface Data Line 24 (O)
GPIO56 SPICLKA XD23	98	K11	J12	General-Purpose Input/Output 56 (I/O/Z) SPI-A clock (I/O) External Interface Data Line 23 (O)

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION (1)
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
GPIO57 $\overline{\text{SPISTEA}}$ XD22	99	K13	H13	General-Purpose Input/Output 57 (I/O/Z) SPI-A slave transmit enable (I/O) External Interface Data Line 22 (O)
GPIO58 MCLKRA XD21	100	K12	H12	General-Purpose Input/Output 58 (I/O/Z) McBSP-A receive clock (I/O) External Interface Data Line 21 (O)
GPIO59 MFSRA XD20	110	H14	H11	General-Purpose Input/Output 59 (I/O/Z) McBSP-A receive frame synch (I/O) External Interface Data Line 20 (O)
GPIO60 MCLKRB XD19	111	G14	G12	General-Purpose Input/Output 60 (I/O/Z) McBSP-B receive clock (I/O) External Interface Data Line 19 (O)
GPIO61 MFSRB XD18	112	G12	F14	General-Purpose Input/Output 61 (I/O/Z) McBSP-B receive frame synch (I/O) External Interface Data Line 18 (O)
GPIO62 SCIRXDC XD17	113	G13	F13	General-Purpose Input/Output 62 (I/O/Z) SCI-C receive data (I) External Interface Data Line 17 (O)
GPIO63 SCITXDC XD16	114	G11	F12	General-Purpose Input/Output 63 (I/O/Z) SCI-C transmit data (O) External Interface Data Line 16 (O)
GPIO64 - XD15	115	G10	E14	General-Purpose Input/Output 64 (I/O/Z) - External Interface Data Line 15 (O)
GPIO65 - XD14	116	F14	E13	General-Purpose Input/Output 65 (I/O/Z) - External Interface Data Line 14 (O)
GPIO66 - XD13	119	F11	E12	General-Purpose Input/Output 66 (I/O/Z) - External Interface Data Line 13 (O)
GPIO67 - XD12	122	E13	D14	General-Purpose Input/Output 67 (I/O/Z) - External Interface Data Line 12 (O)
GPIO68 - XD11	123	E11	D13	General-Purpose Input/Output 68 (I/O/Z) - External Interface Data Line 11 (O)
GPIO69 - XD10	124	F10	D12	General-Purpose Input/Output 69 (I/O/Z) - External Interface Data Line 10 (O)
GPIO70 - XD9	127	D12	C14	General-Purpose Input/Output 70 (I/O/Z) - External Interface Data Line 9 (O)
GPIO71 - XD8	128	C14	C13	General-Purpose Input/Output 71 (I/O/Z) - External Interface Data Line 8 (O)
GPIO72 - XD7	129	B14	B13	General-Purpose Input/Output 72 (I/O/Z) - External Interface Data Line 7 (O)
GPIO73 - XD6	130	C12	A12	General-Purpose Input/Output 73 (I/O/Z) - External Interface Data Line 6 (O)
GPIO74 - XD5	131	C13	B12	General-Purpose Input/Output 74 (I/O/Z) - External Interface Data Line 5 (O)
GPIO75 - XD4	132	A14	C12	General-Purpose Input/Output 75 (I/O/Z) - External Interface Data Line 4 (O)

Table 2-2. Signal Descriptions (continued)

NAME	PIN NO.			DESCRIPTION [1]
	PGF PIN #	ZHH BALL #	ZJZ BALL #	
GPIO76 - XD3	133	B13	A11	General-Purpose Input/Output 76 (I/O/Z) - External Interface Data Line 3 (O)
GPIO77 - XD2	134	A13	B11	General-Purpose Input/Output 77 (I/O/Z) - External Interface Data Line 2 (O)
GPIO78 - XD1	135	B12	C11	General-Purpose Input/Output 78 (I/O/Z) - External Interface Data Line 1 (O)
GPIO79 - XD0	136	A12	B10	General-Purpose Input/Output 79 (I/O/Z) - External Interface Data Line 0 (O)
GPIO80 - XA8	163	C6	A5	General-Purpose Input/Output 80 (I/O/Z) - External Interface Address Line 8 (O)
GPIO81 - XA9	164	E6	B5	General-Purpose Input/Output 81 (I/O/Z) - External Interface Address Line 9 (O)
GPIO82 - XA10	165	C5	C5	General-Purpose Input/Output 82 (I/O/Z) - External Interface Address Line 10 (O)
GPIO83 - XA11	168	D5	A4	General-Purpose Input/Output 83 (I/O/Z) - External Interface Address Line 11 (O)
GPIO84 - XA12	169	E5	B4	General-Purpose Input/Output 84 (I/O/Z) External Interface Address Line 12 (O)
GPIO85 - XA13	172	C4	C4	General-Purpose Input/Output 85 (I/O/Z) - External Interface Address Line 13 (O)
GPIO86 - XA14	173	D4	A3	General-Purpose Input/Output 86 (I/O/Z) - External Interface Address Line 14 (O)
GPIO87 - XA15	174	A3	B3	General-Purpose Input/Output 87 (I/O/Z) - External Interface Address Line 15 (O)
$\overline{\text{XRD}}$	149	B9	A8	External Interface Read Enable

ADVANCE INFORMATION

3 Functional Overview

ADVANCE INFORMATION

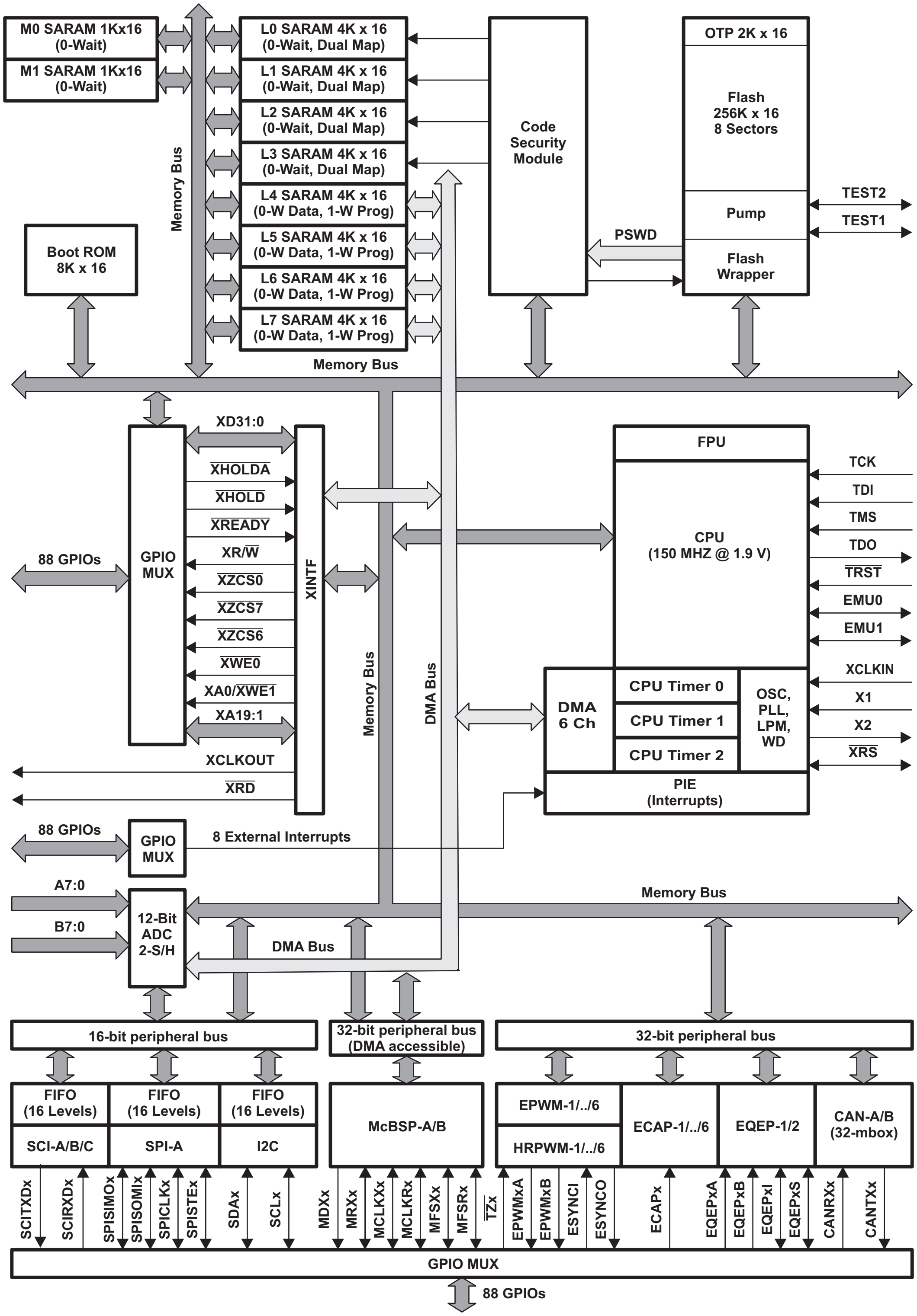


Figure 3-1. Functional Block Diagram

3.1 Memory Maps

In Figure 3-2 through Figure 3-4, the following apply:

- Memory blocks are not to scale.
- Peripheral Frame 0, Peripheral Frame 1, Peripheral Frame 2, and Peripheral Frame 3 memory maps are restricted to data memory only. A user program cannot access these memory maps in program space.
- *Protected* means the order of "Write followed by Read" operations is preserved rather than the pipeline order.
- Certain memory ranges are EALLOW protected against spurious writes after configuration.
- The TI OTP ROM (0x38 0000 – 0x38 03FF) is readable and contains the ADC calibration routine. It is not programmable by the user.
- If the eCAN module is not used in an application, the RAM available (LAM, MOTS, MOTO, and mailbox RAM) can be used as general-purpose RAM. The CAN module clock should be enabled for this.

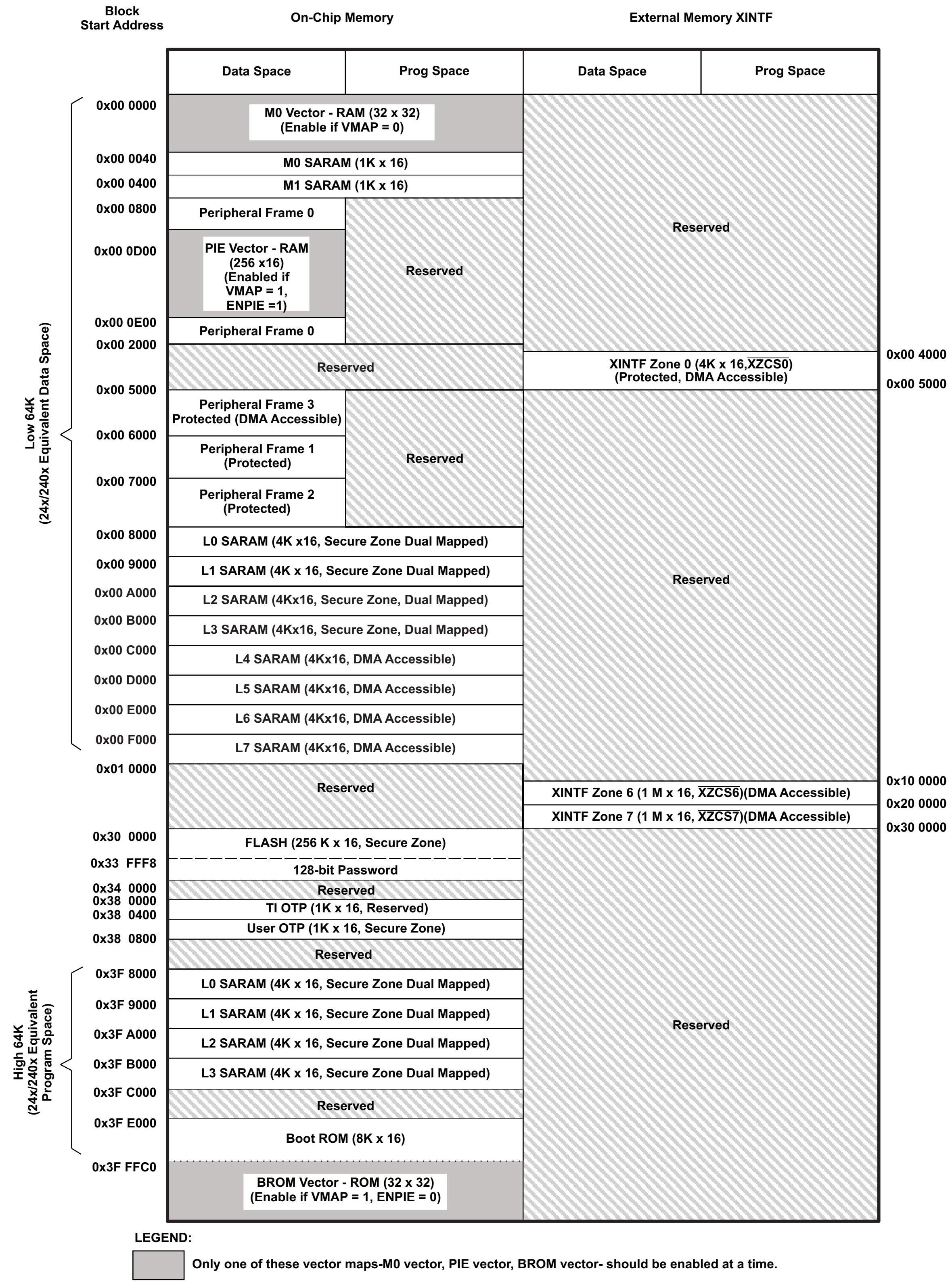
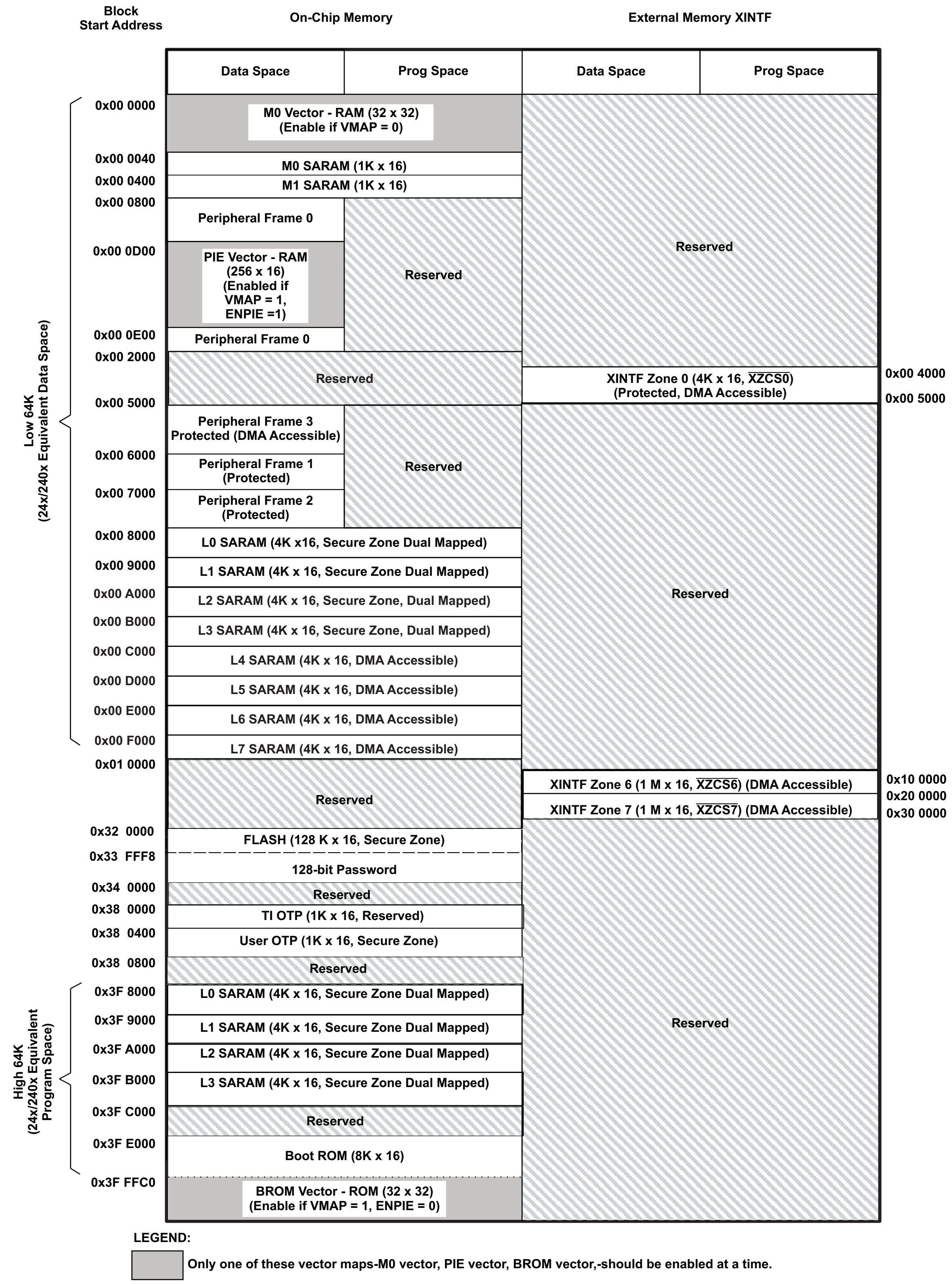
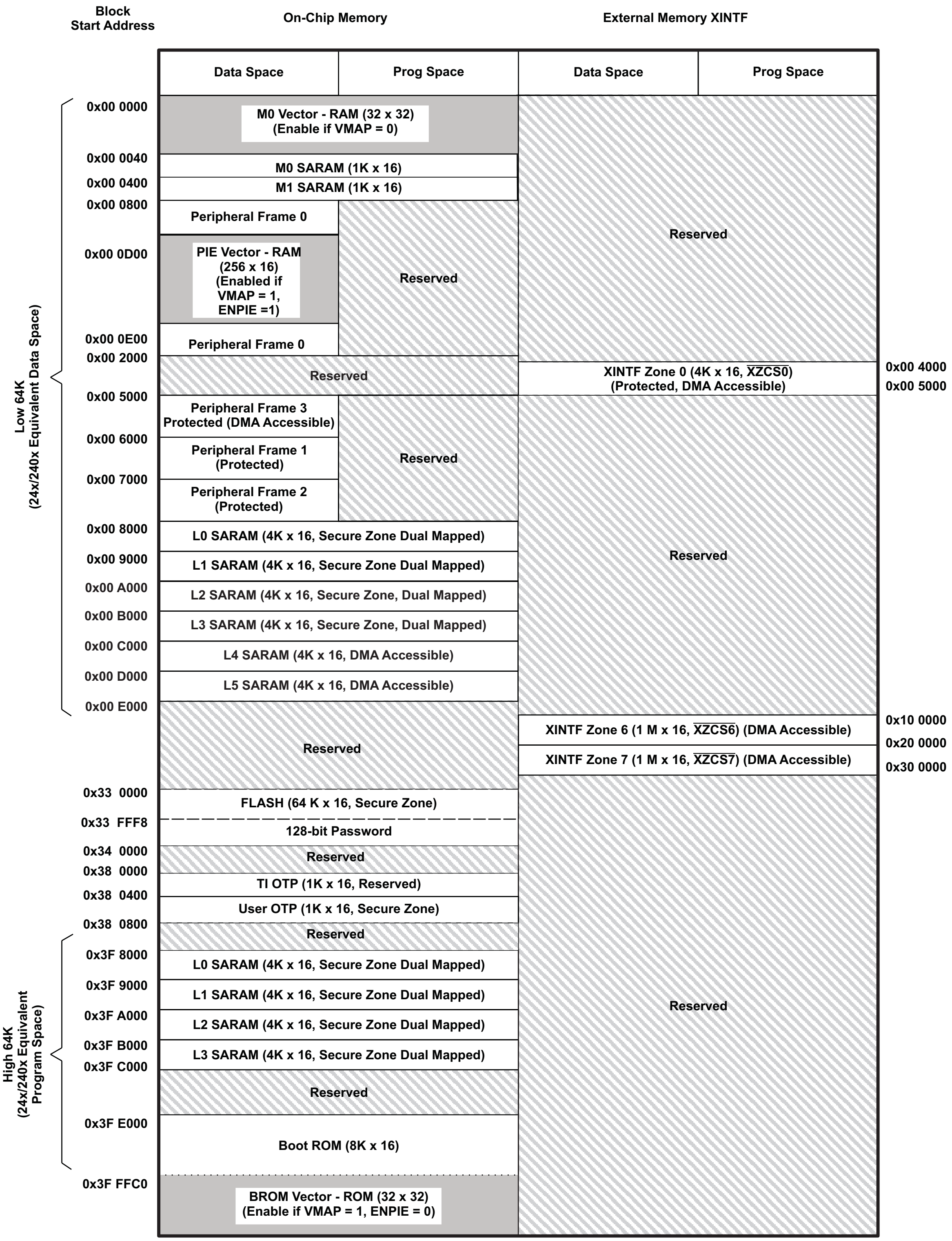


Figure 3-2. F28335 Memory Map



ADVANCE INFORMATION

Figure 3-3. F28334 Memory Map



LEGEND:

▨ Only one of these vector maps-M0 vector, PIE vector, BROM vector,-should be enabled at a time.

Figure 3-4. F28332 Memory Map

Table 3-1. Addresses of Flash Sectors in F28335

ADDRESS RANGE	PROGRAM AND DATA SPACE
0x30 0000 - 0x30 7FFF	Sector H (32K x 16)
0x30 8000 - 0x30 FFFF	Sector G (32K x 16)
0x31 0000 - 0x31 7FFF	Sector F (32K x 16)
0x31 8000 - 0x31 FFFF	Sector E (32K x 16)
0x32 0000 - 0x32 7FFF	Sector D (32K x 16)
0x32 8000 - 0x32 FFFF	Sector C (32K x 16)
0x33 0000 - 0x33 7FFF	Sector B (32K x 16)
0x33 8000 - 0x33 FF7F	Sector A (32K x 16)
0x33 FF80 - 0x33 FFF5	Program to 0x0000 when using the Code Security Module
0x33 FFF6 - 0x33 FFF7	Boot-to-Flash Entry Point (program branch instruction here)
0x33 FFF8 - 0x33 FFFF	Security Password (128-Bit) (Do Not Program to all zeros)

Table 3-2. Addresses of Flash Sectors in F28334

ADDRESS RANGE	PROGRAM AND DATA SPACE
0x32 0000 - 0x32 3FFF	Sector H (16K x 16)
0x32 4000 - 0x32 7FFF	Sector G (16K x 16)
0x32 8000 - 0x32 BFFF	Sector F (16K x 16)
0x32 C000 - 0x32 FFFF	Sector E (16K x 16)
0x33 0000 - 0x33 3FFF	Sector D (16K x 16)
0x33 4000 - 0x33 7FFFF	Sector C (16K x 16)
0x33 8000 - 0x33 BFFF	Sector B (16K x 16)
0x33 C000 - 0x33 FF7F	Sector A (16K x 16)
0x33 FF80 - 0x33 FFF5	Program to 0x0000 when using the Code Security Module
0x33 FFF6 - 0x33 FFF7	Boot-to-Flash Entry Point (program branch instruction here)
0x33 FFF8 - 0x33 FFFF	Security Password (128-Bit) (Do Not Program to all zeros)

Table 3-3. Addresses of Flash Sectors in F28332

ADDRESS RANGE	PROGRAM AND DATA SPACE
0x33 0000 - 0x33 3FFF	Sector D (16K x 16)
0x33 4000 - 0x33 7FFFF	Sector C (16K x 16)
0x33 8000 - 0x33 BFFF	Sector B (16K x 16)
0x33 C000 - 0x33 FF7F	Sector A (16K x 16)
0x33 FF80 - 0x33 FFF5	Program to 0x0000 when using the Code Security Module
0x33 FFF6 - 0x33 FFF7	Boot-to-Flash Entry Point (program branch instruction here)
0x33 FFF8 - 0x33 FFFF	Security Password (128-Bit) (Do Not Program to all zeros)

ADVANCE INFORMATION

NOTE

- When the code-security passwords are programmed, all addresses between 0x33FF80 and 0x33FFF5 cannot be used as program code or data. These locations must be programmed to 0x0000.
- If the code security feature is not used, addresses 0x33FF80 through 0x33FFEF may be used for code or data. Addresses 0x33FFF0 – 0x33FFF5 are reserved for data and should not contain program code. .

Table 3-4 shows how to handle these memory locations.

Table 3-4. Handling Security Code Locations

ADDRESS	FLASH	
	Code security enabled	**Code security disabled**
0x33FF80 - 0x33FFEF	Fill with 0x0000	Application code and data
0x33FFF0 - 0x33FFF5		Reserved for data only

Peripheral Frame 1, Peripheral Frame 2, and Peripheral Frame 3 are grouped together to enable these blocks to be write/read peripheral block protected. The protected mode ensures that all accesses to these blocks happen as written. Because of the C28x pipeline, a write immediately followed by a read, to different memory locations, will appear in reverse order on the memory bus of the CPU. This can cause problems in certain peripheral applications where the user expected the write to occur first (as written). The C28x CPU supports a block protection mode where a region of memory can be protected so as to make sure that operations occur as written (the penalty is extra cycles are added to align the operations). This mode is programmable and by default, it will protect the selected zones.

The wait-states for the various spaces in the memory map area are listed in Table 3-5.

Table 3-5. Wait-states

Area	Wait-States (CPU)	Wait-States (DMA)[1]	Comments
M0 and M1 SARAMs	0-wait		Fixed
Peripheral Frame 0	0-wait (writes) 1-wait (reads)	0-wait (reads)	
Peripheral Frame 3	0-wait (writes) 2-wait (reads)	0-wait (writes) 1-wait (reads)	Assumes no conflicts between CPU and DMA.
Peripheral Frame 1	0-wait (writes) 2-wait (reads)		Cycles can be extended by peripheral generated ready. Consecutive writes to the CAN will experience a 1-cycle pipeline hit.
Peripheral Frame 2	0-wait (writes) 2-wait (reads)		Fixed. Cycles cannot be extended by the peripheral.
L0 SARAM	0-wait data and program		Assumes no CPU conflicts
L1 SARAM			
L2 SARAM			
L3 SARAM			
L4 SARAM	0-wait data (read) 0-wait data (write) 1-wait program (read) 1-wait program (write)	0-wait data (write) 0-wait data (read)	Assumes no conflicts between CPU and DMA.
L5 SARAM			
L6 SARAM			
L7 SARAM			
XINTF	Programmable		Programmed via the XTIMING registers or extendable via external XREADY signal.
	1-wait minimum		1-wait is minimum wait states allowed on external waveforms for both reads and writes on XINTF.
	0-wait minimum writes with write buffer enabled	0-wait data (write) 0-wait data (read)	0-wait minimum for writes assumes write buffer enabled and not full. Assumes no conflicts between CPU and DMA. When DMA and CPU attempt simultaneous conflict, 1-cycle delay is added for arbitration.
OTP	Programmable		Programmed via the Flash registers.
	1-wait minimum		1-wait is minimum number of wait states allowed. 1-wait-state operation is possible at a reduced CPU frequency.
FLASH	Programmable		Programmed via the Flash registers.
	1-wait Paged min		0-wait minimum for paged access is not allowed
	1-wait Random min Random ≥ Paged		1-wait-state operation is possible at a reduced CPU frequency.
FLASH Password	16-wait fixed		Wait states of password locations are fixed.
Boot-ROM	1-wait		0-wait speed is not possible.

(1) The DMA has a base of 4 cycles/word.

3.2 Brief Descriptions

3.2.1 C28x CPU

The F2833x (C28x+FPU) family is a member of the TMS320C2000™ digital signal controller (DSC) platform. The C28x+FPU based controllers have the same 32-bit fixed-point architecture as TI's existing C28x DSCs, but also include a single-precision (32-bit) IEEE 754 floating-point-unit (FPU). It is a very efficient C/C++ engine, hence enabling users to develop not only their system control software in a high-level language, but also enables math algorithms to be developed using C/C++. The device is as efficient in DSP math tasks as it is in system control tasks that typically are handled by microcontroller devices. This efficiency removes the need for a second processor in many systems. The 32 x 32-bit MAC capabilities of the F2833x and its 64-bit processing capabilities, enable it to efficiently handle higher numerical resolution problems. Add to this the fast interrupt response with automatic context save of critical registers, resulting in a device that is capable of servicing many asynchronous events with minimal latency. The device has an 8-level-deep protected pipeline with pipelined memory accesses. This pipelining enables it to execute at high speeds without resorting to expensive high-speed memories. Special branch-look-ahead hardware minimizes the latency for conditional discontinuities. Special store conditional operations further improve performance.

3.2.2 Memory Bus (Harvard Bus Architecture)

As with many DSC type devices, multiple busses are used to move data between the memories and peripherals and the CPU. The C28x memory bus architecture contains a program read bus, data read bus and data write bus. The program read bus consists of 22 address lines and 32 data lines. The data read and write busses consist of 32 address lines and 32 data lines each. The 32-bit-wide data busses enable single cycle 32-bit operations. The multiple bus architecture, commonly termed Harvard Bus, enables the C28x to fetch an instruction, read a data value and write a data value in a single cycle. All peripherals and memories attached to the memory bus will prioritize memory accesses. Generally, the priority of memory bus accesses can be summarized as follows:

Highest:	Data Writes	(Simultaneous data and program writes cannot occur on the memory bus.)
	Program Writes	(Simultaneous data and program writes cannot occur on the memory bus.)
	Data Reads	
	Program Reads	(Simultaneous program reads and fetches cannot occur on the memory bus.)
Lowest:	Fetches	(Simultaneous program reads and fetches cannot occur on the memory bus.)

3.2.3 Peripheral Bus

To enable migration of peripherals between various Texas Instruments (TI) DSC family of devices, the F2833x devices adopt a peripheral bus standard for peripheral interconnect. The peripheral bus bridge multiplexes the various busses that make up the processor Memory Bus into a single bus consisting of 16 address lines and 16 or 32 data lines and associated control signals. Three versions of the peripheral bus are supported on the F2833x. One version supports only 16-bit accesses (called peripheral frame 2). Another version supports both 16- and 32-bit accesses (called peripheral frame 1). The third version supports DMA access and both 16- and 32-bit accesses (called peripheral frame 3).

3.2.4 Real-Time JTAG and Analysis

The F2833x implements the standard IEEE 1149.1 JTAG interface. Additionally, the F2833x supports real-time mode of operation whereby the contents of memory, peripheral and register locations can be modified while the processor is running and executing code and servicing interrupts. The user can also single step through non-time critical code while enabling time-critical interrupts to be serviced without interference. The F2833x implements the real-time mode in hardware within the CPU. This is a unique feature to the F2833x, no software monitor is required. Additionally, special analysis hardware is provided which allows the user to set hardware breakpoint or data/address watch-points and generate various user-selectable break events when a match occurs.

3.2.5 External Interface (XINTF)

This asynchronous interface consists of 20 address lines, 32 data lines, and three chip-select lines. The chip-select lines are mapped to three external zones, Zones 0, 6, and 7. Each of the three zones can be programmed with a different number of wait states, strobe signal setup and hold timing and each zone can be programmed for extending wait states externally or not. The programmable wait-state, chip-select and programmable strobe timing enables glueless interface to external memories and peripherals.

3.2.6 Flash

The F28335 contains 256K × 16 of embedded flash memory, segregated into eight 32K × 16 sectors. The F28334 contains 128K × 16 of embedded flash memory, segregated into eight 16K × 16 sectors. The F28332 device contains 64K ×16 of embedded flash, segregated into four 16K × 16 sectors. All the devices also contain a single 1K × 16 of OTP memory at address range 0x380400 – 0x3807FF. The user can individually erase, program, and validate a flash sector while leaving other sectors untouched. However, it is not possible to use one sector of the flash or the OTP to execute flash algorithms that erase/program other sectors. Special memory pipelining is provided to enable the flash module to achieve higher performance. The flash/OTP is mapped to both program and data space; therefore, it can be used to execute code or store data information. Note that addresses 0x33FFF0 – 0x33FFF5 are reserved for data variables and should not contain program code.

NOTE

The F28335/F28334/F28332 Flash and OTP wait-states can be configured by the application. This allows applications running at slower frequencies to configure the flash to use fewer wait-states.

Flash effective performance can be improved by enabling the flash pipeline mode in the Flash options register. With this mode enabled, effective performance of linear code execution will be much faster than the raw performance indicated by the wait-state configuration alone. The exact performance gain when using the Flash pipeline mode is application-dependent.

For more information on the Flash options, Flash wait-state, and OTP wait-state registers, see the *TMS320F2833x Digital Signal Controller (DSC) System Control and Interrupts Reference Guide* (literature number SPRUFB0).

3.2.7 M0, M1 SARAMs

All F2833x devices contain these two blocks of single access memory, each 1K × 16 in size. The stack pointer points to the beginning of block M1 on reset. The M0 and M1 blocks, like all other memory blocks on C28x devices, are mapped to both program and data space. Hence, the user can use M0 and M1 to execute code or for data variables. The partitioning is performed within the linker. The C28x device presents a unified memory map to the programmer. This makes for easier programming in high-level languages.

3.2.8 L0, L1, L2, L3, L4, L5, L6, L7 SARAMs

The F28335 and F28334 each contain an additional 32K × 16 of single-access RAM, divided into 8 blocks (L0-L7 with 4K each). The F28332 contains an additional 24K × 16 of single-access RAM, divided into 6 blocks (L0-L5 with 4K each). Each block can be independently accessed to minimize CPU pipeline stalls. Each block is mapped to both program and data space. L4, L5, L6, and L7 are DMA accessible

3.2.9 Boot ROM

The Boot ROM is factory-programmed with boot-loading software. Boot-mode signals are provided to tell the bootloader software what boot mode to use on power up. The user can select to boot normally or to download new software from an external connection or to select boot software that is programmed in the internal Flash/ROM. The Boot ROM also contains standard tables, such as SIN/COS waveforms, for use in math related algorithms.

Table 3-6. Boot Mode Selection

MODE	GPIO87/XA15	GPIO86/XA14	GPIO85/XA13	GPIO84/XA12	MODE [1]
F	1	1	1	1	Jump to Flash
E	1	1	1	0	SCI-A boot
D	1	1	0	1	SPI-A boot
C	1	1	0	0	I2C-A boot
B	1	0	1	1	eCAN-A boot
A	1	0	1	0	McBSP-A boot
9	1	0	0	1	Jump to XINTF x16
8	1	0	0	0	Jump to XINTF x32
7	0	1	1	1	Jumpto OTP
6	0	1	1	0	Parallel GPIO I/O boot
5	0	1	0	1	Parallel XINTF boot
4	0	1	0	0	Jump to SARAM
3	0	0	1	1	Branch to check boot mode
2	0	0	1	0	Branch to Flash, skip ADC calibration
1	0	0	0	1	Branch to SARAM, skip ADC calibration
0	0	0	0	0	Branch to SCI, skip ADC calibration

(1) All four GPIO pins have an internal pullup.

NOTE

Modes 0, 1, and 2 in Table 3-6 are for TI debug only. Skipping the ADC calibration function in an application will cause the ADC to operate outside of the stated specifications

3.2.10 Security

The F2833x devices support high levels of security to protect the user firmware from being reverse engineered. The security features a 128-bit password (hardcoded for 16 wait-states), which the user programs into the flash. One code security module (CSM) is used to protect the flash/OTP and the L0/L1/L2/L3 SARAM blocks. The security feature prevents unauthorized users from examining the memory contents via the JTAG port, executing code from external memory or trying to boot-load some undesirable software that would export the secure memory contents. To enable access to the secure blocks, the user must write the correct 128-bit KEY value, which matches the value stored in the password locations within the Flash.

In addition to the CSM, the emulation code security logic (ECSL) has been implemented to prevent unauthorized users from stepping through secure code. Any code or data access to flash, user OTP, L0, L1, L2 or L3 memory while the emulator is connected will trip the ECSL and break the emulation connection. To allow emulation of secure code, while maintaining the CSM protection against secure memory reads, the user must write the correct value into the lower 64 bits of the KEY register, which matches the value stored in the lower 64 bits of the password locations within the flash. Note that dummy reads of all 128 bits of the password in the flash must still be performed. If the lower 64 bits of the password locations are all ones (unprogrammed), then the KEY value does not need to match.

When initially debugging a device with the password locations in flash programmed (i.e., secured), the emulator takes some time to take control of the CPU. During this time, the CPU will start running and may execute an instruction that performs an access to a protected ECSL area. If this happens, the ECSL will trip and cause the emulator connection to be cut. Two solutions to this problem exist:

1. The first is to use the Wait-In-Reset emulation mode, which will hold the device in reset until the emulator takes control. The emulator must support this mode for this option.
2. The second option is to use the “Branch to check boot mode” boot option. This will sit in a loop and

continuously poll the boot mode select pins. The user can select this boot mode and then exit this mode once the emulator is connected by re-mapping the PC to another address or by changing the boot mode selection pin to the desired boot mode.

NOTE

- When the code-security passwords are programmed, all addresses between 0x33FF80 and 0x33FFF5 cannot be used as program code or data. These locations must be programmed to 0x0000.
- If the code security feature is not used, addresses 0x33FF80 through 0x33FFEF may be used for code or data. Addresses 0x33FFF0 – 0x33FFF5 are reserved for data and should not contain program code. .

The 128-bit password (at 0x33 FFF8 – 0x33 FFFF) must not be programmed to zeros. Doing so would permanently lock the device.

disclaimer

Code Security Module Disclaimer

THE CODE SECURITY MODULE (CSM) INCLUDED ON THIS DEVICE WAS DESIGNED TO PASSWORD PROTECT THE DATA STORED IN THE ASSOCIATED MEMORY (EITHER ROM OR FLASH) AND IS WARRANTED BY TEXAS INSTRUMENTS (TI), IN ACCORDANCE WITH ITS STANDARD TERMS AND CONDITIONS, TO CONFORM TO TI'S PUBLISHED SPECIFICATIONS FOR THE WARRANTY PERIOD APPLICABLE FOR THIS DEVICE.

TI DOES NOT, HOWEVER, WARRANT OR REPRESENT THAT THE CSM CANNOT BE COMPROMISED OR BREACHED OR THAT THE DATA STORED IN THE ASSOCIATED MEMORY CANNOT BE ACCESSED THROUGH OTHER MEANS. MOREOVER, EXCEPT AS SET FORTH ABOVE, TI MAKES NO WARRANTIES OR REPRESENTATIONS CONCERNING THE CSM OR OPERATION OF THIS DEVICE, INCLUDING ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

IN NO EVENT SHALL TI BE LIABLE FOR ANY CONSEQUENTIAL, SPECIAL, INDIRECT, INCIDENTAL, OR PUNITIVE DAMAGES, HOWEVER CAUSED, ARISING IN ANY WAY OUT OF YOUR USE OF THE CSM OR THIS DEVICE, WHETHER OR NOT TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. EXCLUDED DAMAGES INCLUDE, BUT ARE NOT LIMITED TO LOSS OF DATA, LOSS OF GOODWILL, LOSS OF USE OR INTERRUPTION OF BUSINESS OR OTHER ECONOMIC LOSS.

3.2.11 Peripheral Interrupt Expansion (PIE) Block

The PIE block serves to multiplex numerous interrupt sources into a smaller set of interrupt inputs. The PIE block can support up to 96 peripheral interrupts. On the F2833x, 58 of the possible 96 interrupts are used by peripherals. The 96 interrupts are grouped into blocks of 8 and each group is fed into 1 of 12 CPU interrupt lines (INT1 to INT12). Each of the 96 interrupts is supported by its own vector stored in a dedicated RAM block that can be overwritten by the user. The vector is automatically fetched by the CPU on servicing the interrupt. It takes 8 CPU clock cycles to fetch the vector and save critical CPU registers. Hence the CPU can quickly respond to interrupt events. Prioritization of interrupts is controlled in hardware and software. Each individual interrupt can be enabled/disabled within the PIE block.

3.2.12 External Interrupts (XINT1-XINT7, XNMI)

The F2833x supports eight masked external interrupts (XINT1-XINT7, XNMI). XNMI can be connected to the INT13 or NMI interrupt of the CPU. Each of the interrupts can be selected for negative, positive, or both negative and positive edge triggering and can also be enabled/disabled (including the XNMI). XINT1, XINT2, and XNMI also contain a 16-bit free running up counter, which is reset to zero when a valid interrupt edge is detected. This counter can be used to accurately time stamp the interrupt. Unlike the 281x devices, there are no dedicated pins for the external interrupts. XINT1 XINT2, and XNMI interrupts can accept inputs from GPIO0 – GPIO31 pins. XINT3 – XINT7 interrupts can accept inputs from GPIO32 – GPIO63 pins.

3.2.13 Oscillator and PLL

The F2833x can be clocked by an external oscillator or by a crystal attached to the on-chip oscillator circuit. A PLL is provided supporting up to 10 input-clock-scaling ratios. The PLL ratios can be changed on-the-fly in software, enabling the user to scale back on operating frequency if lower power operation is desired. Refer to the Electrical Specification section for timing details. The PLL block can be set in bypass mode.

3.2.14 Watchdog

The F2833x devices contain a watchdog timer. The user software must regularly reset the watchdog counter within a certain time frame; otherwise, the watchdog will generate a reset to the processor. The watchdog can be disabled if necessary.

3.2.15 Peripheral Clocking

The clocks to each individual peripheral can be enabled/disabled so as to reduce power consumption when a peripheral is not in use. Additionally, the system clock to the serial ports (except I2C and eCAN) and the ADC blocks can be scaled relative to the CPU clock. This enables the timing of peripherals to be decoupled from increasing CPU clock speeds.

3.2.16 Low-Power Modes

The F2833x devices are full static CMOS devices. Three low-power modes are provided:

IDLE:	Place CPU into low-power mode. Peripheral clocks may be turned off selectively and only those peripherals that need to function during IDLE are left operating. An enabled interrupt from an active peripheral or the watchdog timer will wake the processor from IDLE mode.
STANDBY:	Turns off clock to CPU and peripherals. This mode leaves the oscillator and PLL functional. An external interrupt event will wake the processor and the peripherals. Execution begins on the next valid cycle after detection of the interrupt event
HALT:	Turns off the internal oscillator. This mode basically shuts down the device and places it in the lowest possible power consumption mode. A reset or external signal can wake the device from this mode.

3.2.17 Peripheral Frames 0, 1, 2, 3 (PFn)

The F2833x device segregates peripherals into three sections. The mapping of peripherals is as follows:

PF0:	PIE:	PIE Interrupt Enable and Control Registers Plus PIE Vector Table
	Flash:	Flash Waitstate Registers
	XINTF:	External Interface Registers
	DMA	DMA Registers
	Timers:	CPU-Timers 0, 1, 2 Registers
	CSM:	Code Security Module KEY Registers
	ADC:	ADC Result Registers (dual-mapped)

PF1:	eCAN:	eCAN Mailbox and Control Registers
	GPIO:	GPIO MUX Configuration and Control Registers
	ePWM:	Enhanced Pulse Width Modulator Module and Registers
	eCAP:	Enhanced Capture Module and Registers
	eQEP:	Enhanced Quadrature Encoder Pulse Module and Registers
PF2:	SYS:	System Control Registers
	SCI:	Serial Communications Interface (SCI) Control and RX/TX Registers
	SPI:	Serial Port Interface (SPI) Control and RX/TX Registers
	ADC:	ADC Status, Control, and Result Register
	I2C:	Inter-Integrated Circuit Module and Registers
	XINT	External Interrupt Registers
PF3:	McBSP	Multichannel Buffered Serial Port Registers

3.2.18 General-Purpose Input/Output (GPIO) Multiplexer

Most of the peripheral signals are multiplexed with general-purpose input/output (GPIO) signals. This enables the user to use a pin as GPIO if the peripheral signal or function is not used. On reset, GPIO pins are configured as inputs. The user can individually program each pin for GPIO mode or peripheral signal mode. For specific inputs, the user can also select the number of input qualification cycles. This is to filter unwanted noise glitches. The GPIO signals can also be used to bring the device out of specific low-power modes.

3.2.19 32-Bit CPU-Timers (0, 1, 2)

CPU-Timers 0, 1, and 2 are identical 32-bit timers with presettable periods and with 16-bit clock prescaling. The timers have a 32-bit count down register, which generates an interrupt when the counter reaches zero. The counter is decremented at the CPU clock speed divided by the prescale value setting. When the counter reaches zero, it is automatically reloaded with a 32-bit period value. CPU-Timer 2 is reserved for Real-Time OS (RTOS)/BIOS applications. It is connected to INT14 of the CPU. CPU-Timer 2 is reserved for the DSP/BIOS Real-Time OS, and is connected to INT14 of the CPU. If DSP/BIOS is not being used, CPU-Timer 2 is available for general use. CPU-Timer 1 is for general use and can be connected to INT13 of the CPU. CPU-Timer 0 is also for general use and is connected to the PIE block.

3.2.20 Control Peripherals

The F2833x devices support the following peripherals which are used for embedded control and communication:

ePWM:	The enhanced PWM peripheral supports independent/complementary PWM generation, adjustable dead-band generation for leading/trailing edges, latched/cycle-by-cycle trip mechanism. Some of the PWM pins support HRPWM features.
eCAP:	The enhanced capture peripheral uses a 32-bit time base and registers up to four programmable events in continuous/one-shot capture modes. This peripheral can also be configured to generate an auxiliary PWM signal.
eQEP:	The enhanced QEP peripheral uses a 32-bit position counter, supports low-speed measurement using capture unit and high-speed measurement using a 32-bit unit timer. This peripheral has a watchdog timer to detect motor stall and input error detection logic to identify simultaneous edge transition in QEP signals.
ADC:	The ADC block is a 12-bit converter, single ended, 16-channels. It contains two sample-and-hold units for simultaneous sampling.

3.2.21 Serial Port Peripherals

The F2833x devices support the following serial communication peripherals:

eCAN:	This is the enhanced version of the CAN peripheral. It supports 32 mailboxes, time stamping of messages, and is CAN 2.0B-compliant.
McBSP:	The multichannel buffered serial port (McBSP) connects to E1/T1 lines, phone-quality codecs for modem applications or high-quality stereo audio DAC devices. The McBSP receive and transmit registers are supported by the DMA to significantly reduce the overhead for servicing this peripheral. Each McBSP module can be configured as an SPI as required.
SPI:	The SPI is a high-speed, synchronous serial I/O port that allows a serial bit stream of programmed length (one to sixteen 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 DSC 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. Multi-device communications are supported by the master/slave operation of the SPI. On the F2833x, the SPI contains a 16-level receive and transmit FIFO for reducing interrupt servicing overhead.
SCI:	The serial communications interface is a two-wire asynchronous serial port, commonly known as UART. On the F2833x, the SCI contains a 16-level receive and transmit FIFO for reducing interrupt servicing overhead.
I2C:	The inter-integrated circuit (I2C) module provides an interface between a DSC and other devices compliant with Philips Semiconductors Inter-IC bus (I2C-bus) specification version 2.1 and connected by way of an I2C-bus. External components attached to this 2-wire serial bus can transmit/receive up to 8-bit data to/from the DSC through the I2C module. On the F2833x, the I2C contains a 16-level receive and transmit FIFO for reducing interrupt servicing overhead.

3.3 Register Map

The F2833x devices contain four peripheral register spaces. The spaces are categorized as follows:

Peripheral Frame 0:	These are peripherals that are mapped directly to the CPU memory bus. See Table 3-7
Peripheral Frame 1	These are peripherals that are mapped to the 32-bit peripheral bus. See Table 3-8
Peripheral Frame 2:	These are peripherals that are mapped to the 16-bit peripheral bus. See Table 3-9
Peripheral Frame 3:	These are peripherals that are mapped to the 32-bit DMA-accessible peripheral bus. See Table 3-10

Table 3-7. Peripheral Frame 0 Registers[1]

NAME	ADDRESS RANGE	SIZE (×16)	ACCESS TYPE[2]
Device Emulation Registers	0x00 0880 - 0x00 09FF	384	EALLOW protected
FLASH Registers[3]	0x00 0A80 - 0x00 0ADF	96	EALLOW protected
Code Security Module Registers	0x00 0AE0 - 0x00 0AEF	16	EALLOW protected

(1) Registers in Frame 0 support 16-bit and 32-bit accesses.
(2) If registers are EALLOW protected, then writes cannot be performed until the EALLOW instruction is executed. The EDIS instruction disables writes to prevent stray code or pointers from corrupting register contents.
(3) The Flash Registers are also protected by the Code Security Module (CSM).

Table 3-7. Peripheral Frame 0 Registers (continued)

NAME	ADDRESS RANGE	SIZE (×16)	ACCESS TYPE[2]
ADC registers (dual-mapped) 0 wait (DMA), 1 wait (CPU), read only	0x00 0B00 - 0x00 0B0F	16	Not EALLOW protected
XINTF Registers	0x00 0B20 - 0x00 0B3F	32	Not EALLOW protected
CPU–TIMER0/1/2 Registers	0x00 0C00 - 0x00 0C3F	64	Not EALLOW protected
PIE Registers	0x00 0CE0 - 0x00 0CFF	32	Not EALLOW protected
PIE Vector Table	0x00 0D00 - 0x00 0DFF	256	EALLOW protected
DMA Registers	0x00 1000 - 0x00 11FF	512	EALLOW protected

Table 3-8. Peripheral Frame 1 Registers

NAME	ADDRESS RANGE	SIZE (×16)
ECAN-A Registers	0x0000 6000 - 0x0000 61FF	512
ECAN-B Registers	0x0000 6200 - 0x0000 63FF	512
EPWM1 + HRPWM1 Registers	0x0000 6800 - 0x0000 683F	64
EPWM2 + HRPWM2 Registers	0x0000 6840 - 0x0000 687F	64
EPWM3 + HRPWM3 Registers	0x0000 6880 - 0x0000 68BF	64
EPWM4 + HRPWM4 Registers	0x0000 68C0 - 0x0000 68FF	64
EPWM5 + HRPWM5 Registers	0x0000 6900 - 0x0000 693F	64
EPWM6 + HRPWM6 Registers	0x0000 6940 - 0x0000 697F	64
ECAP1 Registers	0x0000 6A00 - 0x0000 6A1F	32
ECAP2 Registers	0x0000 6A20 - 0x0000 6A3F	32
ECAP3 Registers	0x0000 6A40 - 0x0000 6A5F	32
ECAP4 Registers	0x0000 6A60 - 0x0000 6A7F	32
ECAP5 Registers	0x0000 6A80 - 0x0000 6A9F	32
ECAP6 Registers	0x0000 6AA0 - 0x0000 6ABF	32
EQEP1 Registers	0x0000 6B00 - 0x0000 6B3F	64
EQEP2 Registers	0x0000 6B40 - 0x0000 6B7F	64
GPIO Registers	0x0000 6F80 - 0x0000 6FFF	128

ADVANCE INFORMATION

Table 3-9. Peripheral Frame 2 Registers

NAME	ADDRESS RANGE	SIZE (×16)
System Control Registers	0x0000 7010 - 0x0000 702F	32
SPI-A Registers	0x0000 7040 - 0x0000 704F	16
SCI-A Registers	0x0000 7050 - 0x0000 705F	16
External Interrupt Registers	0x0000 7070 - 0x0000 707F	16
ADC Registers	0x0000 7100 - 0x0000 711F	32
SCI-B Registers	0x0000 7750 - 0x0000 775F	16
SCI-C Registers	0x0000 7770 - 0x0000 777F	16
I2C-A Registers	0x0000 7900 - 0x0000 793F	64

Table 3-10. Peripheral Frame 3 Registers

NAME	ADDRESS RANGE	SIZE (×16)
McBSP-A Registers	0x0000 5000 - 0x0000 503F	64
McBSP-B Registers	0x0000 5040 - 0x0000 507F	64

3.4 Device Emulation Registers

These registers are used to control the protection mode of the C28x CPU and to monitor some critical device signals. The registers are defined in Table 3-11.

Table 3-11. Device Emulation Registers

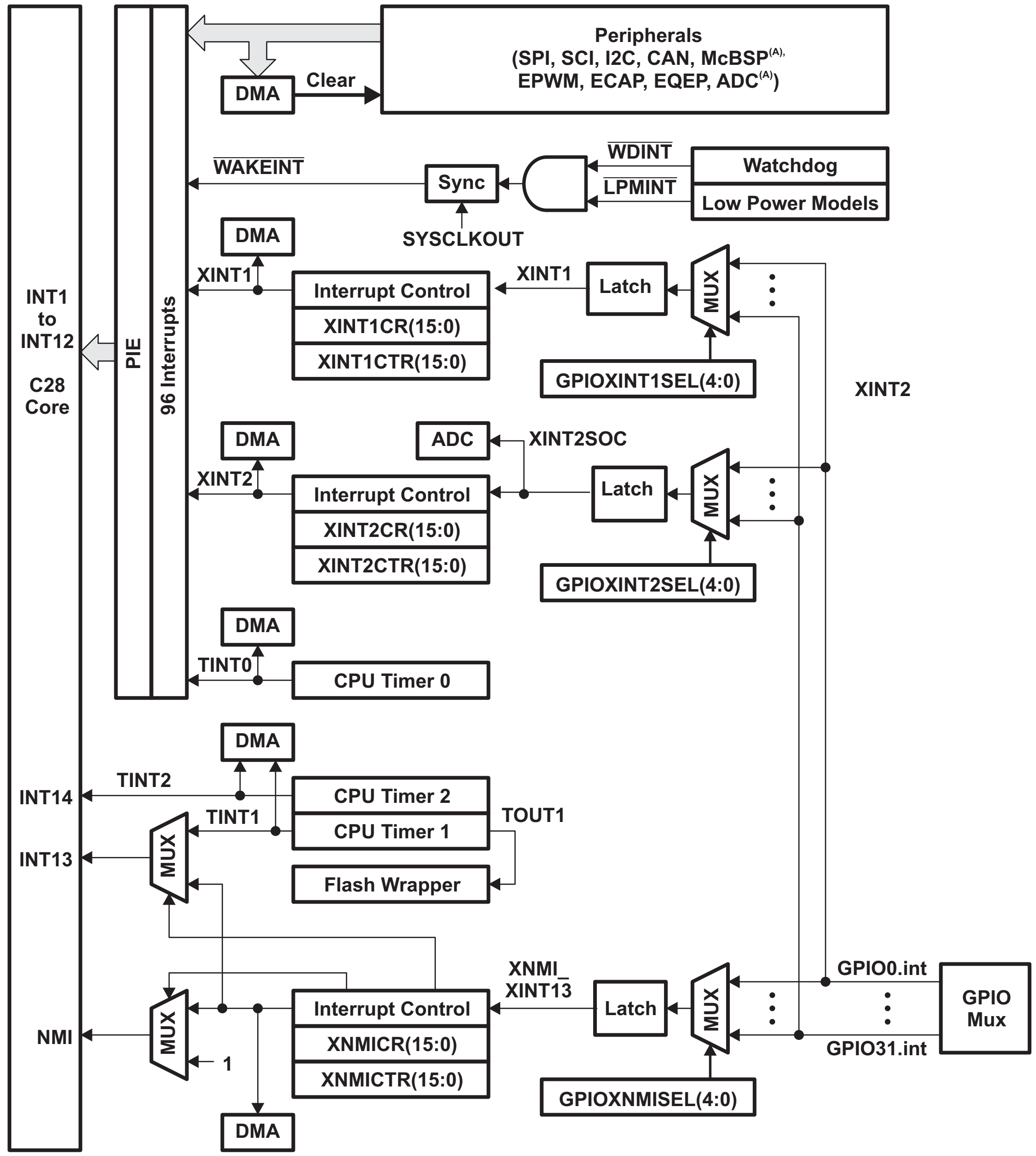
NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION	
DEVICECNF	0x0880 0x0881	2	Device Configuration Register	
PARTID	0x0882	1	Part ID Register	0x00F8[1] - F28332 0x00F9 - F28334 0x00FA - F28335
REVID	0x0883	1	Revision ID Register	0x0000 - Silicon Rev. 0 - TMX
PROTSTART	0x0884	1	Block Protection Start Address Register	
PROTRANGE	0x0885	1	Block Protection Range Address Register	

(1) The first byte (00) denotes flash devices. FF denotes ROM devices. Other values are reserved for future devices.

ADVANCE INFORMATION

3.5 Interrupts

Figure 3-5 shows how the various interrupt sources are multiplexed within the F2833x devices.



A. DMA-accessible

Figure 3-5. External and PIE Interrupt Sources

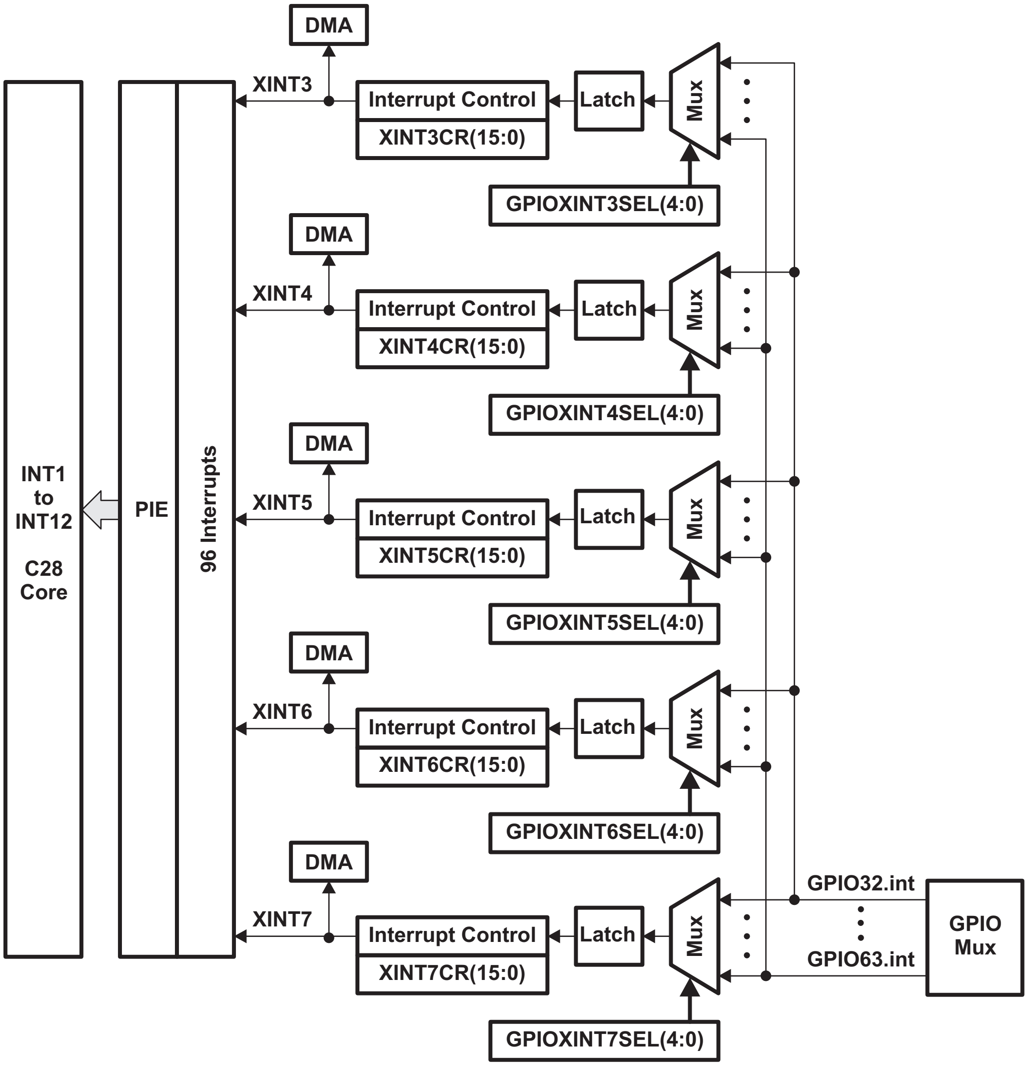


Figure 3-6. External Interrupts

Eight PIE block interrupts are grouped into one CPU interrupt. In total, 12 CPU interrupt groups, with 8 interrupts per group equals 96 possible interrupts. On the F2833x, 58 of these are used by peripherals as shown in Table 3-12.

The TRAP #VectorNumber instruction transfers program control to the interrupt service routine corresponding to the vector specified. TRAP #0 attempts to transfer program control to the address pointed to by the reset vector. The PIE vector table does not, however, include a reset vector. Therefore, TRAP #0 should not be used when the PIE is enabled. Doing so will result in undefined behavior.

When the PIE is enabled, TRAP #1 through TRAP #12 will transfer program control to the interrupt service routine corresponding to the first vector within the PIE group. For example: TRAP #1 fetches the vector from INT1.1, TRAP #2 fetches the vector from INT2.1, and so forth.

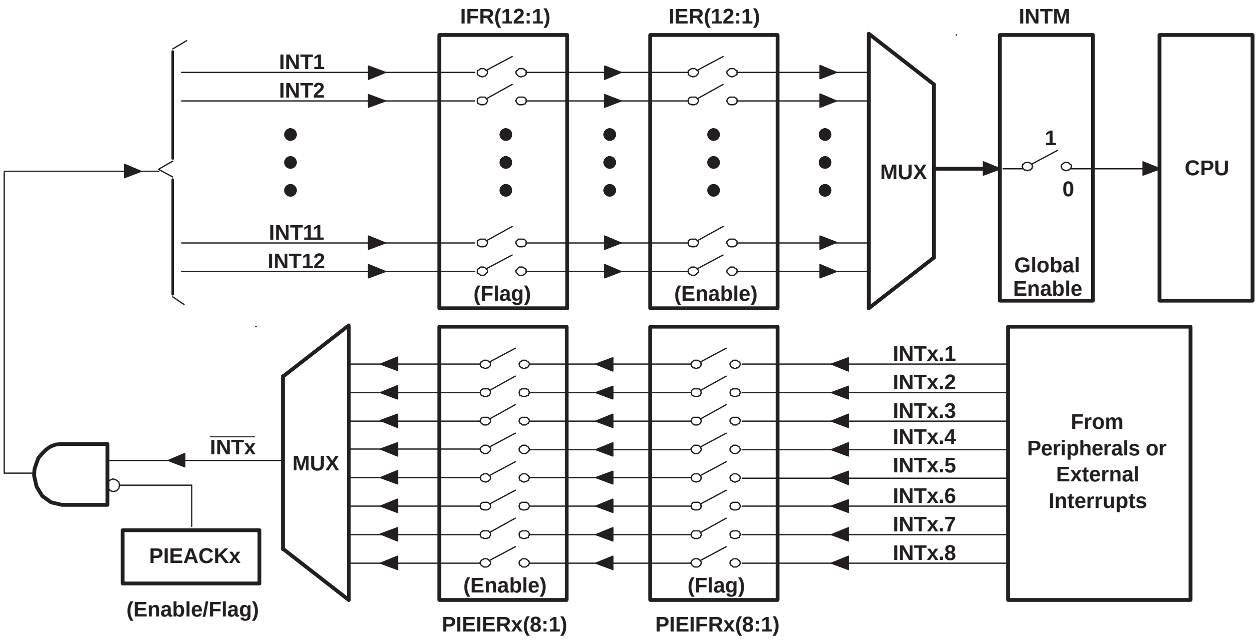


Figure 3-7. Multiplexing of Interrupts Using the PIE Block

Table 3-12. PIE Peripheral Interrupts[1]

CPU INTERRUPTS	PIE INTERRUPTS							
	INTx.8	INTx.7	INTx.6	INTx.5	INTx.4	INTx.3	INTx.2	INTx.1
INT1	WAKEINT (LPM/WD)	TINT0 (TIMER 0)	ADCINT (ADC)	XINT2	XINT1	Reserved	SEQ2INT (ADC)	SEQ1INT (ADC)
INT2	Reserved	Reserved	EPWM6_TZINT (ePWM6)	EPWM5_TZINT (ePWM5)	EPWM4_TZINT (ePWM4)	EPWM3_TZINT (ePWM3)	EPWM2_TZINT (ePWM2)	EPWM1_TZINT (ePWM1)
INT3	Reserved	Reserved	EPWM6_INT (ePWM6)	EPWM5_INT (ePWM5)	EPWM4_INT (ePWM4)	EPWM3_INT (ePWM3)	EPWM2_INT (ePWM2)	EPWM1_INT (ePWM1)
INT4	Reserved	Reserved	ECAP6_INT (ECAP6)	ECAP5_INT (ECAP5)	ECAP4_INT (eCAP4)	ECAP3_INT (eCAP3)	ECAP2_INT (eCAP2)	ECAP1_INT (eCAP1)
INT5	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	EQEP2_INT (eQEP2)	EQEP1_INT (eQEP1)
INT6	Reserved	Reserved	MXINTA (McBSP-A)	MRINTA (McBSP-A)	MXINTB (McBSP-B)	MRINTB (McBSP-B)	SPITXINTA (SPI-A)	SPIRXINTA (SPI-A)
INT7	Reserved	Reserved	DINTCH6 (DMA)	DINTCH5 (DMA)	DINTCH4 (DMA)	DINTCH3 (DMA)	DINTCH2 (DMA)	DINTCH1 (DMA)
INT8	Reserved	Reserved	SCITXINTC (SCI-C)	SCIRXINTC (SCI-C)	Reserved	Reserved	I2CINT2A (I2C-A)	I2CINT1A (I2C-A)
INT9	ECAN1_INTB (CAN-B)	ECAN0_INTB (CAN-B)	ECAN1_INTA (CAN-A)	ECAN0_INTA (CAN-A)	SCITXINTB (SCI-B)	SCIRXINTB (SCI-B)	SCITXINTA (SCI-A)	SCIRXINTA (SCI-A)
INT10	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
INT11	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved
INT12	LUF (FPU)	LVF (FPU)	Reserved	XINT7	XINT6	XINT5	XINT4	XINT3

(1) Out of the 96 possible interrupts, 58 interrupts are currently used. The remaining interrupts are reserved for future devices. These interrupts can be used as software interrupts if they are enabled at the PIEIFRx level, provided none of the interrupts within the group is being used by a peripheral. Otherwise, interrupts coming in from peripherals may be lost by accidentally clearing their flag while modifying the PIEIFR. To summarize, there are two safe cases when the reserved interrupts could be used as software interrupts:
1) No peripheral within the group is asserting interrupts.
2) No peripheral interrupts are assigned to the group (example PIE group 11).

Table 3-13. PIE Configuration and Control Registers

NAME	ADDRESS	SIZE (X16)	DESCRIPTION[1]
PIECTRL	0x0CE0	1	PIE, Control Register
PIEACK	0x0CE1	1	PIE, Acknowledge Register
PIEIER1	0x0CE2	1	PIE, INT1 Group Enable Register
PIEIFR1	0x0CE3	1	PIE, INT1 Group Flag Register
PIEIER2	0x0CE4	1	PIE, INT2 Group Enable Register
PIEIFR2	0x0CE5	1	PIE, INT2 Group Flag Register
PIEIER3	0x0CE6	1	PIE, INT3 Group Enable Register
PIEIFR3	0x0CE7	1	PIE, INT3 Group Flag Register
PIEIER4	0x0CE8	1	PIE, INT4 Group Enable Register
PIEIFR4	0x0CE9	1	PIE, INT4 Group Flag Register
PIEIER5	0x0CEA	1	PIE, INT5 Group Enable Register
PIEIFR5	0x0CEB	1	PIE, INT5 Group Flag Register
PIEIER6	0x0CEC	1	PIE, INT6 Group Enable Register
PIEIFR6	0x0CED	1	PIE, INT6 Group Flag Register
PIEIER7	0x0CEE	1	PIE, INT7 Group Enable Register
PIEIFR7	0x0CEF	1	PIE, INT7 Group Flag Register
PIEIER8	0x0CF0	1	PIE, INT8 Group Enable Register
PIEIFR8	0x0CF1	1	PIE, INT8 Group Flag Register
PIEIER9	0x0CF2	1	PIE, INT9 Group Enable Register
PIEIFR9	0x0CF3	1	PIE, INT9 Group Flag Register
PIEIER10	0x0CF4	1	PIE, INT10 Group Enable Register
PIEIFR10	0x0CF5	1	PIE, INT10 Group Flag Register
PIEIER11	0x0CF6	1	PIE, INT11 Group Enable Register
PIEIFR11	0x0CF7	1	PIE, INT11 Group Flag Register
PIEIER12	0x0CF8	1	PIE, INT12 Group Enable Register
PIEIFR12	0x0CF9	1	PIE, INT12 Group Flag Register
Reserved	0x0CFA 0x0CFF	6	Reserved

(1) The PIE configuration and control registers are not protected by EALLOW mode. The PIE vector table is protected.

3.5.1 External Interrupts

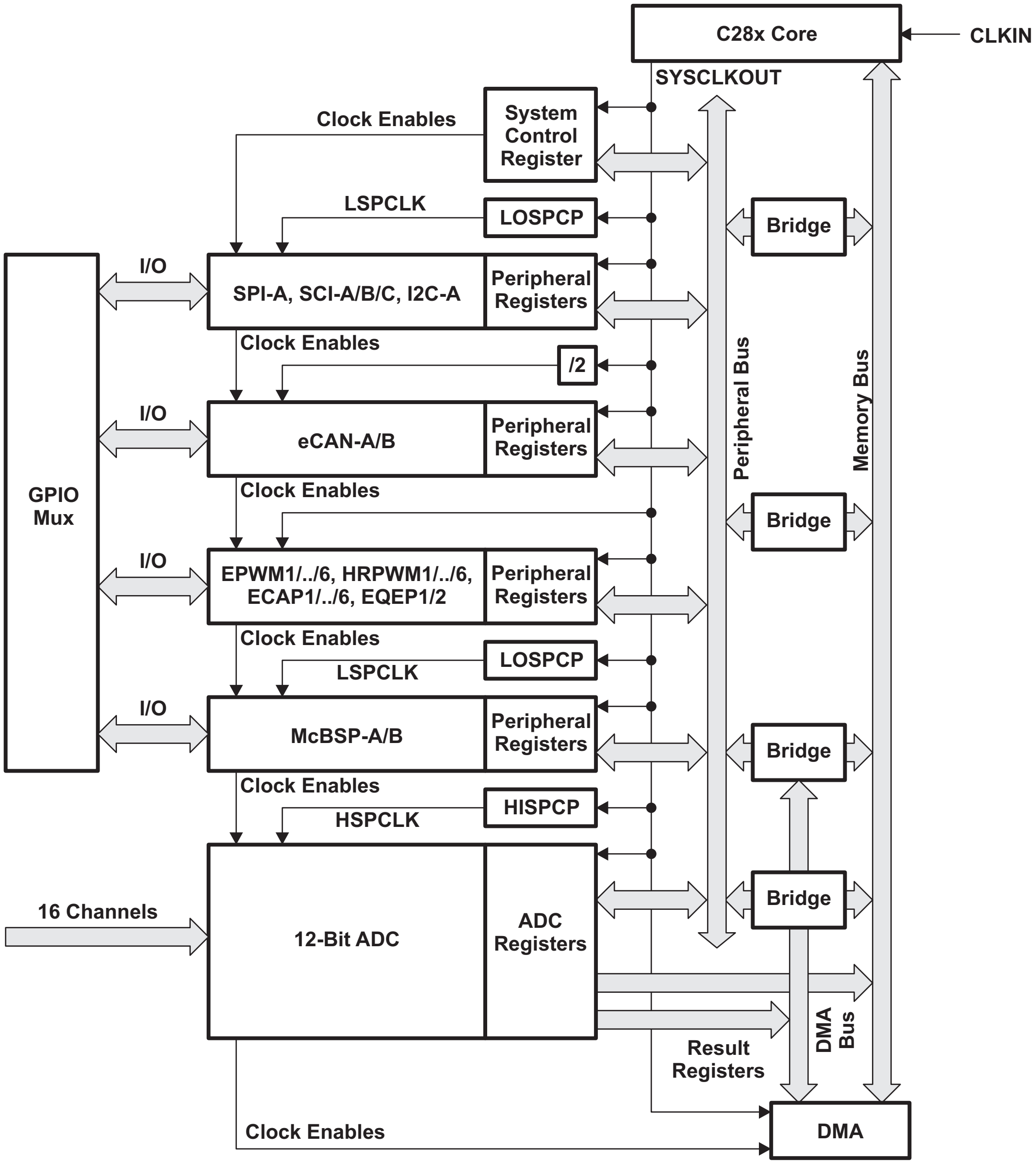
Table 3-14. External Interrupt Registers

Name	Address	Size (x16)	Description
XINT1CR	0x0000 7070	1	XINT1 configuration register
XINT2CR	0x0000 7071	1	XINT2 configuration register
XINT3CR	0x0000 7072	1	XINT3 configuration register
XINT4CR	0x0000 7073	1	XINT4 configuration register
XINT5CR	0x0000 7074	1	XINT5 configuration register
XINT6CR	0x0000 7075	1	XINT6 configuration register
XINT7CR	0x0000 7076	1	XINT7 configuration register
XNMICR	0x0000 7077	1	XNMI configuration register
XINT1CTR	0x0000 7078	1	XINT1 counter register
XINT2CTR	0x0000 7079	1	XINT2 counter register
Reserved	0x707A - 0x707E	5	
XNMICTR	0x0000 707F	1	XNMI counter register

Each external interrupt can be enabled/disabled or qualified using positive, negative, or both positive and negative edge. For more information, see the *TMS320F2833x Digital Signal Controller (DSC) System and Interrupts Reference Guide* (literature number SPRUFB0).

3.6 System Control

This section describes the F2833x oscillator, PLL and clocking mechanisms, the watchdog function and the low power modes. Figure 3-8 shows the various clock and reset domains in the F2833x devices that will be discussed.



A. CLKIN is the clock into the CPU. It is passed out of the CPU as SYSCLKOUT (that is, CLKIN is the same frequency as SYSCLKOUT). See Figure 3-9 for an illustration of how CLKIN is derived.

Figure 3-8. Clock and Reset Domains

The PLL, clocking, watchdog and low-power modes, are controlled by the registers listed in Table 3-15.

Table 3-15. PLL, Clocking, Watchdog, and Low-Power Mode Registers

Name	Address	Size (x16)	Description
PLLSTS	0x0000-7011	1	PLL Status Register
Reserved	0x0000-7012 - 0x0000-7018	7	
HISPCP	0x0000-701A	1	High-Speed Peripheral Clock Pre-Scaler Register
LOSPCP	0x0000-701B	1	Low-Speed Peripheral Clock Pre-Scaler Register
PCLKCR0	0x0000-701C	1	Peripheral Clock Control Register 0
PCLKCR1	0x0000-701D	1	Peripheral Clock Control Register 1
LPMCR0	0x0000-701E	1	Low Power Mode Control Register 0
Reserved	0x0000-701F	1	Low Power Mode Control Register 1
PCLKCR3	0x0000-7020	1	Peripheral Clock Control Register 3
PLLCR	0x0000-7021	1	PLL Control Register
SCSR	0x0000-7022	1	System Control and Status Register
WDCNTR	0x0000-7023	1	Watchdog Counter Register
Reserved	0x0000-7024	1	
WDKEY	0x0000-7025	1	Watchdog Reset Key Register
Reserved	0x0000-7026 - 0x0000-7028	3	
WDCR	0x0000-7029	1	Watchdog Control Register
Reserved	0x0000-702A - 0x0000-702F	6	

3.6.1 OSC and PLL Block

Figure 3-9 shows the OSC and PLL block on the F2833x.

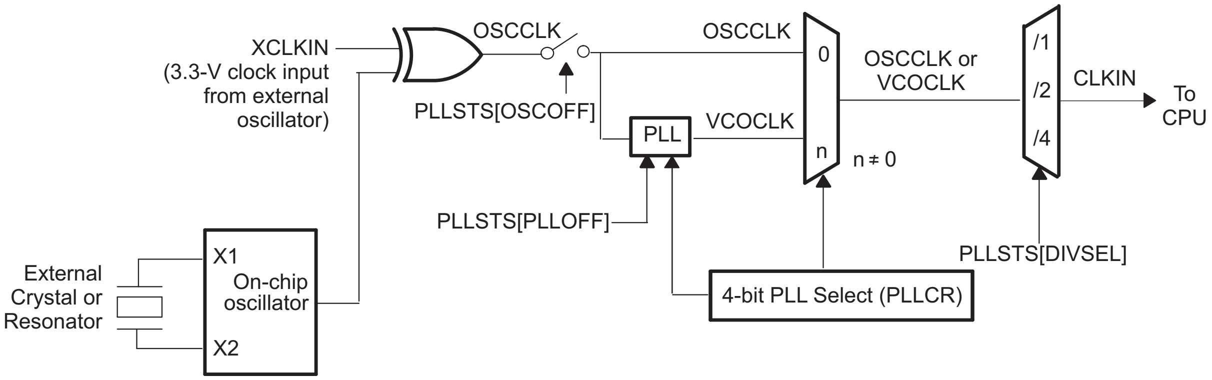


Figure 3-9. OSC and PLL Block Diagram

The on-chip oscillator circuit enables a crystal/resonator to be attached to the F2833x devices using the X1 and X2 pins. If the on-chip oscillator is not used, an external oscillator can be used in either one of the following configurations:

1. A 3.3-V external oscillator can be directly connected to the XCLKIN pin. The X2 pin should be left unconnected and the X1 pin tied low. The logic-high level in this case should not exceed V_{DDIO}.
2. A 1.9-V external oscillator can be directly connected to the X1 pin. The X2 pin should be left unconnected and the XCLKIN pin tied low. The logic-high level in this case should not exceed V_{DD}.

The three possible input-clock configurations are shown in Figure 3-10 through Figure 3-12

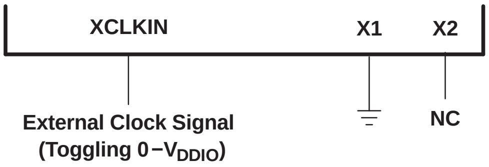


Figure 3-10. Using a 3.3-V External Oscillator

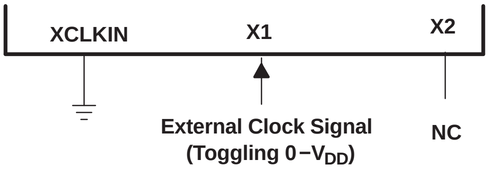


Figure 3-11. Using a 1.9-V External Oscillator

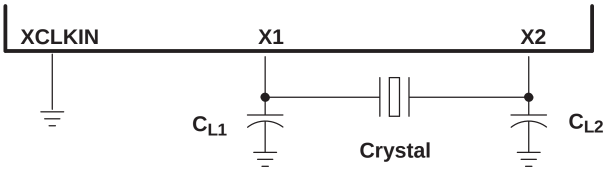


Figure 3-12. Using the Internal Oscillator

3.6.1.1 External Reference Oscillator Clock Option

The typical specifications for the external quartz crystal for a frequency of 20 MHz are listed below:

- Fundamental mode, parallel resonant
- C_L (load capacitance) = 12 pF
- $C_{L1} = C_{L2} = 24$ pF
- $C_{shunt} = 6$ pF
- ESR range = 30 to 60 Ω

TI recommends that customers have the resonator/crystal vendor characterize the operation of their device with the DSC chip. The resonator/crystal vendor has the equipment and expertise to tune the tank circuit. The vendor can also advise the customer regarding the proper tank component values that will produce proper start up and stability over the entire operating range.

3.6.1.2 PLL-Based Clock Module

The F2833x devices have 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 PLL has a 4-bit ratio control PLLCR[DIV] to select different CPU clock rates. The watchdog module should be disabled before writing to the PLLCR register. It can be re-enabled (if need be) after the PLL module has stabilized, which takes 131072 OSCCLK cycles.

Table 3-16. PLLCR[1] Bit Descriptions

PLLCR[DIV] VALUE[2]	SYSCLKOUT (CLKIN)		
	PLLSTS[DIVSEL] = 0 or 1	**PLLSTS[DIVSEL] = 2**	**PLLSTS[DIVSEL] = 3**
0000 (PLL bypass)	OSCCLK/4 (Default)	OSCCLK/2	OSCCLK
0001	(OSCCLK * 1)/4	(OSCCLK*1)/2	OSCCLK*1
0010	(OSCCLK * 2)/4	(OSCCLK*2)/2	OSCCLK*2
0011	(OSCCLK * 3)/4	(OSCCLK*3)/2	OSCCLK*3
0100	(OSCCLK * 4)/4	(OSCCLK*4)/2	OSCCLK*4
0101	(OSCCLK * 5)/4	(OSCCLK*5)/2	OSCCLK*5
0110	(OSCCLK * 6)/4	(OSCCLK*6)/2	OSCCLK*6
0111	(OSCCLK * 7)/4	(OSCCLK*7)/2	OSCCLK*7
1000	(OSCCLK * 8)/4	(OSCCLK*8)/2	OSCCLK*8
1001	(OSCCLK * 9)/4	(OSCCLK*9)/2	OSCCLK*9
1010	(OSCCLK * 10)/4	(OSCCLK*10)/2	OSCCLK*10
1011 - 1111	Reserved	Reserved	Reserved

(1) PLLSTS[DIVSEL] must be 0 before writing to the PLLCR and must be set only to 2 or 3 after PLLSTS[PLLLOCKS] = 1. By default, PLLSTS[DIVSEL] is configured for /4. The boot ROM changes this to /2.

(2) The PLL control register (PLLCR) and PLL Status Register (PLLSTS) are reset to their default state by the $\overline{\text{XRS}}$ signal or a watchdog reset only. A reset issued by the debugger or the missing clock detect logic have no effect.

Table 3-17. CLKIN Divide Options

PLLSTS [DIVSEL]	CLKIN DIVIDE
0	/4
1	/4
2	/2
3	/1

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

- Crystal-operation - This mode allows the use of an external crystal/resonator to provide the time base to the device.
- External clock source operation - This mode allows the internal oscillator to be bypassed. The device clocks are generated from an external clock source input on the X1 or the XCLKIN pin.

Table 3-18. Possible PLL Configuration Modes

PLL MODE	REMARKS	PLLSTS[DIVSEL][1]	CLKIN AND SYSCLKOUT
PLL Off	Invoked by the user setting the PLLOFF bit in the PLLSTS register. The PLL block is disabled in this mode. This can be useful to reduce system noise and for low power operation. The PLLCR register must first be set to 0x0000 (PLL Bypass) before entering this mode. The CPU clock (CLKIN) is derived directly from the input clock on either X1/X2, X1 or XCLKIN.	0, 1 2 3	OSCCLK/4 OSCCLK/2 OSCCLK/1
PLL Bypass	PLL Bypass is the default PLL configuration upon power-up or after an external reset ($\overline{\text{XRS}}$). This mode is selected when the PLLCR register is set to 0x0000 or while the PLL locks to a new frequency after the PLLCR register has been modified. In this mode, the PLL itself is bypassed but the PLL is not turned off.	0, 1 2 3	OSCCLK/4 OSCCLK/2 OSCCLK/1
PLL Enable	Achieved by writing a non-zero value n into the PLLCR register. Upon writing to the PLLCR the device will switch to PLL Bypass mode until the PLL locks.	0, 1 2 3	OSCCLK*n/4 OSCCLK*n/2 OSCCLK*n/1

(1) PLLSTS[DIVSEL] must be 0 before writing to the PLLCR and must only be set to 1 after PLLSTS[PLLLOCKS] = 1. See the *TMS320F2833x Digital Signal Controller (DSC) System Control and Interrupts Reference Guide* (literature Number SPRUFB0) for more information.

3.6.1.3 Loss of Input Clock

In PLL-enabled and PLL-bypass mode, if the input clock OSCCLK is removed or absent, the PLL will still issue a limp-mode clock. The limp-mode clock continues to clock the CPU and peripherals at a typical frequency of 1-5 MHz. Limp mode is not specified to work from power-up, only after input clocks have been present initially. In PLL bypass mode, the limp mode clock from the PLL is automatically routed to the CPU if the input clock is removed or absent.

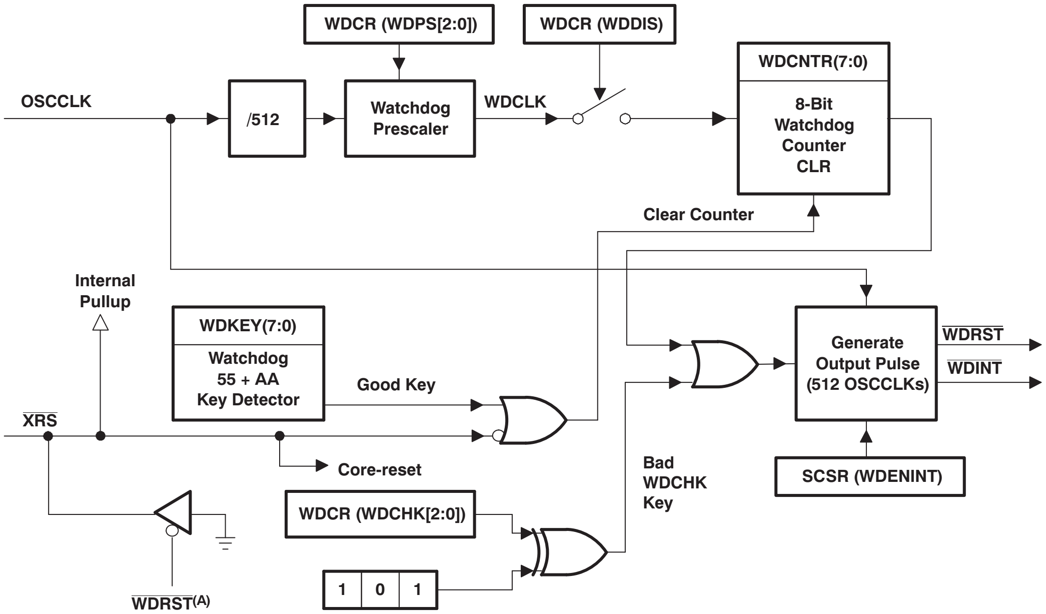
Normally, when the input clocks are present, the watchdog counter decrements to initiate a watchdog reset or WDINT interrupt. However, when the external input clock fails, the watchdog counter stops decrementing (i.e., the watchdog counter does not change with the limp-mode clock). In addition to this, the device will be reset and the “Missing Clock Status” (MCLKSTS) bit will be set. These conditions could be used by the application firmware to detect the input clock failure and initiate necessary shut-down procedure for the system.

NOTE

Applications in which the correct CPU operating frequency is absolutely critical should implement a mechanism by which the DSC will be held in reset, should the input clocks ever fail. For example, an R-C circuit may be used to trigger the $\overline{\text{XRS}}$ pin of the DSC, should the capacitor ever get fully charged. An I/O pin may be used to discharge the capacitor on a periodic basis to prevent it from getting fully charged. Such a circuit would also help in detecting failure of the flash memory and the V_{DD3VFL} rail.

3.6.2 *Watchdog Block*

The watchdog block on the F2833x is similar to the one used on the 240x and 281x devices. The watchdog module generates an output pulse, 512 oscillator clocks wide (OSCCLK), whenever the 8-bit watchdog up counter has reached its maximum value. To prevent this, the user disables the counter or the software must periodically write a 0x55 + 0xAA sequence into the watchdog key register which will reset the watchdog counter. Figure 3-13 shows the various functional blocks within the watchdog module.



A. The $\overline{\text{WDRST}}$ signal is driven low for 512 OSCCLK cycles.

Figure 3-13. Watchdog Module

The $\overline{\text{WDINT}}$ signal enables the watchdog to be used as a wakeup from IDLE/STANDBY mode.

In STANDBY mode, all peripherals are turned off on the device. The only peripheral that remains functional is the watchdog. The WATCHDOG module will run off OSCCLK. The $\overline{\text{WDINT}}$ signal is fed to the LPM block so that it can wake the device from STANDBY (if enabled). See Section Section 3.7, Low-Power Modes Block, for more details.

In IDLE mode, the $\overline{\text{WDINT}}$ signal can generate an interrupt to the CPU, via the PIE, to take the CPU out of IDLE mode.

In HALT mode, this feature cannot be used because the oscillator (and PLL) are turned off and hence so is the WATCHDOG.

3.7 Low-Power Modes Block

The low-power modes on the F2833x are similar to the 240x devices. Table 3-19 summarizes the various modes.

Table 3-19. Low-Power Modes

MODE	LPMCR0(1:0)	OSCCLK	CLKIN	SYSCLKOUT	EXIT(1)
IDLE	00	On	On	On(2)	$\overline{\text{XRS}}$, Watchdog interrupt, any enabled interrupt, XNMI
STANDBY	01	On (watchdog still running)	Off	Off	$\overline{\text{XRS}}$, Watchdog interrupt, GPIO Port A signal, debugger(3), XNMI
HALT	1X	Off (oscillator and PLL turned off, watchdog not functional)	Off	Off	$\overline{\text{XRS}}$, GPIO Port A signal, XNMI, debugger(3)

(1) The Exit column lists which signals or under what conditions the low power mode will be exited. A low signal, on any of the signals, will exit the low power condition. This signal must be kept low long enough for an interrupt to be recognized by the device. Otherwise the IDLE mode will not be exited and the device will go back into the indicated low power mode.

(2) The IDLE mode on the C28x behaves differently than on the 24x/240x. On the C28x, the clock output from the CPU (SYSCLKOUT) is still functional while on the 24x/240x the clock is turned off.

(3) On the C28x, the JTAG port can still function even if the CPU clock (CLKIN) is turned off.

The various low-power modes operate as follows:

IDLE Mode:	This mode is exited by any enabled interrupt or an XNMI that is recognized by the processor. The LPM block performs no tasks during this mode as long as the LPMCR0(LPM) bits are set to 0,0.
STANDBY Mode:	Any GPIO port A signal (GPIO[31:0]) can wake the device from STANDBY mode. The user must select which signal(s) will wake the device in the GPIOLPMSEL register. The selected signal(s) are also qualified by the OSCCLK before waking the device. The number of OSCCLKs is specified in the LPMCR0 register.
HALT Mode:	Only the $\overline{\text{XRS}}$ and any GPIO port A signal (GPIO[31:0]) can wake the device from HALT mode. The user selects the signal in the GPIOLPMSEL register.

NOTE

The low-power modes do not affect the state of the output pins (PWM pins included). They will be in whatever state the code left them in when the IDLE instruction was executed. See the *TMS320F2833x Digital Signal Controller (DSC) System and Interrupts Reference Guide* (literature number SPRUFB0) for more details.

4 Peripherals

The integrated peripherals of the F2833x are described in the following subsections:

- 6-channel Direct Memory Access (DMA)
- Three 32-bit CPU-Timers
- Up to six enhanced PWM modules (ePWM1, ePWM2, ePWM3, ePWM4, ePWM5, ePWM6)
- Up to six enhanced capture modules (eCAP1, eCAP2, eCAP3, eCAP4, eCAP5, eCAP6)
- Up to two enhanced QEP modules (eQEP1, eQEP2)
- Enhanced analog-to-digital converter (ADC) module
- Up to two enhanced controller area network (eCAN) modules (eCAN-A, eCAN-B)
- Up to three serial communications interface modules (SCI-A, SCI-B, SCI-C)
- One serial peripheral interface (SPI) module (SPI-A)
- Inter-integrated circuit module (I2C)
- Up to two multichannel buffered serial port (McBSP-A, McBSP-B) modules
- Digital I/O and shared pin functions
- External Interface (XINTF)

4.1 DMA Overview

Features:

- 6 Channels with independent PIE interrupts
- Trigger Sources:
  - ADC Sequencer 1 and Sequencer 2
  - McBSP-A and McBSP-B transmit and receive logic
  - XINT1-7 and XINT13
  - CPU Timers
  - Software
- Data Sources/Destinations:
  - L4-L7 16k x 16 SARAM
  - All XINTF zones
  - ADC Memory Bus mapped RESULT registers
  - McBSP-A and McBSP-B transmit and receive buffers
- Word Size: 16-bit or 32-bit (McBSPs limited to 16-bit)
- Throughput: 4 cycles/word (5 cycles/word for McBSP reads)

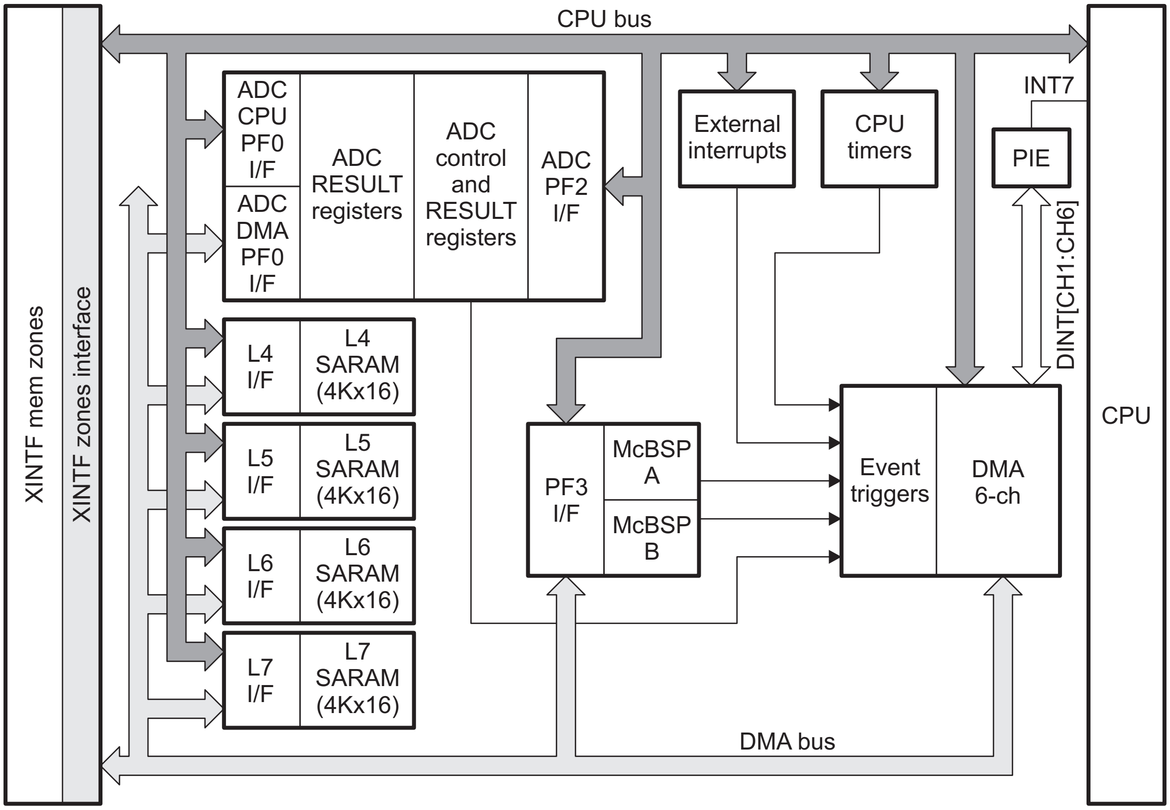


Figure 4-1. DMA Functional Block Diagram

4.2 32-Bit CPU-Timers 0/1/2

There are three 32-bit CPU-timers on the F2833x devices (CPU-TIMER0/1/2).

Timer 2 is reserved for DSP/BIOS™. CPU-Timer 0 and CPU-Timer 1 can be used in user applications. These timers are different from the timers that are present in the ePWM modules.

NOTE

NOTE: If the application is not using DSP/BIOS, then CPU-Timer 2 can be used in the application.

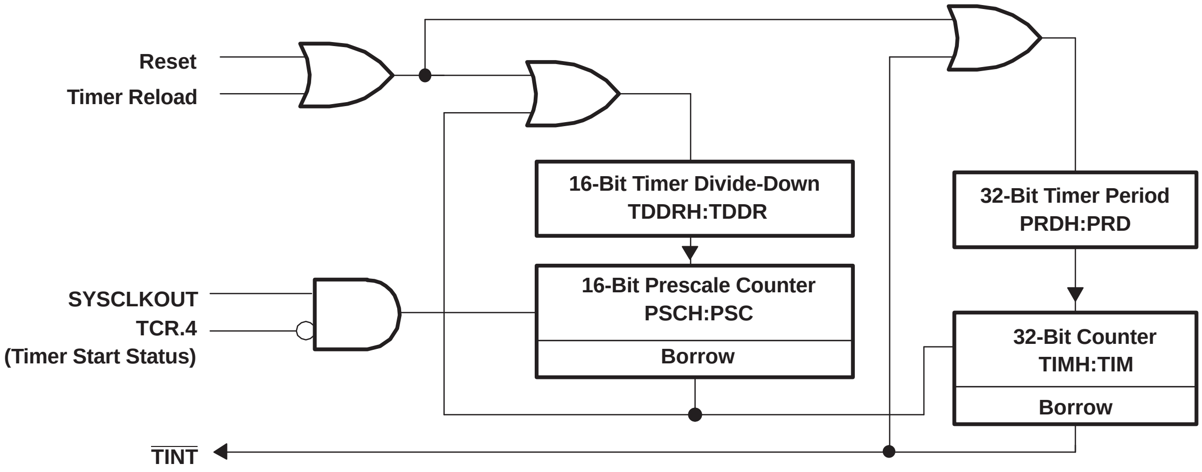
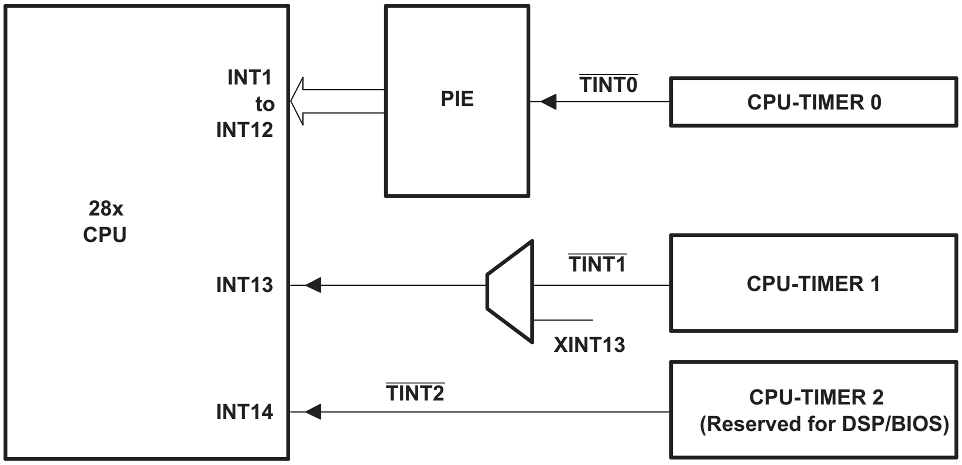


Figure 4-2. CPU-Timers

In the F2833x devices, the timer interrupt signals ($\overline{\text{TINT0}}$, $\overline{\text{TINT1}}$, $\overline{\text{TINT2}}$) are connected as shown in Figure 4-3.



A. The timer registers are connected to the memory bus of the C28x processor.

B. The timing of the timers is synchronized to SYSCLKOUT of the processor clock.

Figure 4-3. CPU-Timer Interrupt Signals and Output Signal

The general operation of the timer is as follows: The 32-bit counter register "TIMH:TIM" is loaded with the value in the period register "PRDH:PRD". The counter register decrements at the SYSCLKOUT rate of the C28x. When the counter reaches 0, a timer interrupt output signal generates an interrupt pulse. The registers listed in Table 4-1 are used to configure the timers. For more information, see the *TMS320F2833x Digital Signal Controller (DSC) System Control and Interrupts Reference Guide* (literature number SPRUFB0)

Table 4-1. CPU-Timers 0, 1, 2 Configuration and Control Registers

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
TIMER0TIM	0x0C00	1	CPU-Timer 0, Counter Register
TIMER0TIMH	0x0C01	1	CPU-Timer 0, Counter Register High
TIMER0PRD	0x0C02	1	CPU-Timer 0, Period Register
TIMER0PRDH	0x0C03	1	CPU-Timer 0, Period Register High
TIMER0TCR	0x0C04	1	CPU-Timer 0, Control Register
Reserved	0x0C05	1	
TIMER0TPR	0x0C06	1	CPU-Timer 0, Prescale Register
TIMER0TPRH	0x0C07	1	CPU-Timer 0, Prescale Register High
TIMER1TIM	0x0C08	1	CPU-Timer 1, Counter Register
TIMER1TIMH	0x0C09	1	CPU-Timer 1, Counter Register High
TIMER1PRD	0x0C0A	1	CPU-Timer 1, Period Register
TIMER1PRDH	0x0C0B	1	CPU-Timer 1, Period Register High
TIMER1TCR	0x0C0C	1	CPU-Timer 1, Control Register
Reserved	0x0C0D	1	
TIMER1TPR	0x0C0E	1	CPU-Timer 1, Prescale Register
TIMER1TPRH	0x0C0F	1	CPU-Timer 1, Prescale Register High
TIMER2TIM	0x0C10	1	CPU-Timer 2, Counter Register
TIMER2TIMH	0x0C11	1	CPU-Timer 2, Counter Register High
TIMER2PRD	0x0C12	1	CPU-Timer 2, Period Register
TIMER2PRDH	0x0C13	1	CPU-Timer 2, Period Register High
TIMER2TCR	0x0C14	1	CPU-Timer 2, Control Register
Reserved	0x0C15	1	
TIMER2TPR	0x0C16	1	CPU-Timer 2, Prescale Register
TIMER2TPRH	0x0C17	1	CPU-Timer 2, Prescale Register High
Reserved	0x0C18 0x0C3F	40	

4.3 Enhanced PWM Modules (ePWM1/2/3/4/5/6)

The F2833x device contains up to six enhanced PWM Modules (ePWM). Figure 4-4 shows a block diagram of multiple ePWM modules. Figure 4-4 shows the signal interconnections with the ePWM. See the *TMS320x28xx, 28xxx Enhanced Pulse Width Modulator (ePWM) Module Reference Guide* (literature number SPRU791) for more details.

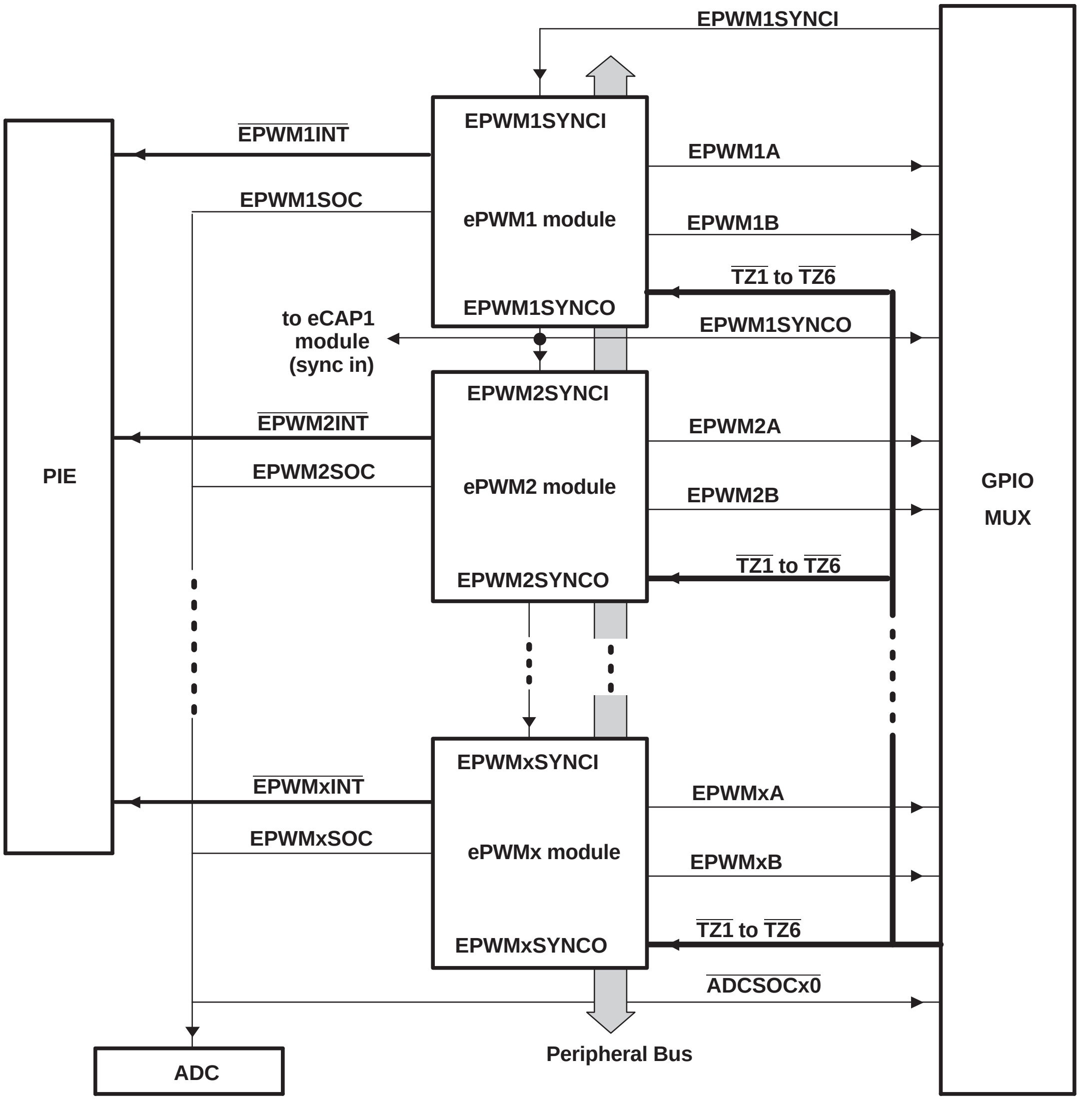


Figure 4-4. Multiple PWM Modules in a F2833x System

Table 4-2 shows the complete ePWM register set per module.

Table 4-2. ePWM Control and Status Registers

NAME	EPWM1	EPWM2	EPWM3	EPWM4	EPWM5	EPWM6	SIZE (x16) / #SHADOW	DESCRIPTION
TBCTL	0x6800	0x6840	0x6880	0x68C0	0x6900	0x6940	1 / 0	Time Base Control Register
TBSTS	0x6801	0x6841	0x6881	0x68C1	0x6901	0x6941	1 / 0	Time Base Status Register
TBPHSHR	0x6802	0x6842	0x6882	0x68C2	0x6902	0x6942	1 / 0	Time Base Phase HRPWM Register
TBPHS	0x6803	0x6843	0x6883	0x68C3	0x6903	0x6943	1 / 0	Time Base Phase Register
TBCTR	0x6804	0x6844	0x6884	0x68C4	0x6904	0x6944	1 / 0	Time Base Counter Register
TBPRD	0x6805	0x6845	0x6885	0x68C5	0x6905	0x6945	1 / 1	Time Base Period Register Set
CMPCTL	0x6807	0x6847	0x6887	0x68C7	0x6907	0x6947	1 / 0	Counter Compare Control Register
CMPAHR	0x6808	0x6848	0x6888	0x68C8	0x6908	0x6948	1 / 1	Time Base Compare A HRPWM Register
CMPA	0x6809	0x6849	0x6889	0x68C9	0x6909	0x6949	1 / 1	Counter Compare A Register Set
CMPB	0x680A	0x684A	0x688A	0x68CA	0x690A	0x694A	1 / 1	Counter Compare B Register Set
AQCTLA	0x680B	0x684B	0x688B	0x68CB	0x690B	0x694B	1 / 0	Action Qualifier Control Register For Output A
AQCTLB	0x680C	0x684C	0x688C	0x68CC	0x690C	0x694C	1 / 0	Action Qualifier Control Register For Output B
AQSFRC	0x680D	0x684D	0x688D	0x68CD	0x690D	0x694D	1 / 0	Action Qualifier Software Force Register
AQCSFRC	0x680E	0x684E	0x688E	0x68CE	0x690E	0x694E	1 / 1	Action Qualifier Continuous S/W Force Register Set
DBCTL	0x680F	0x684F	0x688F	0x68CF	0x690F	0x694F	1 / 1	Dead-Band Generator Control Register
DBRED	0x6810	0x6850	0x6890	0x68D0	0x6910	0x6950	1 / 0	Dead-Band Generator Rising Edge Delay Count Register
DBFED	0x6811	0x6851	0x6891	0x68D1	0x6911	0x6951	1 / 0	Dead-Band Generator Falling Edge Delay Count Register
TZSEL	0x6812	0x6852	0x6892	0x68D2	0x6912	0x6952	1 / 0	Trip Zone Select Register(1)
TZCTL	0x6814	0x6854	0x6894	0x68D4	0x6914	0x6954	1 / 0	Trip Zone Control Register(1)
TZEINT	0x6815	0x6855	0x6895	0x68D5	0x6915	0x6955	1 / 0	Trip Zone Enable Interrupt Register(1)
TZFLG	0x6816	0x6856	0x6896	0x68D6	0x6916	0x6956	1 / 0	Trip Zone Flag Register
TZCLR	0x6817	0x6857	0x6897	0x68D7	0x6917	0x6957	1 / 0	Trip Zone Clear Register(1)
TZFRC	0x6818	0x6858	0x6898	0x68D8	0x6918	0x6958	1 / 0	Trip Zone Force Register(1)
ETSEL	0x6819	0x6859	0x6899	0x68D9	0x6919	0x6959	1 / 0	Event Trigger Selection Register
ETPS	0x681A	0x685A	0x689A	0x68DA	0x691A	0x695A	1 / 0	Event Trigger Prescale Register
ETFLG	0x681B	0x685B	0x689B	0x68DB	0x691B	0x695B	1 / 0	Event Trigger Flag Register
ETCLR	0x681C	0x685C	0x689C	0x68DC	0x691C	0x695C	1 / 0	Event Trigger Clear Register
ETFRC	0x681D	0x685D	0x689D	0x68DD	0x691D	0x695D	1 / 0	Event Trigger Force Register
PCCTL	0x681E	0x685E	0x689E	0x68DE	0x691E	0x695E	1 / 0	PWM Chopper Control Register
HRCNFG	0x6820	0x6860	0x68A0	0x68E0	0x6920	0x6960	1 / 0	HRPWM Configuration Register(1)

(1) Registers that are EALLOW protected.

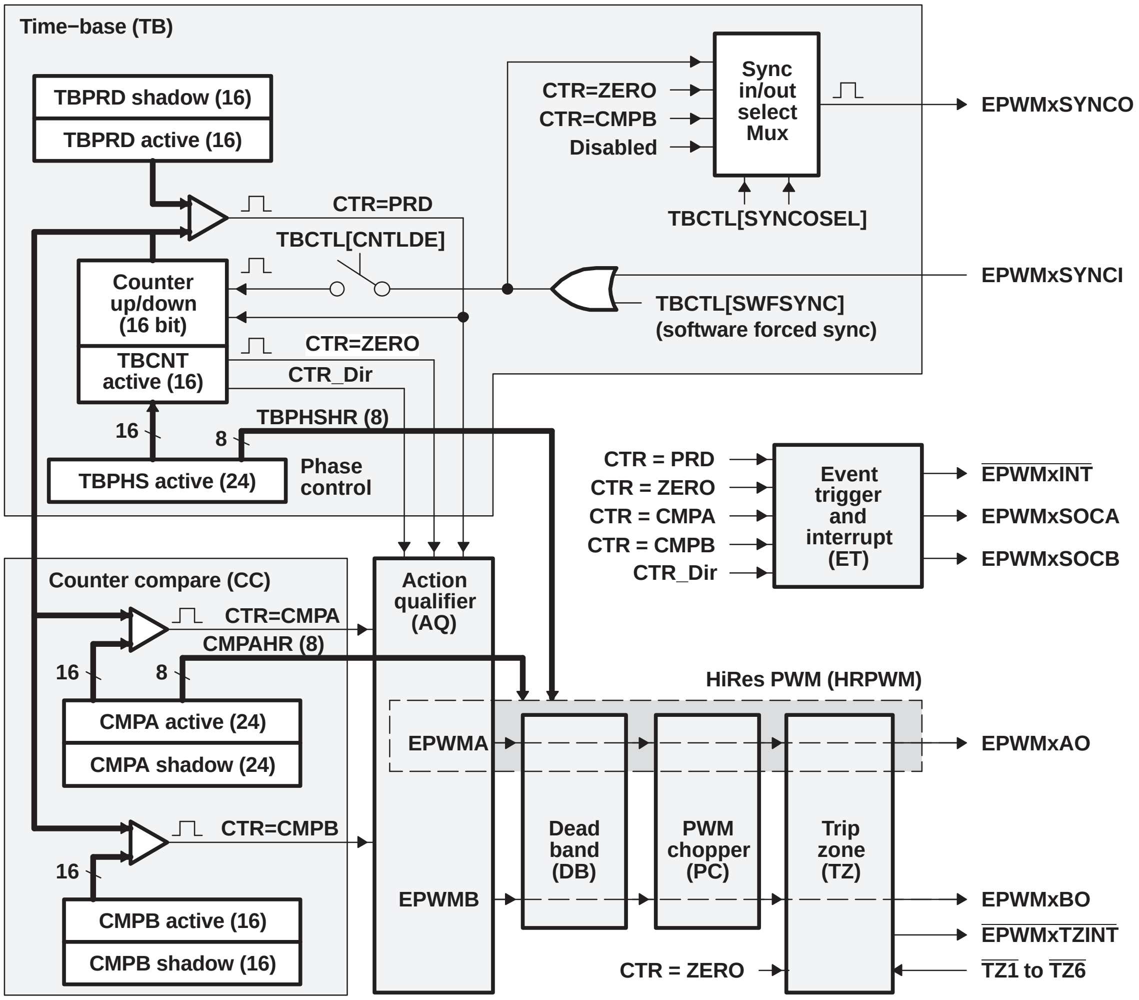


Figure 4-5. ePWM Sub-Modules Showing Critical Internal Signal Interconnections

4.4 High-Resolution PWM (HRPWM)

The HRPWM module offers PWM resolution (time granularity) which is significantly better than what can be achieved using conventionally derived digital PWM methods. The key points for the HRPWM module are:

- Significantly extends the time resolution capabilities of conventionally derived digital PWM
- Typically used when effective PWM resolution falls below ~ 9-10 bits. This occurs at PWM frequencies greater than ~200 KHz when using a CPU/System clock of 100 MHz.
- This capability can be utilized in both duty cycle and phase-shift control methods.
- Finer time granularity control or edge positioning is controlled via extensions to the Compare A and Phase registers of the ePWM module.
- HRPWM capabilities are offered only on the A signal path of an ePWM module (i.e., on the EPWMxA output). EPWMxB output has conventional PWM capabilities.

4.5 Enhanced CAP Modules (eCAP1/2/3/4/5/6)

The F2833x device contains up to six enhanced capture (eCAP) modules. Figure 4-6 shows a functional block diagram of a module. See the *TMS320x28xx, 28xxx Enhanced Capture (eCAP) Module Reference Guide* (literature number SPRU807) for more details.

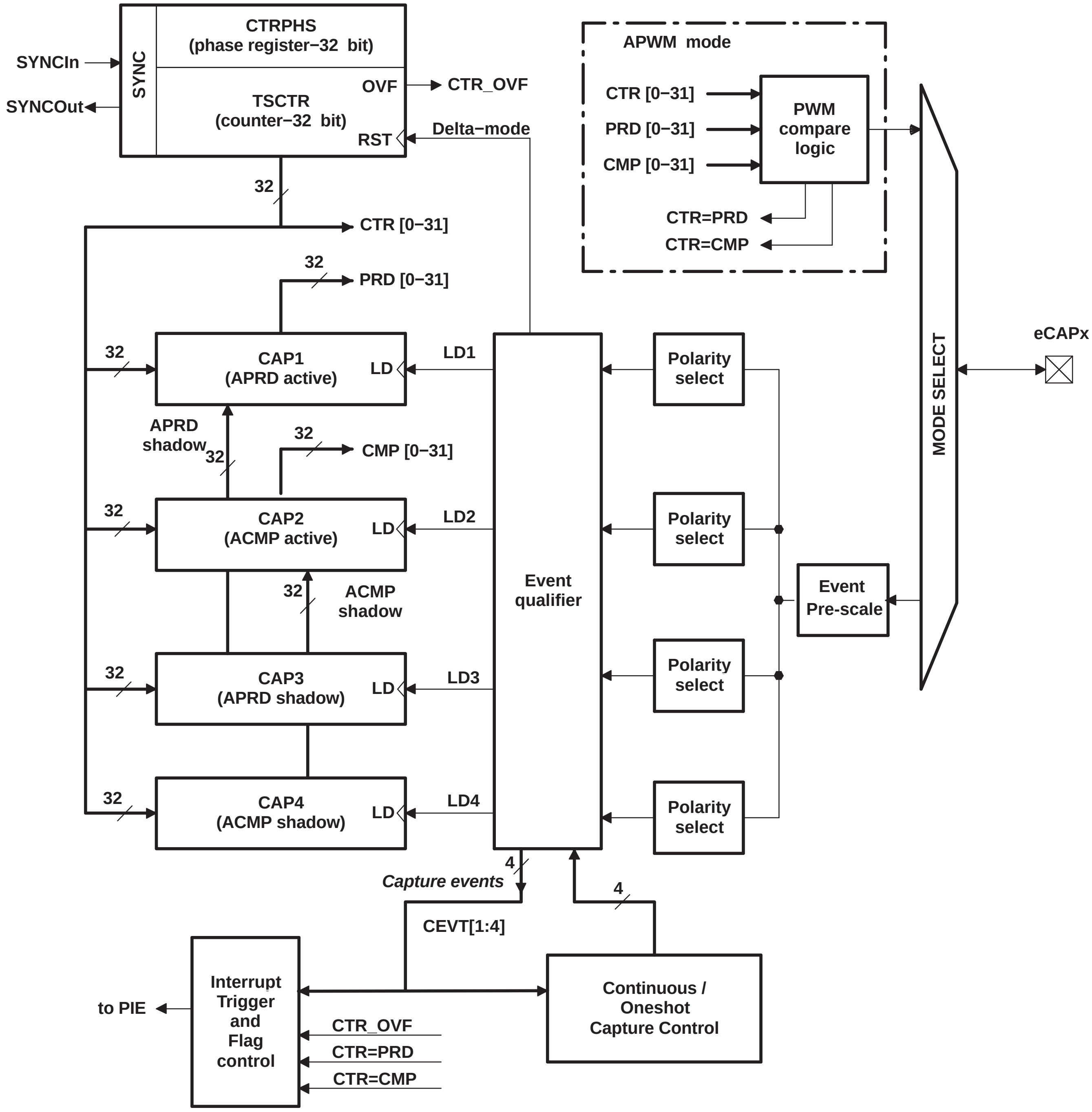


Figure 4-6. eCAP Functional Block Diagram

The eCAP modules are clocked at the SYSCLKOUT rate.

The clock enable bits (ECAP1/2/3/4/5/6ENCLK) in the PCLKCR1 register are used to turn off the eCAP modules individually (for low power operation). Upon reset, ECAP1ENCLK, ECAP2ENCLK, ECAP3ENCLK, ECAP4ENCLK, ECAP5ENCLK, and ECAP6ENCLK are set to low, indicating that the peripheral clock is off.

Table 4-3. eCAP Control and Status Registers

NAME	ECAP1	ECAP2	ECAP3	ECAP4	ECAP5	ECAP6	SIZE (x16)	DESCRIPTION
TSCTR	0x6A00	0x6A20	0x6A40	0x6A60	0x6A80	0x6AA0	2	Time-Stamp Counter
CTRPHS	0x6A02	0x6A22	0x6A42	0x6A62	0x6A82	0x6AA2	2	Counter Phase Offset Value Register
CAP1	0x6A04	0x6A24	0x6A44	0x6A64	0x6A84	0x6AA4	2	Capture 1 Register
CAP2	0x6A06	0x6A26	0x6A46	0x6A66	0x6A86	0x6AA6	2	Capture 2 Register
CAP3	0x6A08	0x6A28	0x6A48	0x6A68	0x6A88	0x6AA8	2	Capture 3 Register
CAP4	0x6A0A	0x6A2A	0x6A4A	0x6A6A	0x6A8A	0x6AAA	2	Capture 4 Register
Reserved	0x6A0C-0x6A12	0x6A2C-0x6A32	0x6A4C-0x6A52	0x6A6C-0x6A72	0x6A8C-0x6A92	0x6AAC-0x6AB2	8	Reserved
ECCTL1	0x6A14	0x6A34	0x6A54	0x6A74	0x6A94	0x6AB4	1	Capture Control Register 1
ECCTL2	0x6A15	0x6A35	0x6A55	0x6A75	0x6A95	0x6AB5	1	Capture Control Register 2
ECEINT	0x6A16	0x6A36	0x6A56	0x6A76	0x6A96	0x6AB6	1	Capture Interrupt Enable Register
ECFLG	0x6A17	0x6A37	0x6A57	0x6A77	0x6A97	0x6AB7	1	Capture Interrupt Flag Register
ECCLR	0x6A18	0x6A38	0x6A58	0x6A78	0x6A98	0x6AB8	1	Capture Interrupt Clear Register
ECFRC	0x6A19	0x6A39	0x6A59	0x6A79	0x6A99	0x6AB9	1	Capture Interrupt Force Register
Reserved	0x6A1A-0x6A1F	0x6A3A-0x6A3F	0x6A5A-0x6A5F	0x6A7A-0x6A7F	0x6A9A-0x6A9F	0x6ABA-0x6ABF	6	Reserved

ADVANCE INFORMATION

4.6 Enhanced QEP Modules (eQEP1/2)

The F2833x device contains up to two enhanced quadrature encoder (eQEP) modules. See the *TMS320x28xx, 28xxx Enhanced Quadrature Encoder (eQEP) Module Reference Guide* (literature number SPRU790) for more details.

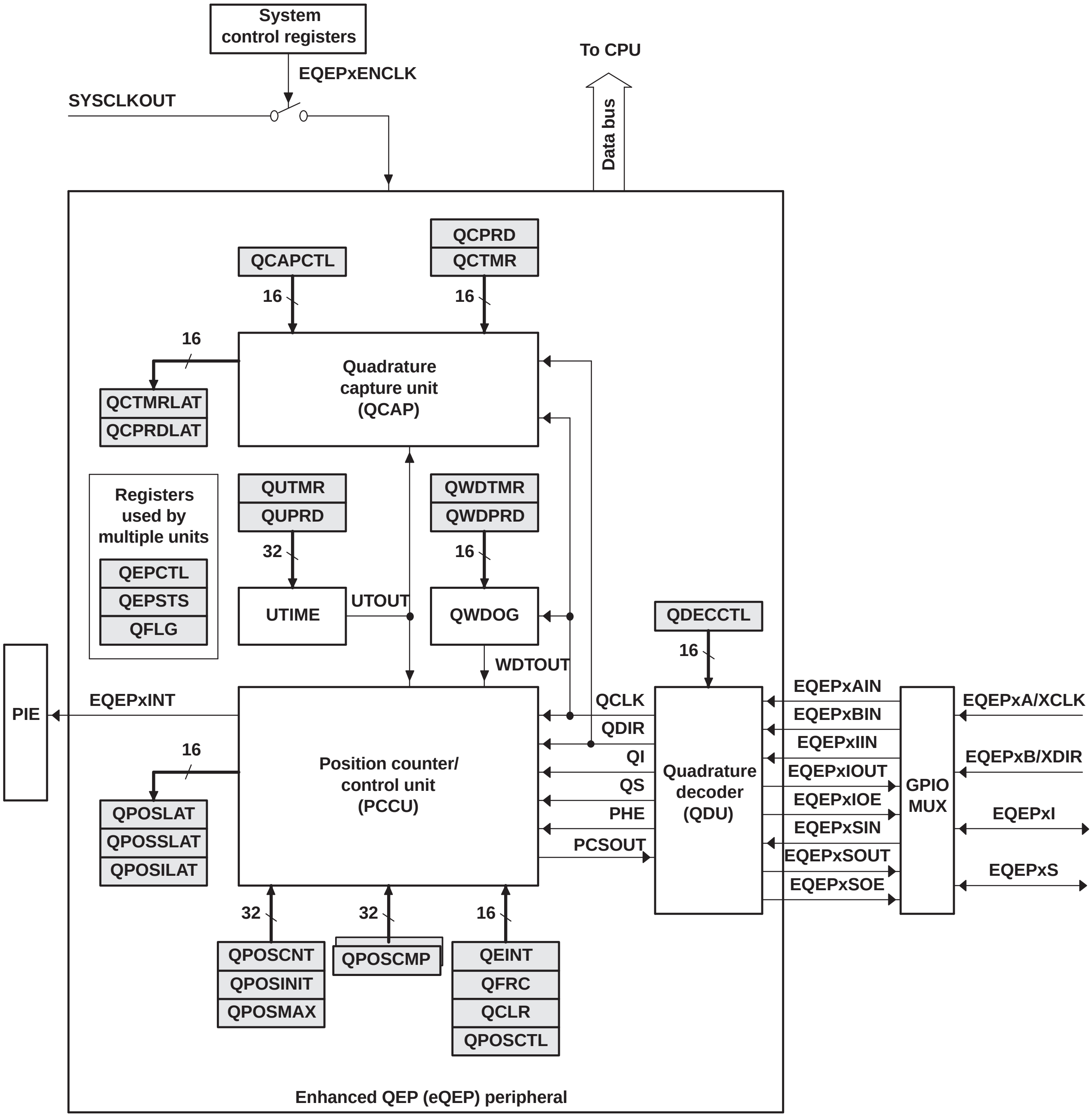


Figure 4-7. eQEP Functional Block Diagram

ADVANCE INFORMATION

Table 4-4. eQEP Control and Status Registers

NAME	EQEP1 ADDRESS	EQEP2 ADDRESS	EQEP1 SIZE(x16)/ #SHADOW	REGISTER DESCRIPTION
QPOSCNT	0x6B00	0x6B40	2/0	eQEP Position Counter
QPOSINIT	0x6B02	0x6B42	2/0	eQEP Initialization Position Count
QPOSMAX	0x6B04	0x6B44	2/0	eQEP Maximum Position Count
QPOSCMP	0x6B06	0x6B46	2/1	eQEP Position-compare
QPOSILAT	0x6B08	0x6B48	2/0	eQEP Index Position Latch
QPOSSLAT	0x6B0A	0x6B4A	2/0	eQEP Strobe Position Latch
QPOSLAT	0x6B0C	0x6B4C	2/0	eQEP Position Latch
QUTMR	0x6B0E	0x6B4E	2/0	eQEP Unit Timer
QUPRD	0x6B10	0x6B50	2/0	eQEP Unit Period Register
QWDTMR	0x6B12	0x6B52	1/0	eQEP Watchdog Timer
QWDPRD	0x6B13	0x6B53	1/0	eQEP Watchdog Period Register
QDECCTL	0x6B14	0x6B54	1/0	eQEP Decoder Control Register
QEPCTL	0x6B15	0x6B55	1/0	eQEP Control Register
QCAPCTL	0x6B16	0x6B56	1/0	eQEP Capture Control Register
QPOSCTL	0x6B17	0x6B57	1/0	eQEP Position-compare Control Register
QEINT	0x6B18	0x6B58	1/0	eQEP Interrupt Enable Register
QFLG	0x6B19	0x6B59	1/0	eQEP Interrupt Flag Register
QCLR	0x6B1A	0x6B5A	1/0	eQEP Interrupt Clear Register
QFRC	0x6B1B	0x6B5B	1/0	eQEP Interrupt Force Register
QEPSTS	0x6B1C	0x6B5C	1/0	eQEP Status Register
QCTMR	0x6B1D	0x6B5D	1/0	eQEP Capture Timer
QCPRD	0x6B1E	0x6B5E	1/0	eQEP Capture Period Register
QCTMRLAT	0x6B1F	0x6B5F	1/0	eQEP Capture Timer Latch
QCPRDLAT	0x6B20	0x6B60	1/0	eQEP Capture Period Latch
Reserved	0x6B21-0x6B3F	0x6B61-0x6B7F	31/0	

4.7 Analog-to-Digital Converter (ADC) Module

A simplified functional block diagram of the ADC module is shown in Figure 4-8. The ADC module consists of a 12-bit ADC with a built-in sample-and-hold (S/H) circuit. Functions of the ADC module include:

- 12-bit ADC core with built-in S/H
- Analog input: 0.0 V to 3.0 V (Voltages above 3.0 V produce full-scale conversion results.)
- Fast conversion rate: Up to 80 ns at 25-MHz ADC clock, 12.5 MSPS
- 16-channel, MUXed inputs
- Autosequencing capability provides up to 16 "autoconversions" in a single session. Each conversion can be programmed to select any 1 of 16 input channels
- Sequencer can be operated as two independent 8-state sequencers or as one large 16-state sequencer (i.e., two cascaded 8-state sequencers)
- Sixteen result registers (individually addressable) to store conversion values
  - The digital value of the input analog voltage is derived by:

$$\text{Digital Value} = 0, \qquad \text{when input} \le 0\ \text{V}$$

$$\text{Digital Value} = 4096 \times \frac{\text{Input Analog Voltage} - \text{ADCLO}}{3} \qquad \text{when } 0\ \text{V} < \text{input} < 3\ \text{V}$$

$$\text{Digital Value} = 4095, \qquad \text{when input} \ge 3\ \text{V}$$

A. All fractional values are truncated.

- Multiple triggers as sources for the start-of-conversion (SOC) sequence
  - S/W - software immediate start
  - ePWM start of conversion
  - XINT2 ADC start of conversion
- Flexible interrupt control allows interrupt request on every end-of-sequence (EOS) or every other EOS.
- Sequencer can operate in "start/stop" mode, allowing multiple "time-sequenced triggers" to synchronize conversions.
- SOCA and SOCB triggers can operate independently in dual-sequencer mode.
- Sample-and-hold (S/H) acquisition time window has separate prescale control.

The ADC module in the F2833x has been enhanced to provide flexible interface to ePWM peripherals. The ADC interface is built around a fast, 12-bit ADC module with a fast conversion rate of up to 80 ns at 25-MHz ADC clock. The ADC module has 16 channels, configurable as two independent 8-channel modules. The two independent 8-channel modules can be cascaded to form a 16-channel module. Although there are multiple input channels and two sequencers, there is only one converter in the ADC module. Figure 4-8 shows the block diagram of the ADC module.

The two 8-channel modules have the capability to autosequence a series of conversions, each module has the choice of selecting any one of the respective eight channels available through an analog MUX. In the cascaded mode, the autosequencer functions as a single 16-channel sequencer. On each sequencer, once the conversion is complete, the selected channel value is stored in its respective RESULT register. Autosequencing allows the system to convert the same channel multiple times, allowing the user to perform oversampling algorithms. This gives increased resolution over traditional single-sampled conversion results.

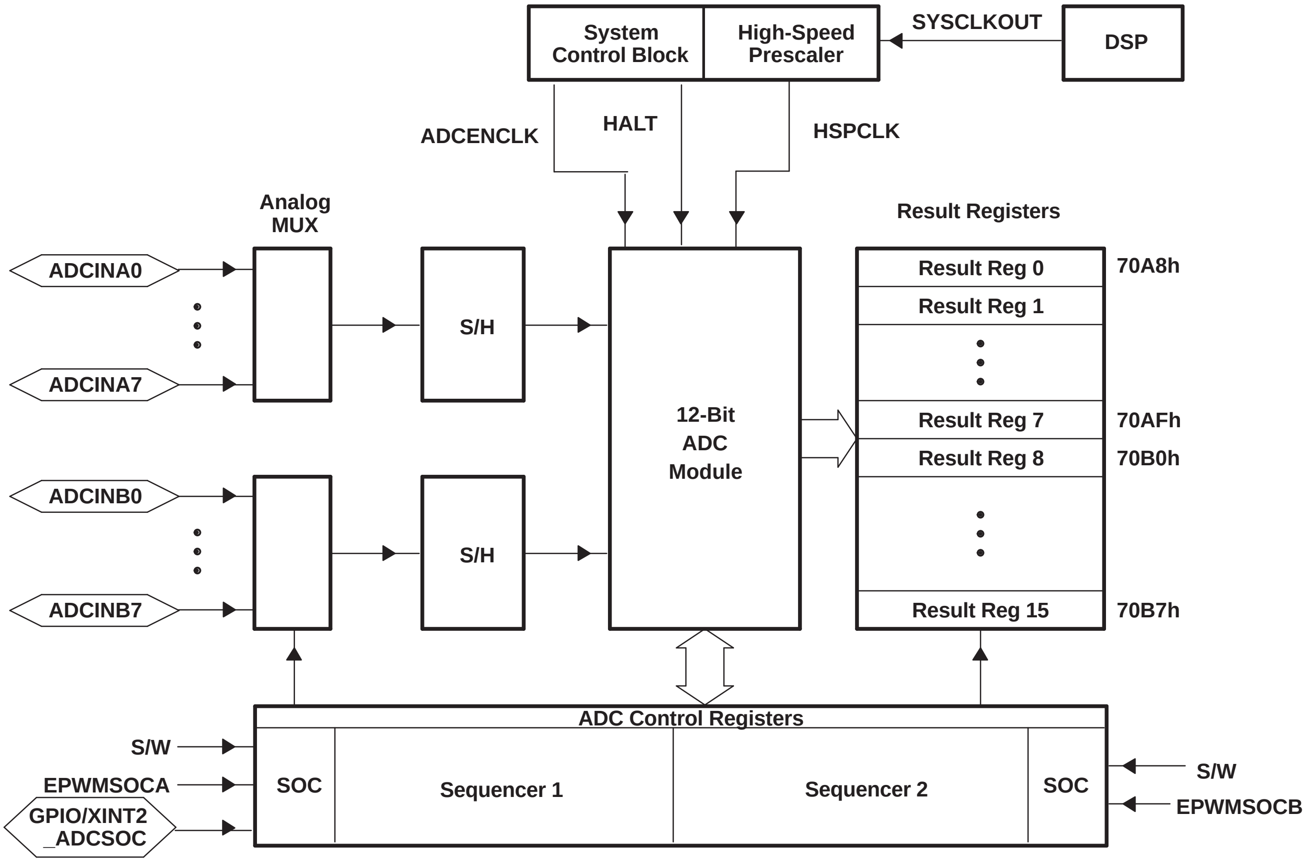


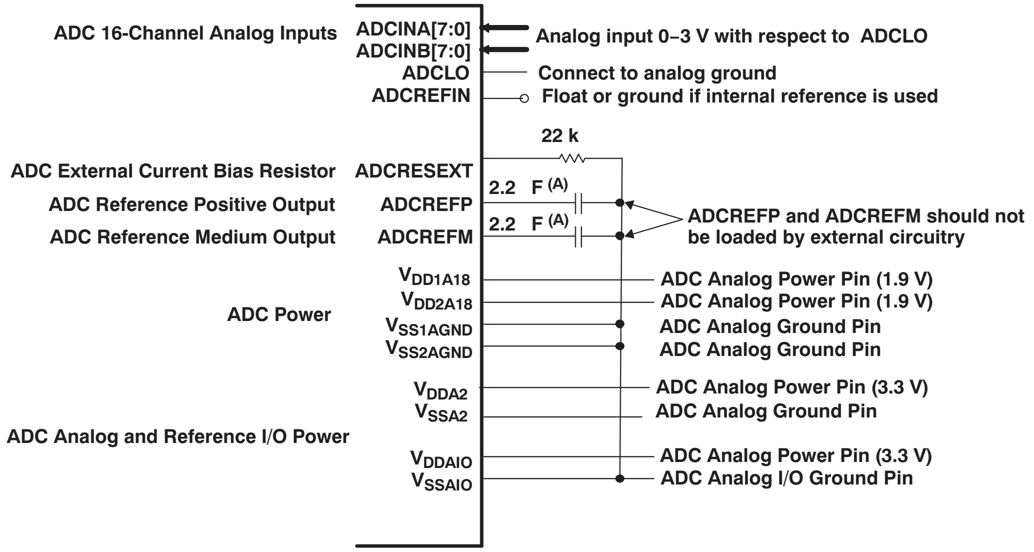
Figure 4-8. Block Diagram of the ADC Module

To obtain the specified accuracy of the ADC, proper board layout is very critical. To the best extent possible, traces leading to the ADCIN pins should not run in close proximity to the digital signal paths. This is to minimize switching noise on the digital lines from getting coupled to the ADC inputs. Furthermore, proper isolation techniques must be used to isolate the ADC module power pins (V_{DD1A18}, V_{DD2A18} , V_{DDA2}, V_{DDAIO}) from the digital supply. Figure 4-9 shows the ADC pin connections for the F2833x devices.

NOTE

1. The ADC registers are accessed at the SYSCLKOUT rate. The internal timing of the ADC module is controlled by the high-speed peripheral clock (HSPCLK).
2. The behavior of the ADC module based on the state of the ADCENCLK and HALT signals is as follows:
   - *ADCENCLK:* On reset, this signal will be low. While reset is active-low ($\overline{XRS}$) the clock to the register will still function. This is necessary to make sure all registers and modes go into their default reset state. The analog module, however, will be in a low-power inactive state. As soon as reset goes high, then the clock to the registers will be disabled. When the user sets the ADCENCLK signal high, then the clocks to the registers will be enabled and the analog module will be enabled. There will be a certain time delay (ms range) before the ADC is stable and can be used.
   - *HALT:* This mode only affects the analog module. It does not affect the registers. In this mode, the ADC module goes into low-power mode. This mode also will stop the clock to the CPU, which will stop the HSPCLK; therefore, the ADC register logic will be turned off indirectly.

Figure 4-9 shows the ADC pin-biasing for internal reference and Figure 4-10 shows the ADC pin-biasing for external reference.

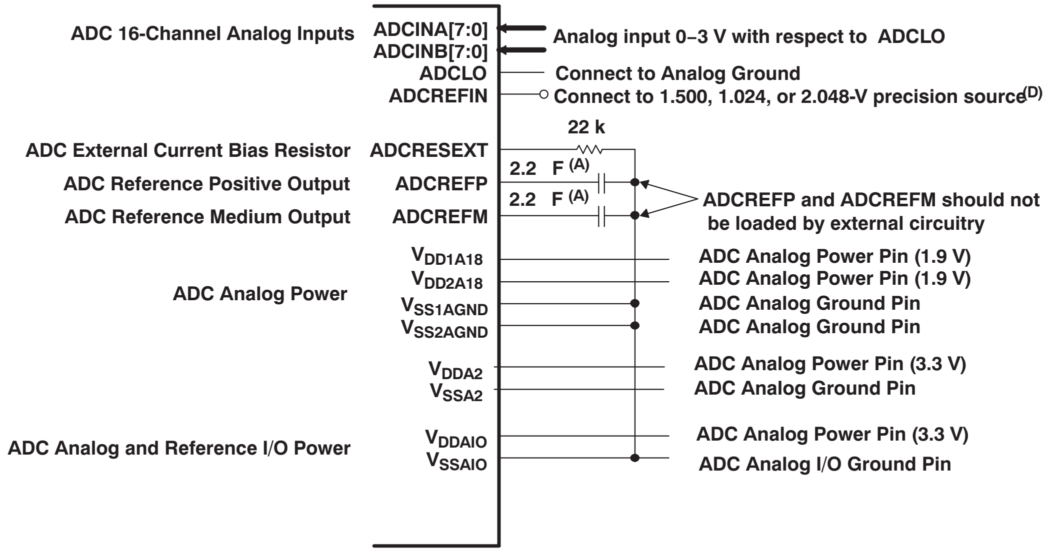


A. TAIYO YUDEN LMK212BJ225MG-T or equivalent

B. External decoupling capacitors are recommended on all power pins.

C. Analog inputs must be driven from an operational amplifier that does not degrade the ADC performance.

Figure 4-9. ADC Pin Connections With Internal Reference



A. TAIYO YUDEN LMK212BJ225MG-T or equivalent

B. External decoupling capacitors are recommended on all power pins.

C. Analog inputs must be driven from an operational amplifier that does not degrade the ADC performance.

D. External voltage on ADCREFIN is enabled by changing bits 15:14 in the ADC Reference Select register depending on the voltage used on this pin. TI recommends TI part REF3020 or equivalent for 2.048-V generation. Overall gain accuracy will be determined by accuracy of this voltage source.

Figure 4-10. ADC Pin Connections With External Reference

NOTE

The temperature rating of any recommended component must match the rating of the end product.

4.7.1 ADC Connections if the ADC Is Not Used

It is recommended to keep the connections for the analog power pins, even if the ADC is not used. Following is a summary of how the ADC pins should be connected, if the ADC is not used in an application:

- V_{DD1A18}/V_{DD2A18} – Connect to V_{DD}
- V_{DDA2}, V_{DDAIO} – Connect to V_{DDIO}
- $V_{SS1AGND}$/$V_{SS2AGND}$, V_{SSA2}, V_{SSAIO} – Connect to V_{SS}
- ADCLO – Connect to V_{SS}
- ADCREFIN – Connect to V_{SS}
- ADCREFP/ADCREFM – Connect a 100-nF cap to V_{SS}
- ADCRESEXT – Connect a 20-kΩ resistor (very loose tolerance) to V_{SS}.
- ADCINAn, ADCINBn - Connect to V_{SS}

When the ADC is not used, be sure that the clock to the ADC module is not turned on to realize power savings.

When the ADC module is used in an application, unused ADC input pins should be connected to analog ground ($V_{SS1AGND}$/$V_{SS2AGND}$)

NOTE

ADC parameters for gain error and offset error are specified only if the ADC calibration routine is executed from the Boot ROM. See Section 4.7.3 for more information.

4.7.2 ADC Registers

The ADC operation is configured, controlled, and monitored by the registers listed in Table 4-5.

Table 4-5. ADC Registers[1]

NAME	ADDRESS[1]	ADDRESS[2]	SIZE (x16)	DESCRIPTION
ADCTRL1	0x7100		1	ADC Control Register 1
ADCTRL2	0x7101		1	ADC Control Register 2
ADCMAXCONV	0x7102		1	ADC Maximum Conversion Channels Register
ADCCHSELSEQ1	0x7103		1	ADC Channel Select Sequencing Control Register 1
ADCCHSELSEQ2	0x7104		1	ADC Channel Select Sequencing Control Register 2
ADCCHSELSEQ3	0x7105		1	ADC Channel Select Sequencing Control Register 3
ADCCHSELSEQ4	0x7106		1	ADC Channel Select Sequencing Control Register 4
ADCASEQSR	0x7107		1	ADC Auto-Sequence Status Register
ADCRESULT0	0x7108	0x0B00	1	ADC Conversion Result Buffer Register 0
ADCRESULT1	0x7109	0x0B01	1	ADC Conversion Result Buffer Register 1
ADCRESULT2	0x710A	0x0B02	1	ADC Conversion Result Buffer Register 2
ADCRESULT3	0x710B	0x0B03	1	ADC Conversion Result Buffer Register 3
ADCRESULT4	0x710C	0x0B04	1	ADC Conversion Result Buffer Register 4
ADCRESULT5	0x710D	0x0B05	1	ADC Conversion Result Buffer Register 5

(1) The registers in this column are Peripheral Frame 2 Registers.

(2) The ADC result registers are dual mapped in the F2833x DSC. Locations in Peripheral Frame 2 (0x7108-0x7117) are 2 wait-states and left justified. Locations in Peripheral frame 0 space (0x0B00-0x0B0F) are 1 wait-state for CPU accesses and 0 wait state for DMA accesses and right justified. During high speed/continuous conversion use of the ADC, use the 0 wait-state locations for fast transfer of ADC results to user memory.

Table 4-5. ADC Registers (continued)

NAME	ADDRESS[1]	ADDRESS[2]	SIZE (x16)	DESCRIPTION
ADCRESULT6	0x710E	0x0B06	1	ADC Conversion Result Buffer Register 6
ADCRESULT7	0x710F	0x0B07	1	ADC Conversion Result Buffer Register 7
ADCRESULT8	0x7110	0x0B08	1	ADC Conversion Result Buffer Register 8
ADCRESULT9	0x7111	0x0B09	1	ADC Conversion Result Buffer Register 9
ADCRESULT10	0x7112	0x0B0A	1	ADC Conversion Result Buffer Register 10
ADCRESULT11	0x7113	0x0B0B	1	ADC Conversion Result Buffer Register 11
ADCRESULT12	0x7114	0x0B0C	1	ADC Conversion Result Buffer Register 12
ADCRESULT13	0x7115	0x0B0D	1	ADC Conversion Result Buffer Register 13
ADCRESULT14	0x7116	0x0B0E	1	ADC Conversion Result Buffer Register 14
ADCRESULT15	0x7117	0x0B0F	1	ADC Conversion Result Buffer Register 15
ADCTRL3	0x7118		1	ADC Control Register 3
ADCST	0x7119		1	ADC Status Register
Reserved	0x711A 0x711B		2	
ADCREFSEL	0x711C		1	ADC Reference Select Register
ADCOFFTRIM	0x711D		1	ADC Offset Trim Register
Reserved	0x711E 0x711F		2	

4.7.3 ADC Calibration

The ADC_cal() routine is programmed into TI reserved OTP memory by the factory. The boot ROM automatically calls the ADC_cal() routine to initialize the ADCREFSEL and ADCOFFTRIM registers with device specific calibration data. During normal operation, this process occurs automatically and no action is required by the user.

If the boot ROM is bypassed by Code Composer Studio during the development process, then ADCREFSEL and ADCOFFTRIM must be initialized by the application. For working examples, see the ADC initialization in the *C2833x C/C++ Header Files and Peripheral Examples* (SPRC530). Methods for calling the ADC_cal() routine from an application are described in *TMS3202833x Analog-to-Digital Converter (ADC) Module Reference Guide* (SPRU812).

NOTE

FAILURE TO INITIALIZE THESE REGISTERS WILL CAUSE THE ADC TO FUNCTION OUT OF SPECIFICATION.

Because TI reserved OTP memory is secure, the ADC_Cal() routine must be called from secure memory or called from non-secure memory after the Code Security Module is unlocked. If the system is reset or the ADC module is reset using Bit 14 (RESET) from the ADC Control Register 1, the routine must be repeated.

ADVANCE INFORMATION

4.8 Multichannel Buffered Serial Port (McBSP) Module

The McBSP module has the following features:

- Compatible to McBSP in TMS320C54x™/TMS320C55x™ DSC devices
- Full–duplex communication
- Double–buffered data registers that allow a continuous data stream
- Independent framing and clocking for receive and transmit
- External shift clock generation or an internal programmable frequency shift clock
- A wide selection of data sizes including 8–, 12–, 16–, 20–, 24–, or 32–bits
- 8–bit data transfers with LSB or MSB first
- Programmable polarity for both frame synchronization and data clocks
- Highly programmable internal clock and frame generation
- Direct interface to industry–standard CODECs, Analog Interface Chips (AICs), and other serially connected A/D and D/A devices
- Works with SPI–compatible devices

The following application interfaces can be supported on the McBSP:

- T1/E1 framers
- MVIP switching–compatible and ST–BUS–compliant devices including:
  - MVIP framers
  - H.100 framers
  - SCSA framers
  - IOM–2 compliant devices
  - AC97–compliant devices (the necessary multiphase frame synchronization capability is provided.)
  - IIS–compliant devices
- McBSP clock rate,

$$\text{CLKG} = \frac{\text{CLKSRG}}{(1 + \text{CLKGDV})}$$

where CLKSRG source could be LSPCLK, CLKX, or CLKR. Serial port performance is limited by I/O buffer switching speed. Internal prescalers must be adjusted such that the peripheral speed is less than the I/O buffer speed limit—20–MHz maximum.

Figure 4-11 shows the block diagram of the McBSP module.

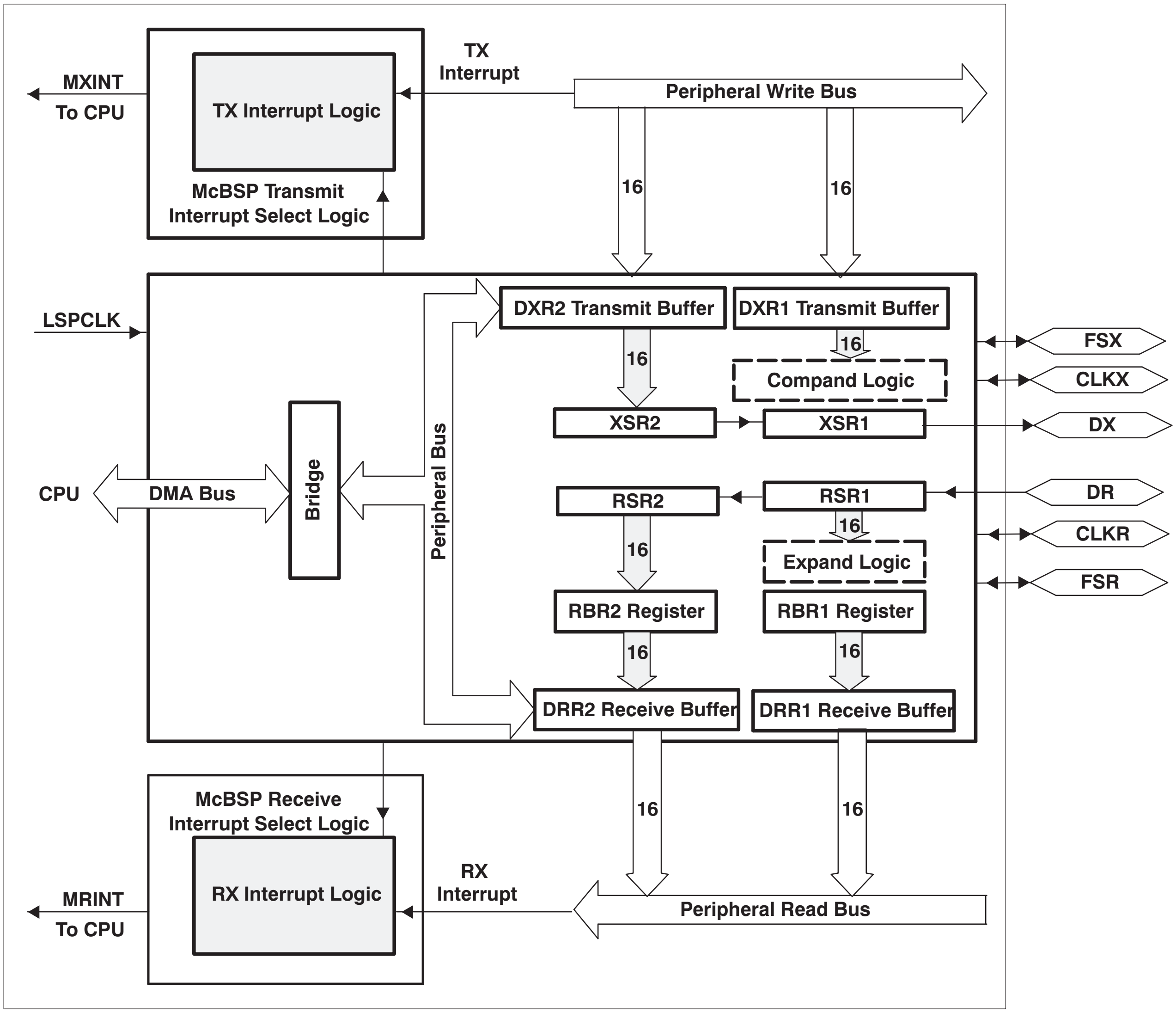


Figure 4-11. McBSP Module

Table 4-6 provides a summary of the McBSP registers.

ADVANCE INFORMATION

Table 4-6. McBSP Register Summary

NAME	McBSP-A ADDRESS	McBSP-B ADDRESS	TYPE	RESET VALUE	DESCRIPTION
DATA REGISTERS, RECEIVE, TRANSMIT					
DRR2	0x5000	0x5040	R	0x0000	McBSP Data Receive Register 2
DRR1	0x5001	0x5041	R	0x0000	McBSP Data Receive Register 1
DXR2	0x5002	0x5042	W	0x0000	McBSP Data Transmit Register 2
DXR1	0x5003	0x5043	W	0x0000	McBSP Data Transmit Register 1
McBSP CONTROL REGISTERS					
SPCR2	0x5004	0x5044	R/W	0x0000	McBSP Serial Port Control Register 2
SPCR1	0x5005	0x5045	R/W	0x0000	McBSP Serial Port Control Register 1
RCR2	0x5006	0x5046	R/W	0x0000	McBSP Receive Control Register 2
RCR1	0x5007	0x5047	R/W	0x0000	McBSP Receive Control Register 1
XCR2	0x5008	0x5048	R/W	0x0000	McBSP Transmit Control Register 2
XCR1	0x5009	0x5049	R/W	0x0000	McBSP Transmit Control Register 1
SRGR2	0x500A	0x504A	R/W	0x0000	McBSP Sample Rate Generator Register 2
SRGR1	0x500B	0x504B	R/W	0x0000	McBSP Sample Rate Generator Register 1
MULTICHANNEL CONTROL REGISTERS					
MCR2	0x500C	0x504C	R/W	0x0000	McBSP Multichannel Register 2
MCR1	0x500D	0x504D	R/W	0x0000	McBSP Multichannel Register 1
RCERA	0x500E	0x504E	R/W	0x0000	McBSP Receive Channel Enable Register Partition A
RCERB	0x500F	0x504F	R/W	0x0000	McBSP Receive Channel Enable Register Partition B
XCERA	0x5010	0x5050	R/W	0x0000	McBSP Transmit Channel Enable Register Partition A
XCERB	0x5011	0x5051	R/W	0x0000	McBSP Transmit Channel Enable Register Partition B
PCR	0x5012	0x5052	R/W	0x0000	McBSP Pin Control Register
RCERC	0x5013	0x5053	R/W	0x0000	McBSP Receive Channel Enable Register Partition C
RCERD	0x5014	0x5054	R/W	0x0000	McBSP Receive Channel Enable Register Partition D
XCERC	0x5015	0x5055	R/W	0x0000	McBSP Transmit Channel Enable Register Partition C
XCERD	0x5016	0x5056	R/W	0x0000	McBSP Transmit Channel Enable Register Partition D
RCERE	0x5017	0x5057	R/W	0x0000	McBSP Receive Channel Enable Register Partition E
RCERF	0x5018	0x5058	R/W	0x0000	McBSP Receive Channel Enable Register Partition F
XCERE	0x5019	0x5059	R/W	0x0000	McBSP Transmit Channel Enable Register Partition E
XCERF	0x501A	0x505A	R/W	0x0000	McBSP Transmit Channel Enable Register Partition F
RCERG	0x501B	0x505B	R/W	0x0000	McBSP Receive Channel Enable Register Partition G
RCERH	0x501C	0x505C	R/W	0x0000	McBSP Receive Channel Enable Register Partition H
XCERG	0x501D	0x505D	R/W	0x0000	McBSP Transmit Channel Enable Register Partition G
XCERH	0x501E	0x505E	R/W	0x0000	McBSP Transmit Channel Enable Register Partition H
MFFINT	0x5023	0x5063	R/W	0x0000	McBSP Interrupt Enable Register
MFFST	0x5024	0x5064	R/W	0x0000	McBSP Pin Status Register

4.9 Enhanced Controller Area Network (eCAN) Modules (eCAN-A and eCAN-B)

The CAN module has the following features:

- Fully compliant with CAN protocol, version 2.0B
- Supports data rates up to 1 Mbps
- Thirty-two mailboxes, each with the following properties:
  - Configurable as receive or transmit
  - Configurable with standard or extended identifier
  - Has a programmable receive mask
  - Supports data and remote frame
  - Composed of 0 to 8 bytes of data
  - Uses a 32-bit time stamp on receive and transmit message
  - Protects against reception of new message
  - Holds the dynamically programmable priority of transmit message
  - Employs a programmable interrupt scheme with two interrupt levels
  - Employs a programmable alarm on transmission or reception time-out
- Low-power mode
- Programmable wake-up on bus activity
- Automatic reply to a remote request message
- Automatic retransmission of a frame in case of loss of arbitration or error
- 32-bit local network time counter synchronized by a specific message (communication in conjunction with mailbox 16)
- Self-test mode
  - Operates in a loopback mode receiving its own message. A "dummy" acknowledge is provided, thereby eliminating the need for another node to provide the acknowledge bit.

NOTE

For a SYSCLKOUT of 100 MHz, the smallest bit rate possible is 15.625 kbps.

For a SYSCLKOUT of 150 MHz, the smallest bit rate possible is 23.4 kbps.

The F2833x CAN has passed the conformance test per ISO/DIS 16845. Contact TI for details.

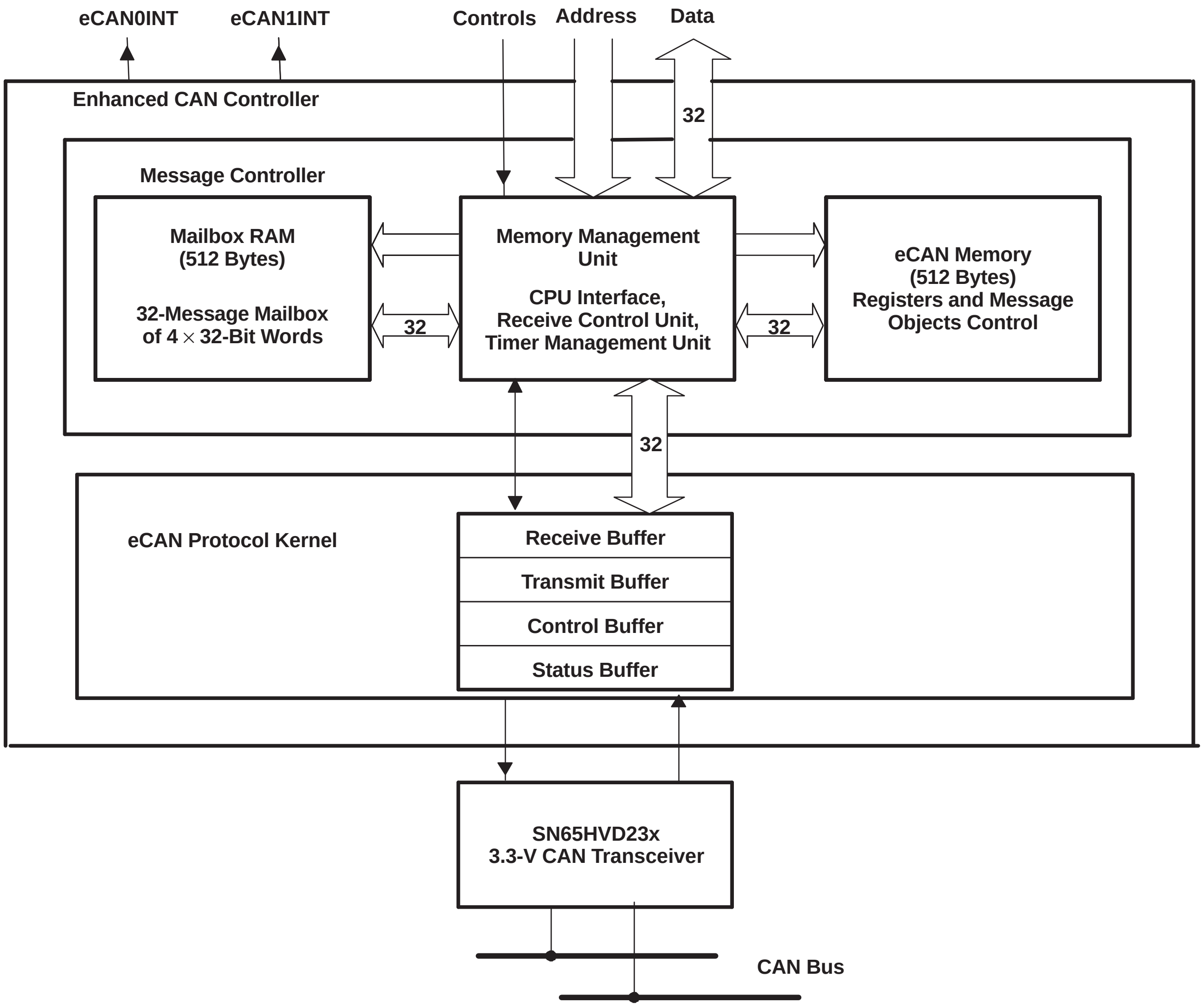


Figure 4-12. eCAN Block Diagram and Interface Circuit

Table 4-7. 3.3-V eCAN Transceivers

PART NUMBER	SUPPLY VOLTAGE	LOW-POWER MODE	SLOPE CONTROL	VREF	OTHER	T_A
SN65HVD230	3.3 V	Standby	Adjustable	Yes	–	-40°C to 85°C
SN65HVD230Q	3.3 V	Standby	Adjustable	Yes	–	-40°C to 125°C
SN65HVD231	3.3 V	Sleep	Adjustable	Yes	–	-40°C to 85°C
SN65HVD231Q	3.3 V	Sleep	Adjustable	Yes	–	-40°C to 125°C
SN65HVD232	3.3 V	None	None	None	–	-40°C to 85°C
SN65HVD232Q	3.3 V	None	None	None	–	-40°C to 125°C
SN65HVD233	3.3 V	Standby	Adjustable	None	Diagnostic Loopback	-40°C to 125°C
SN65HVD234	3.3 V	Standby and Sleep	Adjustable	None	–	-40°C to 125°C
SN65HVD235	3.3 V	Standby	Adjustable	None	Autobaud Loopback	-40°C to 125°C

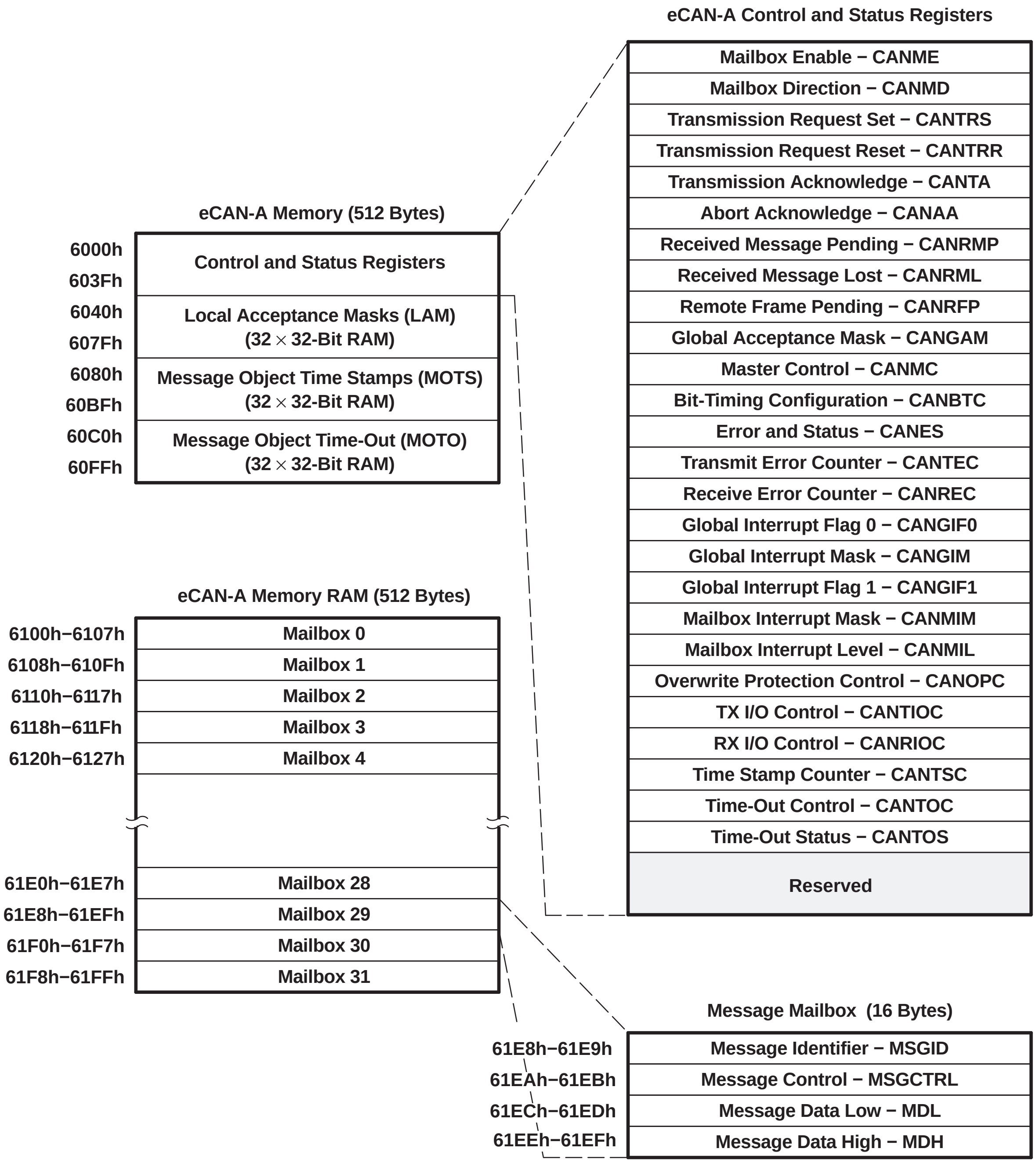


Figure 4-13. eCAN-A Memory Map

NOTE

If the eCAN module is not used in an application, the RAM available (LAM, MOTS, MOTO, and mailbox RAM) can be used as general-purpose RAM. The CAN module clock should be enabled for this.

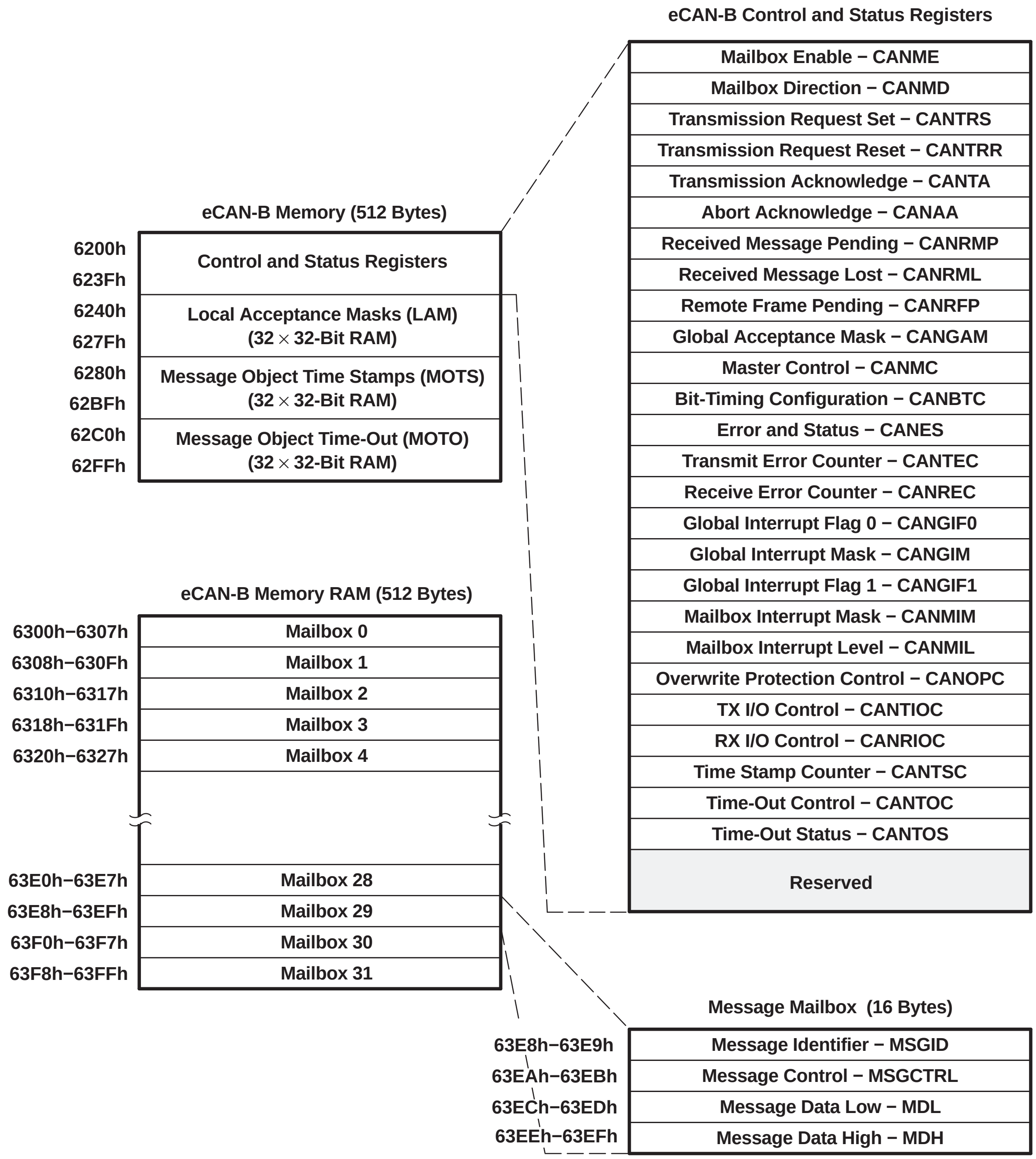


Figure 4-14. eCAN-B Memory Map

The CAN registers listed in Table 4-8 are used by the CPU to configure and control the CAN controller and the message objects. eCAN control registers only support 32-bit read/write operations. Mailbox RAM can be accessed as 16 bits or 32 bits. 32-bit accesses are aligned to an even boundary.

Table 4-8. CAN Register Map[1]

REGISTER NAME	ECAN-A ADDRESS	ECAN-B ADDRESS	SIZE (x32)	DESCRIPTION
CANME	0x6000	0x6200	1	Mailbox enable
CANMD	0x6002	0x6202	1	Mailbox direction
CANTRS	0x6004	0x6204	1	Transmit request set
CANTRR	0x6006	0x6206	1	Transmit request reset
CANTA	0x6008	0x6208	1	Transmission acknowledge
CANAA	0x600A	0x620A	1	Abort acknowledge
CANRMP	0x600C	0x620C	1	Receive message pending
CANRML	0x600E	0x620E	1	Receive message lost
CANRFP	0x6010	0x6210	1	Remote frame pending
CANGAM	0x6012	0x6212	1	Global acceptance mask
CANMC	0x6014	0x6214	1	Master control
CANBTC	0x6016	0x6216	1	Bit-timing configuration
CANES	0x6018	0x6218	1	Error and status
CANTEC	0x601A	0x621A	1	Transmit error counter
CANREC	0x601C	0x621C	1	Receive error counter
CANGIF0	0x601E	0x621E	1	Global interrupt flag 0
CANGIM	0x6020	0x6220	1	Global interrupt mask
CANGIF1	0x6022	0x6222	1	Global interrupt flag 1
CANMIM	0x6024	0x6224	1	Mailbox interrupt mask
CANMIL	0x6026	0x6226	1	Mailbox interrupt level
CANOPC	0x6028	0x6228	1	Overwrite protection control
CANTIOC	0x602A	0x622A	1	TX I/O control
CANRIOC	0x602C	0x622C	1	RX I/O control
CANTSC	0x602E	0x622E	1	Time stamp counter (Reserved in SCC mode)
CANTOC	0x6030	0x6230	1	Time-out control (Reserved in SCC mode)
CANTOS	0x6032	0x6232	1	Time-out status (Reserved in SCC mode)

(1) These registers are mapped to Peripheral Frame 1.

ADVANCE INFORMATION

4.10 Serial Communications Interface (SCI) Modules (SCI-A, SCI-B, SCI-C)

The F2833x devices include three serial communications interface (SCI) modules. The SCI modules support digital communications between the CPU and other asynchronous peripherals that use the standard non-return-to-zero (NRZ) format. The SCI receiver and transmitter are double-buffered, and each has its own separate enable and interrupt bits. Both can be operated independently or simultaneously in the full-duplex mode. To ensure data integrity, the SCI checks received data for break detection, parity, overrun, and framing errors. The bit rate is programmable to over 65000 different speeds through a 16-bit baud-select register.

Features of each SCI module include:

- Two external pins:
  - SCITXD: SCI transmit-output pin
  - SCIRXD: SCI receive-input pin
    NOTE: Both pins can be used as GPIO if not used for SCI.
  - Baud rate programmable to 64K different rates:

$$\text{Baud rate} = \frac{\text{LSPCLK}}{(\text{BRR} + 1) * 8} \quad \text{when BRR} \neq 0$$

$$\text{Baud rate} = \frac{\text{LSPCLK}}{16} \quad \text{when BRR} = 0$$

- 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 flag (monitoring four interrupt conditions)
- Separate enable bits for transmitter and receiver interrupts (except BRKDT)
- $\text{Max bit rate} = \frac{150\text{ MHz}}{16} = 9.375 \times 10^6 \text{ b/s}$ (for 150-MHz devices)
- $\text{Max bit rate} = \frac{100\text{ MHz}}{16} = 6.25 \times 10^6 \text{ b/s}$ (for 100-MHz devices)
- NRZ (non-return-to-zero) format
- Ten SCI module control registers located in the control register frame beginning at address 7050h

---

NOTE

All registers in this module are 8-bit registers that are connected to Peripheral Frame 2. 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.

Enhanced features:

- Auto baud-detect hardware logic

- 16-level transmit/receive FIFO

The SCI port operation is configured and controlled by the registers listed in Table 4-9, Table 4-10, and Table 4-11.

Table 4-9. SCI-A Registers(1)

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
SCICCRA	0x7050	1	SCI-A Communications Control Register
SCICTL1A	0x7051	1	SCI-A Control Register 1
SCIHBAUDA	0x7052	1	SCI-A Baud Register, High Bits
SCILBAUDA	0x7053	1	SCI-A Baud Register, Low Bits
SCICTL2A	0x7054	1	SCI-A Control Register 2
SCIRXSTA	0x7055	1	SCI-A Receive Status Register
SCIRXEMUA	0x7056	1	SCI-A Receive Emulation Data Buffer Register
SCIRXBUFA	0x7057	1	SCI-A Receive Data Buffer Register
SCITXBUFA	0x7059	1	SCI-A Transmit Data Buffer Register
SCIFFTXA(2)	0x705A	1	SCI-A FIFO Transmit Register
SCIFFRXA(2)	0x705B	1	SCI-A FIFO Receive Register
SCIFFCTA(2)	0x705C	1	SCI-A FIFO Control Register
SCIPRIA	0x705F	1	SCI-A Priority Control Register

(1) Registers in this table are mapped to Peripheral Frame 2 space. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.
(2) These registers are new registers for the FIFO mode.

Table 4-10. SCI-B Registers(1) (2)

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
SCICCRB	0x7750	1	SCI-B Communications Control Register
SCICTL1B	0x7751	1	SCI-B Control Register 1
SCIHBAUDB	0x7752	1	SCI-B Baud Register, High Bits
SCILBAUDB	0x7753	1	SCI-B Baud Register, Low Bits
SCICTL2B	0x7754	1	SCI-B Control Register 2
SCIRXSTB	0x7755	1	SCI-B Receive Status Register
SCIRXEMUB	0x7756	1	SCI-B Receive Emulation Data Buffer Register
SCIRXBUFB	0x7757	1	SCI-B Receive Data Buffer Register
SCITXBUFB	0x7759	1	SCI-B Transmit Data Buffer Register
SCIFFTXB(2)	0x775A	1	SCI-B FIFO Transmit Register
SCIFFRXB(2)	0x775B	1	SCI-B FIFO Receive Register
SCIFFCTB(2)	0x775C	1	SCI-B FIFO Control Register
SCIPRIB	0x775F	1	SCI-B Priority Control Register

(1) Registers in this table are mapped to Peripheral Frame 2 space. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.
(2) These registers are new registers for the FIFO mode.

Table 4-11. SCI-C Registers(1) (2)

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
SCICCRC	0x7770	1	SCI-C Communications Control Register
SCICTL1C	0x7771	1	SCI-C Control Register 1
SCIHBAUDC	0x7772	1	SCI-C Baud Register, High Bits
SCILBAUDC	0x7773	1	SCI-C Baud Register, Low Bits
SCICTL2C	0x7774	1	SCI-C Control Register 2
SCIRXSTC	0x7775	1	SCI-C Receive Status Register
SCIRXEMUC	0x7776	1	SCI-C Receive Emulation Data Buffer Register
SCIRXBUFC	0x7777	1	SCI-C Receive Data Buffer Register
SCITXBUFC	0x7779	1	SCI-C Transmit Data Buffer Register
SCIFFTXC(2)	0x777A	1	SCI-C FIFO Transmit Register
SCIFFRXC(2)	0x777B	1	SCI-C FIFO Receive Register
SCIFFCTC(2)	0x777C	1	SCI-C FIFO Control Register
SCIPRC	0x777F	1	SCI-C Priority Control Register

(1) Registers in this table are mapped to Peripheral Frame 2 space. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.
(2) These registers are new registers for the FIFO mode.

ADVANCE INFORMATION

Figure 4-15 shows the SCI module block diagram.

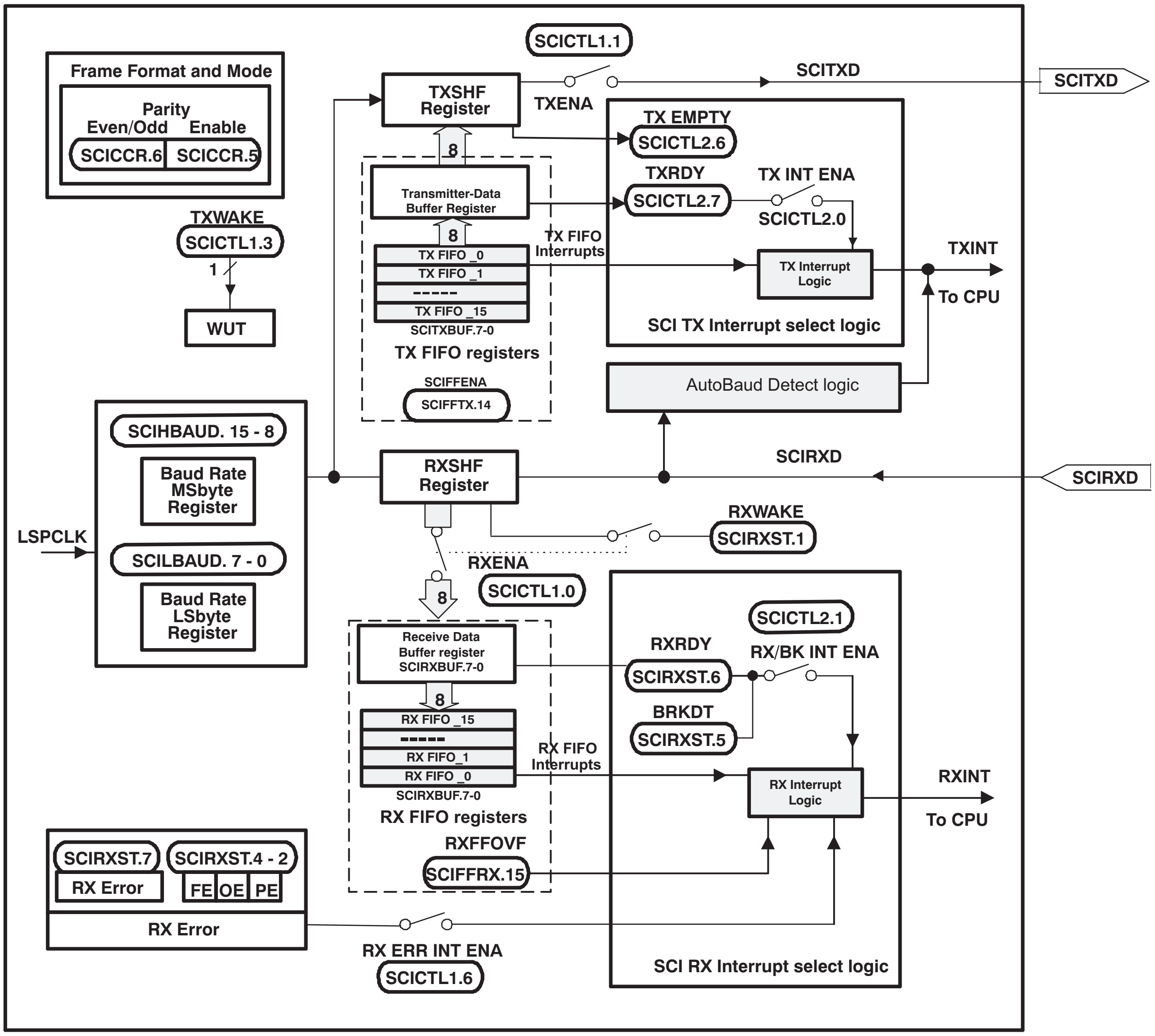


Figure 4-15. Serial Communications Interface (SCI) Module Block Diagram

4.11 Serial Peripheral Interface (SPI) Module (SPI-A)

The F2833x devices include the four-pin serial peripheral interface (SPI) module. One SPI module (SPI-A) is available. The SPI is a high-speed, synchronous serial I/O port that allows a serial bit stream of programmed length (one to sixteen 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 DSC 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:

- Four external pins:
  - SPISOMI: SPI slave-output/master-input pin
  - SPISIMO: SPI slave-input/master-output pin
  - $\overline{\text{SPISTE}}$: SPI slave transmit-enable pin
  - SPICLK: SPI serial-clock pin

NOTE: All four pins can be used as GPIO, if the SPI module is not used.

- Two operational modes: master and slave

  Baud rate: 125 different programmable rates.

$$\text{Baud rate} = \frac{\text{LSPCLK}}{(\text{SPIBRR} + 1)} \qquad \text{when SPIBRR} = 3 \text{ to } 127$$

$$\text{Baud rate} = \frac{\text{LSPCLK}}{4} \qquad \text{when SPIBRR} = 0,1,\ 2$$

- Data word length: one to sixteen 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 operation (transmit function can be disabled in software)
- Transmitter and receiver operations are accomplished through either interrupt-driven or polled algorithms.
- Nine SPI module control registers: Located in control register frame beginning at address 7040h.

NOTE

All registers in this module are 16-bit registers that are connected to Peripheral Frame 2. 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.

Enhanced feature:

- 16-level transmit/receive FIFO
- Delayed transmit control

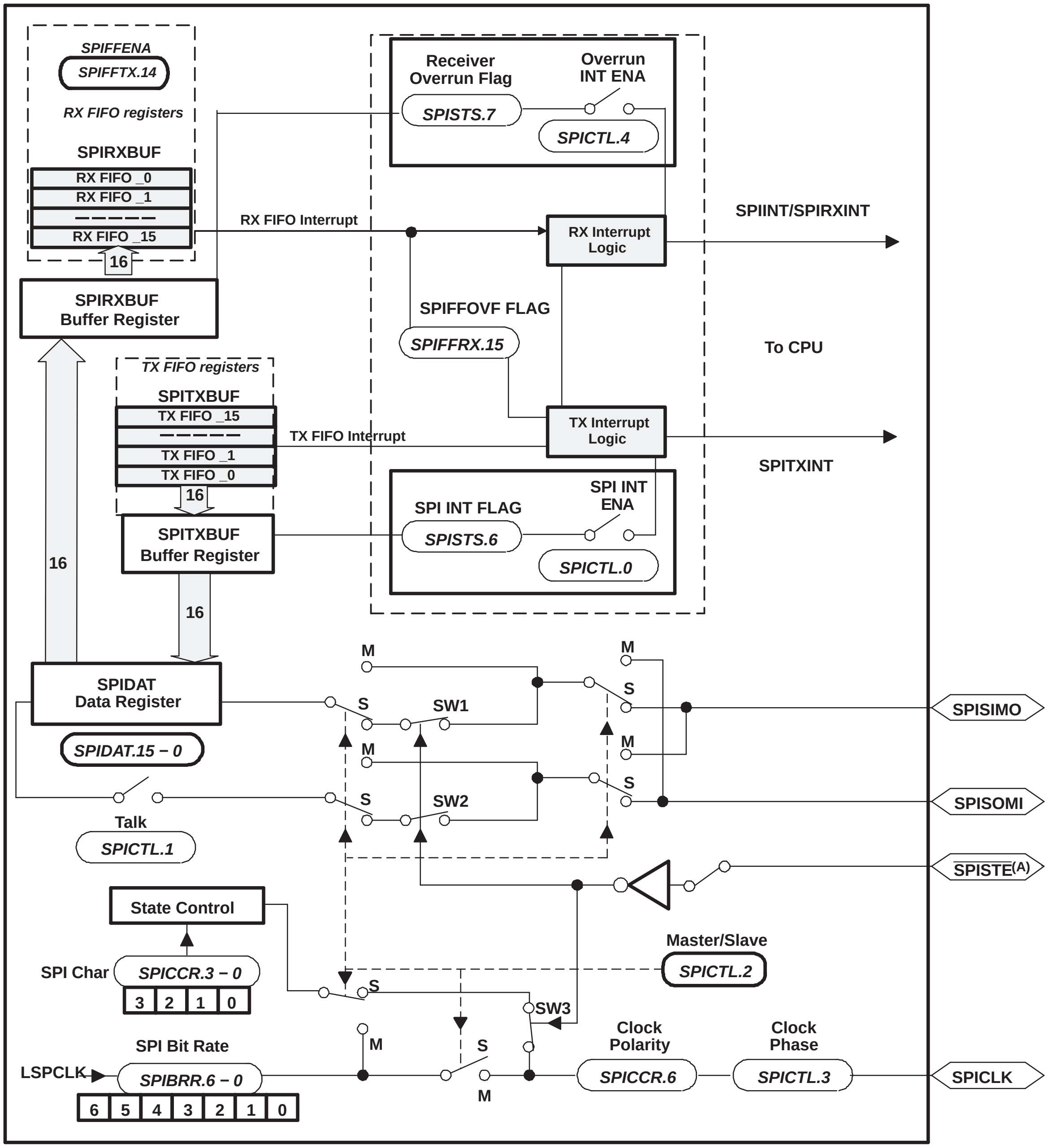
The SPI port operation is configured and controlled by the registers listed in Table 4-12.

Table 4-12. SPI-A Registers

NAME	ADDRESS	SIZE (X16)	DESCRIPTION[1]
SPICCR	0x7040	1	SPI-A Configuration Control Register
SPICTL	0x7041	1	SPI-A Operation Control Register
SPISTS	0x7042	1	SPI-A Status Register
SPIBRR	0x7044	1	SPI-A Baud Rate Register
SPIRXEMU	0x7046	1	SPI-A Receive Emulation Buffer Register
SPIRXBUF	0x7047	1	SPI-A Serial Input Buffer Register
SPITXBUF	0x7048	1	SPI-A Serial Output Buffer Register
SPIDAT	0x7049	1	SPI-A Serial Data Register
SPIFFTX	0x704A	1	SPI-A FIFO Transmit Register
SPIFFRX	0x704B	1	SPI-A FIFO Receive Register
SPIFFCT	0x704C	1	SPI-A FIFO Control Register
SPIPRI	0x704F	1	SPI-A Priority Control Register

(1) Registers in this table are mapped to Peripheral Frame 2. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.

Figure 4-16 is a block diagram of the SPI in slave mode.



A. SPISTE is driven low by the master for a slave device.

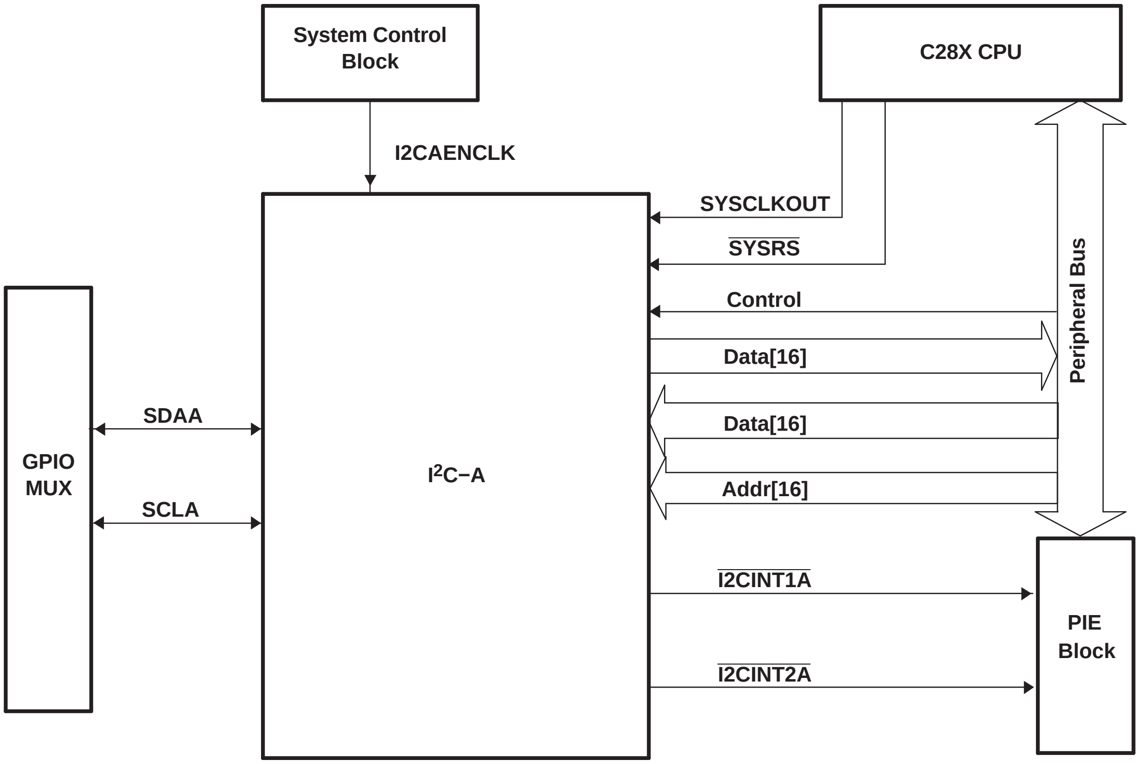
Figure 4-16. SPI Module Block Diagram (Slave Mode)

4.12 Inter-Integrated Circuit (I2C)

The F2833x device contains one I2C Serial Port. Figure 4-15 shows how the I2C peripheral module interfaces within the F2833x device.

The I2C module has the following features:

- Compliance with the Philips Semiconductors I2C-bus specification (version 2.1):
  - Support for 1-bit to 8-bit format transfers
  - 7-bit and 10-bit addressing modes
  - General call
  - START byte mode
  - Support for multiple master-transmitters and slave-receivers
  - Support for multiple slave-transmitters and master-receivers
  - Combined master transmit/receive and receive/transmit mode
  - Data transfer rate of from 10 kbps up to 400 kbps (Philips Fast-mode rate)
- One 16-bit receive FIFO and one 16-bit transmit FIFO
- One interrupt that can be used by the CPU. This interrupt can be generated as a result of one of the following conditions:
  - Transmit-data ready
  - Receive-data ready
  - Register-access ready
  - No-acknowledgment received
  - Arbitration lost
  - Stop condition detected
  - Addressed as slave
- An additional interrupt that can be used by the CPU when in FIFO mode
- Module enable/disable capability
- Free data format mode



A. The I2C registers are accessed at the SYSCLKOUT rate. The internal timing and signal waveforms of the I2C port are also at the SYSCLKOUT rate.

B. The clock enable bit (I2CAENCLK) in the PCLKCRO register turns off the clock to the I2C port for low power operation. Upon reset, I2CAENCLK is clear, which indicates the peripheral internal clocks are off.

Figure 4-17. I2C Peripheral Module Interfaces

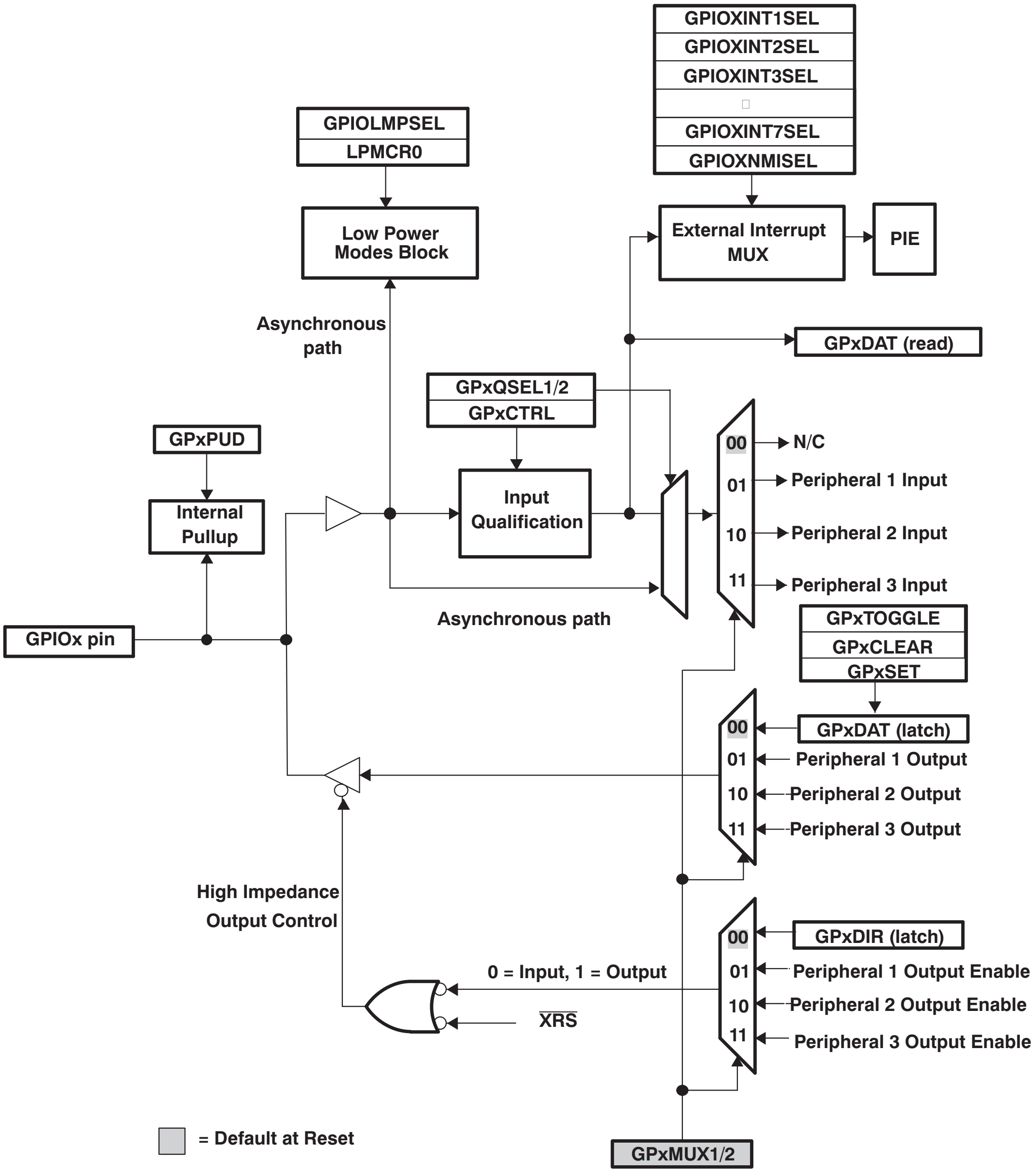
The registers in Table 4-13 configure and control the I2C port operation.

Table 4-13. I2C-A Registers

NAME	ADDRESS	DESCRIPTION
I2COAR	0x7900	I2C own address register
I2CIER	0x7901	I2C interrupt enable register
I2CSTR	0x7902	I2C status register
I2CCLKL	0x7903	I2C clock low-time divider register
I2CCLKH	0x7904	I2C clock high-time divider register
I2CCNT	0x7905	I2C data count register
I2CDRR	0x7906	I2C data receive register
I2CSAR	0x7907	I2C slave address register
I2CDXR	0x7908	I2C data transmit register
I2CMDR	0x7909	I2C mode register
I2CISRC	0x790A	I2C interrupt source register
I2CPSC	0x790C	I2C prescaler register
I2CFFTX	0x7920	I2C FIFO transmit register
I2CFFRX	0x7921	I2C FIFO receive register
I2CRSR	-	I2C receive shift register (not accessible to the CPU)
I2CXSR	-	I2C transmit shift register (not accessible to the CPU)

4.13 GPIO MUX

On the F2833x devices, the GPIO MUX can multiplex up to three independent peripheral signals on a single GPIO pin in addition to providing individual pin bit-banging IO capability. The GPIO MUX block diagram per pin is shown in Figure 4-18. Because of the open drain capabilities of the I2C pins, the GPIO MUX block diagram for these pins differ. See the *TMS320F2833x Digital Signal Controller (DSC) System Control and Interrupts Reference Guide* (literature number SPRUFB0) for details.



A. x stands for the port, either A or B. For example, GPxDIR refers to either the GPADIR and GPBDIR register depending on the particular GPIO pin selected.

B. GPxDAT latch/read are accessed at the same memory location.

C. This is a generic GPIO MUX block diagram. Not all options may be applicable for all GPIO pins. See the *TMS320x2833x System Control and Interrupts Reference Guide* (literature number SPRUFB0) for pin-specific variations.

Figure 4-18. GPIO MUX Block Diagram

ADVANCE INFORMATION

The F2833x supports 88 GPIO pins. The GPIO control and data registers are mapped to Peripheral Frame 1 to enable 32-bit operations on the registers (along with 16-bit operations). Table 4-14 shows the GPIO register mapping.

Table 4-14. GPIO Registers

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
GPIO CONTROL REGISTERS (EALLOW PROTECTED)			
GPACTRL	0x6F80	2	GPIO A Control Register (GPIO0 to 31)
GPAQSEL1	0x6F82	2	GPIO A Qualifier Select 1 Register (GPIO0 to 15)
GPAQSEL2	0x6F84	2	GPIO A Qualifier Select 2 Register (GPIO16 to 31)
GPAMUX1	0x6F86	2	GPIO A MUX 1 Register (GPIO0 to 15)
GPAMUX2	0x6F88	2	GPIO A MUX 2 Register (GPIO16 to 31)
GPADIR	0x6F8A	2	GPIO A Direction Register (GPIO0 to 31)
GPAPUD	0x6F8C	2	GPIO A Pull Up Disable Register (GPIO0 to 31)
Reserved	0x6F8E – 0x6F8F	2	
GPBCTRL	0x6F90	2	GPIO B Control Register (GPIO32 to 35)
GPBQSEL1	0x6F92	2	GPIO B Qualifier Select 1 Register (GPIO32 to 35)
GPBQSEL2	0x6F94	2	Reserved
GPBMUX1	0x6F96	2	GPIO B MUX 1 Register (GPIO32 to 35)
GPBMUX2	0x6F98	2	GPIO B MUX 2 Register (GPIO48 to 63)
GPBDIR	0x6F9A	2	GPIO B Direction Register (GPIO32 to 35)
GPBPUD	0x6F9C	2	GPIO B Pull Up Disable Register (GPIO32 to 35)
Reserved	0x6F9E – 0x6FA5	8	
GPCMUX1	0x6FA6	2	GPIO C MUX1 Register (GPIO64 to 79)
GPCMUX2	0x6FA8	2	GPIO C MUX2 Register (GPIO80 to 87)
GPCDIR	0x6FAA	2	GPIO C Direction Register (GPIO64 to 87)
GPCPUD	0x6FAC	2	GPIO C Pull Up Disable Register (GPIO64 to 87)
Reserved	0x6FAE – 0x6FBF	18	
GPIO DATA REGISTERS (NOT EALLOW PROTECTED)			
GPADAT	0x6FC0	2	GPIO A Data Register (GPIO0 to 31)
GPASET	0x6FC2	2	GPIO A Data Set Register (GPIO0 to 31)
GPACLEAR	0x6FC4	2	GPIO A Data Clear Register (GPIO0 to 31)
GPATOGGLE	0x6FC6	2	GPIO A Data Toggle Register (GPIO0 to 31)
GPBDAT	0x6FC8	2	GPIO B Data Register (GPIO32 to 35)
GPBSET	0x6FCA	2	GPIO B Data Set Register (GPIO32 to 35)
GPBCLEAR	0x6FCC	2	GPIO B Data Clear Register (GPIO32 to 35)
GPBTOGGLE	0x6FCE	2	GPIOB Data Toggle Register (GPIO32 to 35)
GPCDAT	0x6FD0	2	GPIO C Data Register (GPIO64 to 87)
GPCSET	0x6FD2	2	GPIO C Data Set Register (GPIO64 to 87)
GPCCLEAR	0x6FD4	2	GPIO C Data Clear Register (GPIO64 to 87)
GPCTOGGLE	0x6FD6	2	GPIO C Data Toggle Register (GPIO64 to 87)
Reserved	0x6FD8 0x6FDF	8	
GPIO INTERRUPT AND LOW POWER MODES SELECT REGISTERS (EALLOW PROTECTED)			
GPIOXINT1SEL	0x6FE0	1	XINT1 GPIO Input Select Register (GPIO0 to 31)
GPIOXINT2SEL	0x6FE1	1	XINT2 GPIO Input Select Register (GPIO0 to 31)
GPIOXNMISEL	0x6FE2	1	XNMI GPIO Input Select Register (GPIO0 to 31)
GPIOXINT3SEL	0x6FE3	1	XINT3 GPIO Input Select Register (GPIO32 to 63)
GPIOXINT4SEL	0x6FE4	1	XINT4 GPIO Input Select Register (GPIO32 to 63)
GPIOXINT5SEL	0x6FE5	1	XINT5 GPIO Input Select Register (GPIO32 to 63)

Table 4-14. GPIO Registers (continued)

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
GPIOXINT6SEL	0x6FE6	1	XINT6 GPIO Input Select Register (GPIO32 to 63)
GPIOINT7SEL	0x6FE7	1	XINT7 GPIO Input Select Register (GPIO32 to 63)
GPIOLPMSEL	0x6FE8	2	LPM GPIO Select Register (GPIO0 to 31)
Reserved	0x6FEA – 0x6FFF	22	

Table 4-15. GPIO-A Mux Peripheral Selection Matrix

REGISTER BITS			PERIPHERAL SELECTION			
GPADIR GPADAT GPASET GPACLR GPATOGGLE		GPAMUX1 GPAQSEL1	GPIOx GPAMUX1=0,0	PER1 GPAMUX1 = 0, 1	PER2 GPAMUX1 = 1, 0	PER3 GPAMUX1 = 1, 1
QUALPRD0	0	1, 0	GPIO0 (I/O)	EPWM1A (O)		
	1	3, 2	GPIO1 (I/O)	EPWM1B (O)	ECAP6 (I/O)	MFSRB (I/O)
	2	5, 4	GPIO2 (I/O)	EPWM2A (O)		
	3	7, 6	GPIO3 (I/O)	EPWM2B (O)	ECAP5 (I/O)	MCLKRB (I/O)
	4	9, 8	GPIO4 (I/O)	EPWM3A (O)		
	5	11, 10	GPIO5 (I/O)	EPWM3B (O)	MFSRA (I/O)	ECAP1 (I/O)
	6	13, 12	GPIO6 (I/O)	EPWM4A (O)	EPWMSYNCI (I)	EPWMSYNCO (O)
	7	15, 14	GPIO7 (I/O)	EPWM4B (O)	MCLKRA (I/O)	ECAP2 (I/O)
QUALPRD1	8	17, 16	GPIO8 (I/O)	EPWM5A (O)	CANTXB (O)	$\overline{\text{ADCSOCAO}}$ (O)
	9	19, 18	GPIO9 (I/O)	EPWM5B (O)	SCITXDB (O)	ECAP3 (I/O)
	10	21, 20	GPIO10 (I/O)	EPWM6A (O)	CANRXB (I)	$\overline{\text{ADCSOCBO}}$ (O)
	11	23, 22	GPIO11 (I/O)	EPWM6B (O)	SCIRXDB (I)	ECAP4 (I/O)
	12	25, 24	GPIO12 (I/O)	$\overline{\text{TZ1}}$ (I)	CANTXB (O)	MDXB (O)
	13	27, 26	GPIO13 (I/O)	$\overline{\text{TZ2}}$ (I)	CANRXB (I)	MDRB (I)
	14	29, 28	GPIO14 (I/O)	$\overline{\text{TZ3}}$ (I)/$\overline{\text{XHOLD}}$ (I)	SCITXDB (O)	MCLKXB (I/O)
	15	31, 30	GPIO15 (I/O)	$\overline{\text{TZ4}}$ (I)/$\overline{\text{XHOLDA}}$ (O)	SCIRXDB (I)	MFSXB (I/O)
		GPAMUX2 GPAQSEL2	GPAMUX2 =0, 0	GPAMUX2 = 0, 1	GPAMUX2 = 1, 0	GPAMUX2 = 1, 1
QUALPRD2	16	1, 0	GPIO16 (I/O)	SPISIMOA (I/O)	CANTXB (O)	$\overline{\text{TZ5}}$ (I)
	17	3, 2	GPIO17 (I/O)	SPISOMIA (I/O)	CANRXB (I)	$\overline{\text{TZ6}}$ (I)
	18	5, 4	GPIO18 (I/O)	SPICLKA (I/O)	SCITXDB (O)	CANRXA (I)
	19	7, 6	GPIO19 (I/O)	$\overline{\text{SPISTEA}}$ (I/O)	SCIRXDB (I)	CANTXA (O)
	20	9, 8	GPIO20 (I/O)	EQEP1A (I)	MDXA (O)	CANTXB (O)
	21	11, 10	GPIO21 (I/O)	EQEP1B (I)	MDRA (I)	CANRXB (I)
	22	13, 12	GPIO22 (I/O)	EQEP1S (I/O)	MCLKXA (I/O)	SCITXDB (O)
	23	15, 14	GPIO23 (I/O)	EQEP1I (I/O)	MFSXA (I/O)	SCIRXDB (I)
QUALPRD3	24	17, 16	GPIO24 (I/O)	ECAP1 (I/O)	EQEP2A (I)	MDXB (O)
	25	19, 18	GPIO25 (I/O)	ECAP2 (I/O)	EQEP2B (I)	MDRB (I)
	26	21, 20	GPIO26 (I/O)	ECAP3 (I/O)	EQEP2I (I/O)	MCLKXB (I/O)
	27	23, 22	GPIO27 (I/O)	ECAP4 (I/O)	EQEP2S (I/O)	MFSXB (I/O)
	28	25, 24	GPIO28 (I/O)	SCIRXDA (I)	$\overline{\text{XZCS6}}$ (O)	
	29	27, 26	GPIO29 (I/O)	SCITXDA (O)	XA19 (O)	
	30	29, 28	GPIO30 (I/O)	CANRXA (I)	XA18 (O)	
	31	31, 30	GPIO31 (I/O)	CANTXA (O)	XA17 (O)	

Table 4-16. GPIO-B Mux Peripheral Selection Matrix

REGISTER BITS			PERIPHERAL SELECTION			
GPBDIR GPBDAT GPBSET GPBCLR GPBTOGGLE		**GPBMUX1 GPBQSEL1**	**GPIOx GPBMUX1=0, 0**	**PER1 GPBMUX1 = 0, 1**	**PER2 GPBMUX1 = 1, 0**	**PER3 GPBMUX1 = 1, 1**
QUALPRD0	0	1, 0	GPIO32 (I/O)	SDAA (I/OC)[1]	EPWMSYNCI (I)	$\overline{\text{ADCSOCAO}}$ (O)
	1	3, 2	GPIO33 (I/O)	SCLA (I/OC)[1]	EPWMSYNCO (O)	$\overline{\text{ADCSOCBO}}$ (O)
	2	5, 4	GPIO34 (I/O)	ECAP1 (I/O)	XREADY (I)	
	3	7, 6	GPIO35 (I/O)	SCITXDA (O)	XR/$\overline{\text{W}}$ (O)	
	4	9, 8	GPIO36 (I/O)	SCIRXDA (I)	$\overline{\text{XZCS0}}$ (O)	
	5	11, 10	GPIO37 (I/O)	ECAP2 (I/O)	$\overline{\text{XZCS7}}$ (O)	
	6	13, 12	GPIO38 (I/O)	Reserved	$\overline{\text{XWE0}}$ (O)	
	7	15, 14	GPIO39 (I/O)		XA16 (O)	
QUALPRD1	8	17, 16	GPIO40 (I/O)		XA0/$\overline{\text{XWE1}}$ (O)	
	9	19, 18	GPIO41 (I/O)		XA1 (O)	
	10	21, 20	GPIO42 (I/O)		XA2 (O)	
	11	23, 22	GPIO43 (I/O)		XA3 (O)	
	12	25, 24	GPIO44 (I/O)		XA4 (O)	
	13	27, 26	GPIO45 (I/O)		XA5 (O)	
	14	29, 28	GPIO46 (I/O)		XA6 (O)	
	15	31, 30	GPIO47 (I/O)		XA7 (O)	
		GPBMUX2 GPBQSEL2	**GPBMUX2 =0, 0**	**GPBMUX2 = 0, 1**	**GPBMUX2 = 1, 0**	**GPBMUX2 = 1, 1**
QUALPRD2	16	1, 0	GPIO48 (I/O)	ECAP5 (I/O)	XD31 (I/O)	
	17	3, 2	GPIO49 (I/O)	ECAP6 (I/O)	XD30 (I/O)	
	18	5, 4	GPIO50 (I/O)	EQEP1A (I)	XD29 (I/O)	
	19	7, 6	GPIO51 (I/O)	EQEP1B (I)	XD28 (I/O)	
	20	9, 8	GPIO52 (I/O)	EQEP1S (I/O)	XD27 (I/O)	
	21	11, 10	GPIO53 (I/O)	EQEP1I (I/O)	XD26 (I/O)	
	22	13, 12	GPIO54 (I/O)	SPISIMOA (I/O)	XD25 (I/O)	
	23	15, 14	GPIO55 (I/O)	SPISOMIA (I/O)	XD24 (I/O)	
QUALPRD3	24	17, 16	GPIO56 (I/O)	SPICLKA (I/O)	XD23 (I/O)	
	25	19, 18	GPIO57 (I/O)	$\overline{\text{SPISTEA}}$ (I/O)	XD22 (I/O)	
	26	21, 20	GPIO58 (I/O)	MCLKRA (I/O)	XD21 (I/O)	
	27	23, 22	GPIO59 (I/O)	MFSRA (I/O)	XD20 (I/O)	
	28	25, 24	GPIO60 (I/O)	MCLKRB (I/O)	XD19 (I/O)	
	29	27, 26	GPIO61 (I/O)	MFSRB (I/O)	XD18 (I/O)	
	30	29, 28	GPIO62 (I/O)	SCIRXDC (I)	XD17 (I/O)	
	31	31, 30	GPIO63 (I/O)	SCITXDC (O)	XD16 (I/O)	

(1) Open drain

Table 4-17. GPIO-C Mux Peripheral Selection Matrix

REGISTER BITS		**PERIPHERAL SELECTION**		
GPCDIR GPCDAT GPCSET GPCCLR GPCTOGGLE		**GPCMUX1**	**GPIOx or PER1 GPCMUX1 = 0, 0 or 0, 1**	**PER2 or PER3 GPCMUX1 = 1, 0 or 1, 1**
no qual	0	1, 0	GPIO64 (I/O)	XD15 (I/O)
	1	3, 2	GPIO65 (I/O)	XD14 (I/O)
	2	5, 4	GPIO66 (I/O)	XD13 (I/O)
	3	7, 6	GPIO67 (I/O)	XD12 (I/O)
	4	9, 8	GPIO68 (I/O)	XD11 (I/O)
	5	11, 10	GPIO69 (I/O)	XD10 (I/O)
	6	13, 12	GPIO70 (I/O)	XD9 (I/O)
	7	15, 14	GPIO71 (I/O)	XD8 (I/O)
no qual	8	17, 16	GPIO72 (I/O)	XD7 (I/O)
	9	19, 18	GPIO73 (I/O)	XD6 (I/O)
	10	21, 20	GPIO74 (I/O)	XD5 (I/O)
	11	23, 22	GPIO75 (I/O)	XD4 (I/O)
	12	25, 24	GPIO76 (I/O)	XD3 (I/O)
	13	27, 26	GPIO77 (I/O)	XD2 (I/O)
	14	29, 28	GPIO78 (I/O)	XD1 (I/O)
	15	31, 30	GPIO79 (I/O)	XD0 (I/O)
		GPCMUX2	**GPCMUX2 = 0, 0 or 0, 1**	**GPCMUX2 = 1, 0 or 1, 1**
no qual	16	1, 0	GPIO80 (I/O)	XA8 (O)
	17	3, 2	GPIO81 (I/O)	XA9 (O)
	18	5, 4	GPIO82 (I/O)	XA10 (O)
	19	7, 6	GPIO83 (I/O)	XA11 (O)
	20	9, 8	GPIO84 (I/O)	XA12 (O)
	21	11, 10	GPIO85 (I/O)	XA13 (O)
	22	13, 12	GPIO86 (I/O)	XA14 (O)
	23	15, 14	GPIO87 (I/O)	XA15 (O)

The user can select the type of input qualification for each GPIO pin via the GPxQSEL1/2 registers from four choices:

- Synchronization To SYSCLKOUT Only (GPxQSEL1/2=0, 0): This is the default mode of all GPIO pins at reset and it simply synchronizes the input signal to the system clock (SYSCLKOUT).
- Qualification Using Sampling Window (GPxQSEL1/2=0, 1 and 1, 0): In this mode the input signal, after synchronization to the system clock (SYSCLKOUT), is qualified by a specified number of cycles before the input is allowed to change.

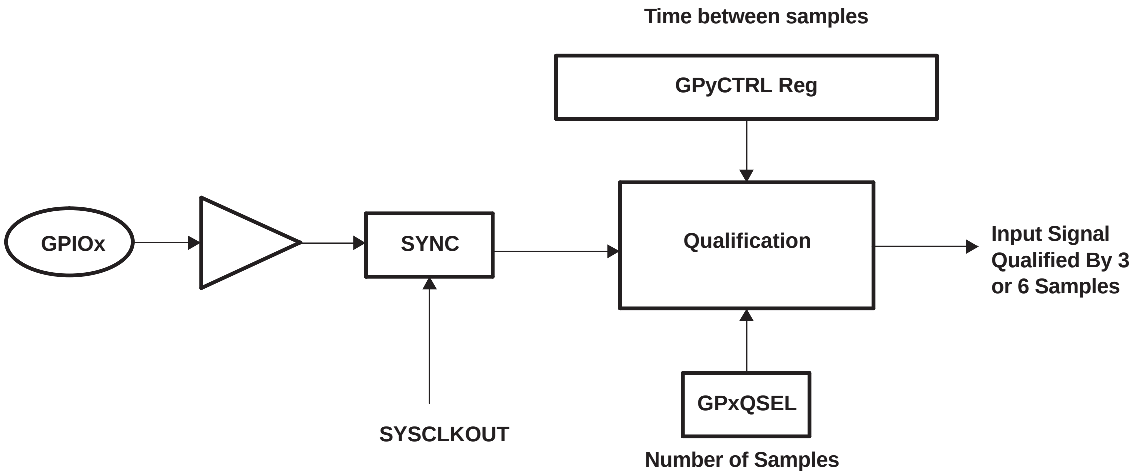


Figure 4-19. Qualification Using Sampling Window

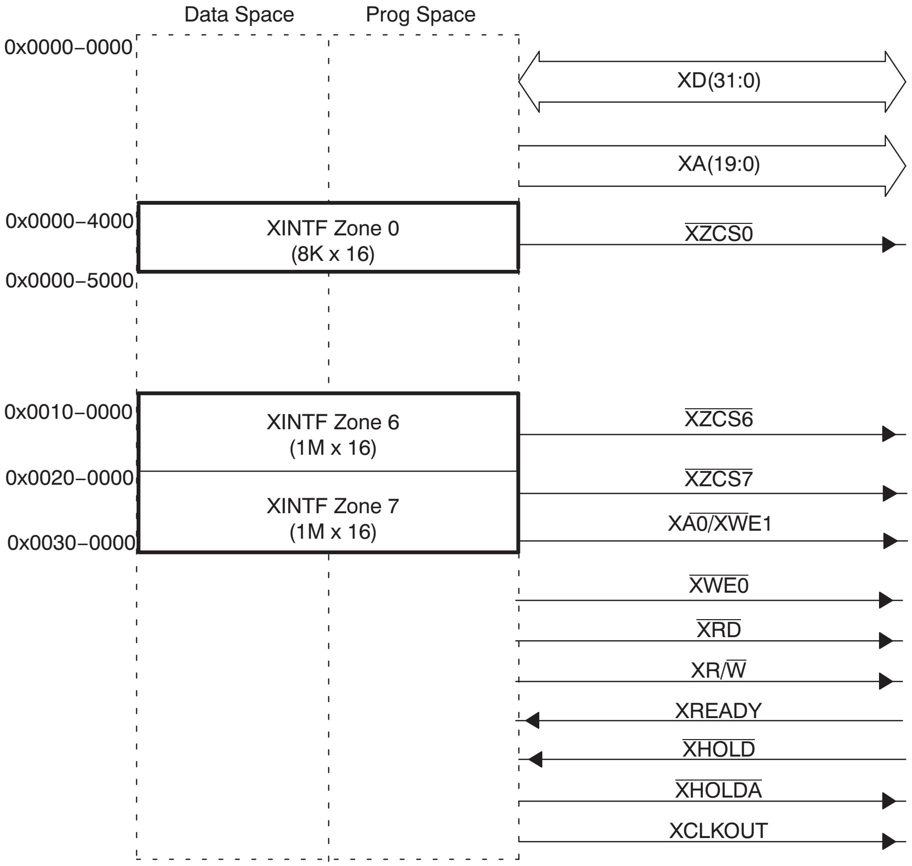
- The sampling period is specified by the QUALPRD bits in the GPxCTRL register and is configurable in groups of 8 signals. It specifies a multiple of SYSCLKOUT cycles for sampling the input signal. The sampling window is either 3-samples or 6-samples wide and the output is only changed when ALL samples are the same (all 0s or all 1s) as shown in Figure 4-18 (for 6 sample mode).
- No Synchronization (GPxQSEL1/2=1,1): This mode is used for peripherals where synchronization is not required (synchronization is performed within the peripheral).

Due to the multi-level multiplexing that is required on the F2833x device, there may be cases where a peripheral input signal can be mapped to more then one GPIO pin. Also, when an input signal is not selected, the input signal will default to either a 0 or 1 state, depending on the peripheral.

4.14 External Interface (XINTF)

This section gives a top-level view of the external interface (XINTF) that is implemented on the F2833x devices.

The XINTF is a non-multiplexed asynchronous bus, similar to the 2812 XINTF. The XINTF on the F2833x is mapped into three fixed zones shown in Figure 4-20.



A. Each zone can be programmed with different wait states, setup and hold timings, and is supported by zone chip selects that toggle when an access to a particular zone is performed. These features enable glueless connection to many external memories and peripherals.

B. Zones 1 – 5 are reserved for future expansion.

C. Zones 0, 6, and 7 are always enabled.

Figure 4-20. External Interface Block Diagram

Figure 4-21 and Figure 4-22 show typical 16-bit and 32-bit data bus XINTF connections, illustrating how the functionality of the XA0/XWE1 signal changes, depending on the configuration. Table 4-18 defines XINTF configuration and control registers.

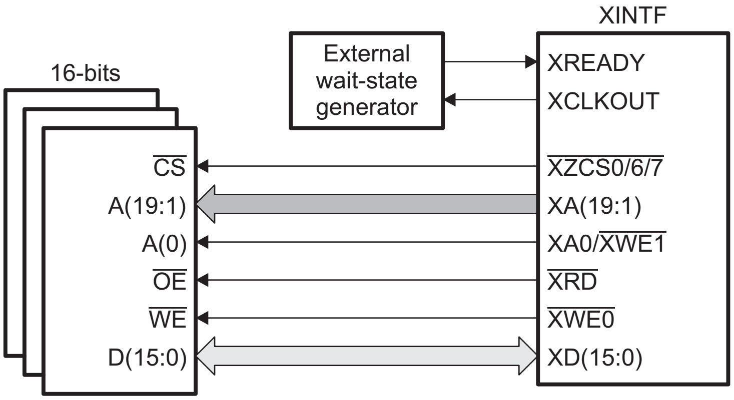


Figure 4-21. Typical 16-bit Data Bus XINTF Connections

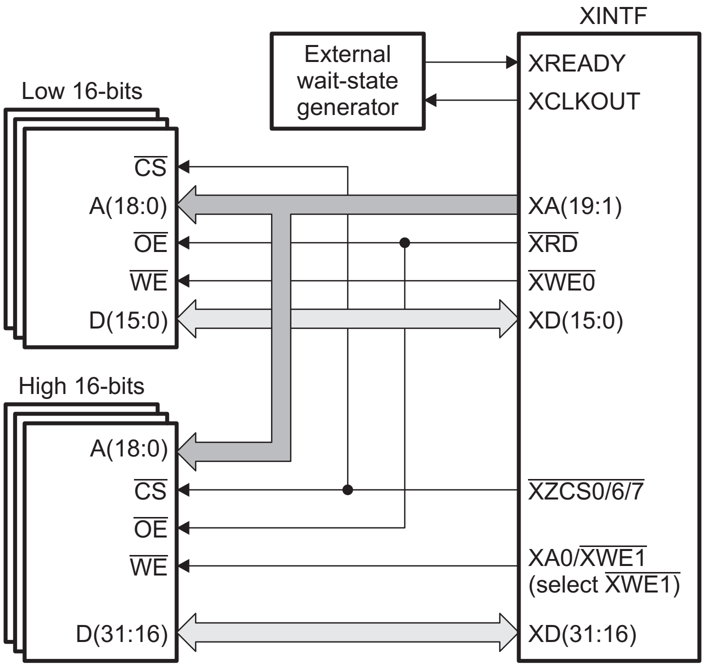


Figure 4-22. Typical 32-bit Data Bus XINTF Connections

Table 4-18. XINTF Configuration and Control Register Mapping

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
XTIMING0	0x0000–0B20	2	XINTF Timing Register, Zone 0
XTIMING6(1)	0x0000–0B2C	2	XINTF Timing Register, Zone 6
XTIMING7	0x0000–0B2E	2	XINTF Timing Register, Zone 7
XINTCNF2(2)	0x0000–0B34	2	XINTF Configuration Register
XBANK	0x0000–0B38	1	XINTF Bank Control Register
XREVISION	0x0000–0B3A	1	XINTF Revision Register
XRESET	0x0000 083D	1	XINTF Reset Register

(1) XTIMING1 - XTIMING5 are reserved for future expansion and are not currently used.
(2) XINTCNF1 is reserved and not currently used.

5 Device Support

Texas Instruments (TI) offers an extensive line of development tools for the C28x™ generation of DSCs, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules.

The following products support development of 2833x-based applications:

Software Development Tools

- Code Composer Studio™ Integrated Development Environment (IDE)
  - C/C++ Compiler
  - Code generation tools
  - Assembler/Linker
  - Cycle Accurate Simulator
- Application algorithms
- Sample applications code

Hardware Development Tools

- 2833x development board
- Evaluation modules
- JTAG-based emulators - SPI515, XDS510PP, XDS510PP Plus, XDS510USB
- Universal 5-V dc power supply
- Documentation and cables

5.1 Device and Development Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all TMS320™ DSC devices and support tools. Each TMS320™ DSP commercial family member has one of three prefixes: TMX, TMP, or TMS (e.g., **TMS**320F28335). Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS).

Device development evolutionary flow:

TMX	Experimental device that is not necessarily representative of the final device's electrical specifications
TMP	Final silicon die that conforms to the device's electrical specifications but has not completed quality and reliability verification
TMS	Fully qualified production device

Support tool development evolutionary flow:

TMDX	Development-support product that has not yet completed Texas Instruments internal qualification testing
TMDS	Fully qualified development-support product

TMX and TMP devices and TMDX development-support tools are shipped against the following disclaimer:
"Developmental product is intended for internal evaluation purposes."

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

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

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, PBK) and temperature range (for example, A). Figure 5-1 provides a legend for reading the complete device name for any family member.

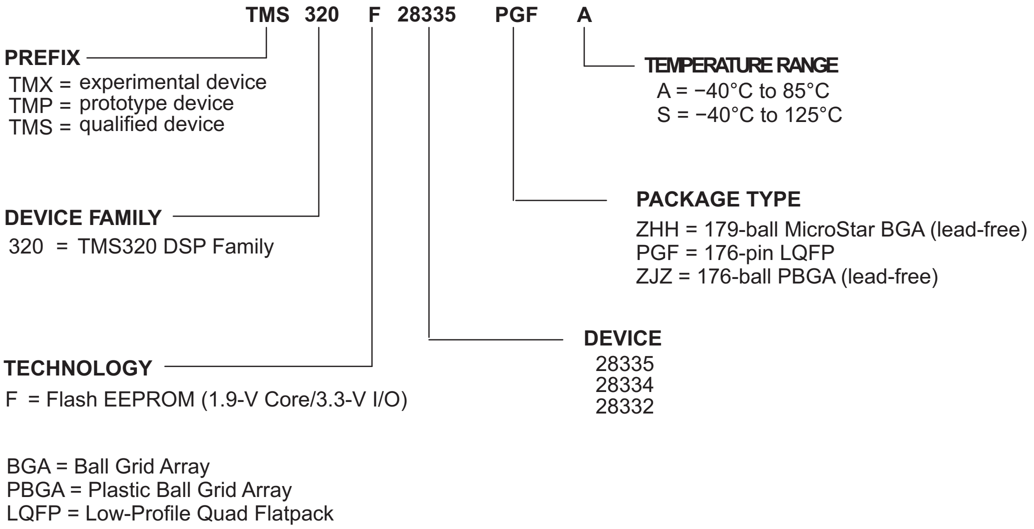


Figure 5-1. Example of F2833x Device Nomenclature

5.2 Documentation Support

Extensive documentation supports all of the TMS320™ DSP family generations of devices from product announcement through applications development. The types of documentation available include: data sheets and data manuals, with design specifications; and hardware and software applications. Useful reference documentation includes:

CPU User's Guides

SPRU430	TMS320C28x DSP CPU and Instruction Set Reference Guide describes the central processing unit (CPU) and the assembly language instructions of the TMS320C28x fixed-point digital signal processors (DSPs). It also describes emulation features available on these DSPs.
SPRUEO2	TMS320C28x Floating Point Unit and Instruction Set Reference Guide describes the floating-point unit and includes the instructions for the FPU.

Peripheral Guides

SPRU566	TMS320x28xx, 28xxx Peripheral Reference Guide describes the peripheral reference guides of the 28x digital signal processors (DSPs).
SPRUFB0	TMS320x2833x System Control and Interrupts Reference Guide describes the various interrupts and system control features of the 2833x digital signal controllers (DSCs).
SPRU812	TMS320x2833x Analog-to-Digital Converter (ADC) Reference Guide describes how to configure and use the on-chip ADC module, which is a 12-bit pipelined ADC.
SPRU949	TMS320x2833x External Interface (XINTF) User's Guide describes the XINTF, which is a nonmultiplexed asynchronous bus, as it is used on the 2833x devices.
SPRU963	TMS320x2833x Boot ROM User's Guide describes the purpose and features of the bootloader (factory-programmed boot-loading software) and provides examples of code. It also describes other contents of the device on-chip boot ROM and identifies where all of the information is located within that memory.
SPRUFB7	TMS320x2833x Multichannel Buffered Serial Port (McBSP) User's Guide describes the McBSP available on the F2833x devices. The McBSPs allow direct interface between a DSP and other devices in a system.
SPRUFB8	TMS320x2833x Direct Memory Access (DMA) Reference Guide describes the DMA on the 2833x devices.
SPRU791	TMS320x28xx, 28xxx Enhanced Pulse Width Modulator (ePWM) Module Reference Guide describes the main areas of the enhanced pulse width modulator that include digital motor control, switch mode power supply control, UPS (uninterruptible power supplies), and other forms of power conversion.
SPRU924	TMS320x28xx, 28xxx High-Resolution Pulse Width Modulator (HRPWM) describes the operation of the high-resolution extension to the pulse width modulator (HRPWM).
SPRU807	TMS320x28xx, 28xxx Enhanced Capture (eCAP) Module Reference Guide describes the enhanced capture module. It includes the module description and registers.
SPRU790	TMS320x28xx, 28xxx Enhanced Quadrature Encoder Pulse (eQEP) Reference Guide describes the eQEP module, which is used for interfacing with a linear or rotary incremental encoder to get position, direction, and speed information from a rotating machine in high performance motion and position control systems. It includes the module description and registers.
SPRU074	TMS320x28xx, 28xxx Enhanced Controller Area Network (eCAN) Reference Guide describes the eCAN that uses established protocol to communicate serially with other controllers in electrically noisy environments.

SPRU051	TMS320x28xx, 28xxx Serial Communication Interface (SCI) Reference Guide describes the SCI, which is a two-wire asynchronous serial port, commonly known as a UART. The SCI modules support digital communications between the CPU and other asynchronous peripherals that use the standard non-return-to-zero (NRZ) format.
SPRU059	TMS320x28xx, 28xxx Serial Peripheral Interface (SPI) Reference Guide describes the SPI - a high-speed synchronous serial input/output (I/O) port - that allows a serial bit stream of programmed length (one to sixteen bits) to be shifted into and out of the device at a programmed bit-transfer rate.
SPRU721	TMS320x28xx, 28xxx Inter-Integrated Circuit (I2C) Reference Guide describes the features and operation of the inter-integrated circuit (I2C) module that is available on the TMS320x280x digital signal processor (DSP).

Tools Guides

SPRU513	TMS320C28x Assembly Language Tools User's Guide describes the assembly language tools (assembler and other tools used to develop assembly language code), assembler directives, macros, common object file format, and symbolic debugging directives for the TMS320C28x device.
SPRU514	TMS320C28x Optimizing C Compiler User's Guide describes the TMS320C28x™ C/C++ compiler. This compiler accepts ANSI standard C/C++ source code and produces TMS320 DSP assembly language source code for the TMS320C28x device.
SPRU608	The TMS320C28x Instruction Set Simulator Technical Overview describes the simulator, available within the Code Composer Studio for TMS320C2000 IDE, that simulates the instruction set of the C28x™ core.
SPRU625	TMS320C28x DSP/BIOS Application Programming Interface (API) Reference Guide describes development using DSP/BIOS.

Application Reports

SPRAAM0	Getting Started With TMS320C28x™ Digital Signal Controllers is organized by development flow and functional areas to make your design effort as seamless as possible. Tips on getting started with C28x™ DSP software and hardware development are provided to aid in your initial design and debug efforts. Each section includes pointers to valuable information including technical documentation, software, and tools for use in each phase of design.
SPRAAD5	Power Line Communication for Lighting Apps using BPSK w/ a Single DSP Controller presents a complete implementation of a power line modem following CEA-709 protocol using a single DSP.
SPRAA85	Programming TMS320x28xx and 28xxx Peripherals in C/C++ explores a hardware abstraction layer implementation to make C/C++ coding easier on 28x DSPs. This method is compared to traditional #define macros and topics of code efficiency and special case registers are also addressed.
SPRA958	Running an Application from Internal Flash Memory on the TMS320F28xx DSP covers the requirements needed to properly configure application software for execution from on-chip flash memory. Requirements for both DSP/BIOS™ and non-DSP/BIOS projects are presented. Example code projects are included.
SPRAA91	TMS320F280x DSC USB Connectivity Using TUSB3410 USB-to-UART Bridge Chip presents hardware connections as well as software preparation and operation of the development system using a simple communication echo program.
SPRAA58	TMS320x281x to TMS320x280x Migration Overview describes differences between the Texas Instruments TMS320x281x and TMS320x280x DSPs to assist in application migration from the 281x to the 280x. While the main focus of this document is migration from 281x to 280x, users considering migrating in the reverse direction (280x to 281x) will also find this

document useful.

SPRAAD8	TMS320280x and TMS320F2801x ADC Calibration describes a method for improving the absolute accuracy of the 12-bit ADC found on the TMS320280x and TMS3202801x devices. Inherent gain and offset errors affect the absolute accuracy of the ADC. The methods described in this report can improve the absolute accuracy of the ADC to levels better than 0.5%. This application report has an option to download an example program that executes from RAM on the F2808 EzDSP.
SPRAAI1	Using Enhanced Pulse Width Modulator (ePWM) Module for 0-100% Duty Cycle Control provides a guide for the use of the ePWM module to provide 0% to 100% duty cycle control and is applicable to the TMS320x280x family of processors.
SPRAA88	Using PWM Output as a Digital-to-Analog Converter on a TMS320F280x presents a method for utilizing the on-chip pulse width modulated (PWM) signal generators on the TMS320F280x family of digital signal controllers as a digital-to-analog converter (DAC).
SPRAAH1	Using the Enhanced Quadrature Encoder Pulse (eQEP) Module provides a guide for the use of the eQEP module as a dedicated capture unit and is applicable to the TMS320x280x, 28xxx family of processors.
SPRA820	Online Stack Overflow Detection on the TMS320C28x DSP presents the methodology for online stack overflow detection on the TMS320C28x™ DSP. C-source code is provided that contains functions for implementing the overflow detection on both DSP/BIOS™ and non-DSP/BIOS applications.
SPRA806	An Easy Way of Creating a C-callable Assembly Function for the TMS320C28x DSP provides instructions and suggestions to configure the C compiler to assist with understanding of parameter-passing conventions and environments expected by the C compiler.

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

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

To send comments regarding this data manual (literature number SPRS230), use the *comments@books.sc.ti.com* email address, which is a repository for feedback. For questions and support, contact the Product Information Center listed at the **http://www.ti.com/sc/docs/pic/home.htm** site.

6 Electrical Specifications

This section provides the absolute maximum ratings and the recommended operating conditions.

6.1 Absolute Maximum Ratings(1)(2)

Unless otherwise noted, the list of absolute maximum ratings are specified over operating temperature ranges.

Supply voltage range, V_{DDIO}, V_{DD3VFL}	with respect to V_{SS}	– 0.3 V to 4.6 V
Supply voltage range, V_{DDA2}, V_{DDAIO}	with respect to V_{SSA}	– 0.3 V to 4.6 V
Supply voltage range, V_{DD}	with respect to V_{SS}	– 0.3 V to 2.5 V
Supply voltage range, V_{DD1A18}, V_{DD2A18}	with respect to V_{SSA}	– 0.3 V to 2.5 V
Supply voltage range, V_{SSA2}, V_{SSAIO}, $V_{SS1AGND}$, $V_{SS2AGND}$	with respect to V_{SS}	– 0.3 V to 0.3 V
Input voltage range, V_{IN}		– 0.3 V to 4.6 V
Output voltage range, V_O		– 0.3 V to 4.6 V
Input clamp current, I_{IK} ($V_{IN} < 0$ or $V_{IN} > V_{DDIO}$)(3)		± 20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{DDIO}$)		± 20 mA
Operating ambient temperature ranges,	T_A: A version (4)	– 40°C to 85°C
	T_A: S version	– 40°C to 125°C
Junction temperature range, T_j(4)		– 40°C to 150°C
Storage temperature range, T_{stg}(4)		– 65°C to 150°C

(1) 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 Section 6.2 is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

(2) All voltage values are with respect to V_{SS}, unless otherwise noted.

(3) Continuous clamp current per pin is ± 2 mA. This includes the analog inputs which have an internal clamping circuit that clamps the voltage to a diode drop above V_{DDA2} or below V_{SSA2}.

(4) Long-term high-temperature storage and/or extended use at maximum temperature conditions may result in a reduction of overall device life. For additional information, see *IC Package Thermal Metrics Application Report* (literature number SPRA953) and *Reliability Data for TMS320LF24x and TMS320F281x Devices Application Report* (literature number SPRA963)

6.2 Recommended Operating Conditions

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

		MIN	NOM	MAX	UNIT
Device supply voltage, I/O, V_{DDIO}		3.2	3.3	3.4	V
Device supply voltage CPU, V_{DD}		1.84	1.9	1.96	V
Supply ground, V_{SS}, V_{SSIO}			0		V
ADC supply voltage (3.3 V), V_{DDA2}, V_{DDAIO}		3.2	3.3	3.4	V
ADC supply voltage (1.9 V), V_{DD1A18}, V_{DD2A18}		1.84	1.9	1.96	V
Flash supply voltage, V_{DD3VFL}		3.2	3.3	3.4	V
Device clock frequency (system clock), $f_{SYSCLKOUT}$	F28335, F28334	2		150	MHz
	F28332	2		100	
High-level input voltage, V_{IH}		2		V_{DDIO}	V
Low-level input voltage, V_{IL}				0.8	
High-level output source current, V_{OH} = 2.4 V, I_{OH}	All I/Os except Group 2			– 4	mA
	Group 2[1]			-8	
Low-level output sink current, V_{OL} = V_{OL} MAX, I_{OL}	All I/Os except Group 2			4	mA
	Group 2[1]			8	
Ambient temperature, T_A	A version	– 40		85	°C
	S version	– 40		125	
Junction temperature, T_j				125	°C

[1] Group 2 pins are as follows: GPIO28, GPIO29, GPIO30, GPIO31, TDO, XCLKOUT, EMU0, EMU1, XINTF pins, GPIO35-87, $\overline{XRD}$.

6.3 Electrical Characteristics

over recommended operating conditions (unless otherwise noted)

PARAMETER			TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage		I_{OH} = I_{OH}MAX		2.4			V
			I_{OH} = 50 µA		V_{DDIO} – 0.2			
V_{OL}	Low-level output voltage		I_{OL} = I_{OL}MAX				0.4	V
I_{IL}	Input current (low level)	Pin with pullup enabled	V_{DDIO} = 3.3 V, V_{IN} = 0 V	All I/Os (including $\overline{XRS}$)	– 80	– 140	– 190	µA
		Pin with pulldown enabled	V_{DDIO} = 3.3 V, V_{IN} = 0 V				± 2	
I_{IH}	Input current (high level)	Pin with pullup enabled	V_{DDIO} = 3.3 V, V_{IN} = V_{DDIO}				± 2	µA
		Pin with pulldown enabled	V_{DDIO} = 3.3 V, V_{IN} = V_{DDIO}		28	50	80	
I_{OZ}	Output current, pullup or pulldown disabled		V_O = V_{DDIO} or 0 V				± 2	µA
C_I	Input capacitance					2		pF

6.4 Current Consumption

Table 6-1. TMS320F28335 Current Consumption by Power-Supply Pins at 150-MHz SYSCLKOUT

MODE	TEST CONDITIONS	I_{DD}		I_{DDIO} (1)		I_{DD3VFL}		I_{DDA18} (2)		I_{DDA33} (3)	
		TYP (4)	MAX	TYP (4)	MAX	TYP	MAX	TYP (4)	MAX	TYP (4)	MAX
Operational (Flash) (5)	The following peripheral clocks are enabled: • ePWM1/2/3/4/5/6 • eCAP1/2/3/4/5/6 • eQEP1/2 • eCAN-A • SCI-A/B (FIFO mode) • SPI-A (FIFO mode) • ADC • I2C • CPU Timer 0/1/2 All PWM pins are toggled at 150 kHz. All I/O pins are left unconnected. (6)	290 mA		25 mA		35 mA	40 mA	30 mA	38 mA	1.5 mA	2 mA
IDLE	Flash is powered down. XCLKOUT is turned off. The following peripheral clocks are enabled: • eCAN-A • SCI-A • SPI-A • I2C	75 mA	90 mA	500 µA	2 mA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA
STANDBY	Flash is powered down. Peripheral clocks are off.	6 mA	12 mA	100 µA	500 µA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA
HALT	Flash is powered down. Peripheral clocks are off. Input clock is disabled.	70 µA		60 µA	120 µA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA

(1) I_{DDIO} current is dependent on the electrical loading on the I/O pins.
(2) I_{DDA18} includes current into V_{DD1A18} and V_{DD2A18} pins. In order to realize the I_{DDA18} currents shown for IDLE, STANDBY, and HALT, clock to the ADC module must be turned off explicitly by writing to the PCLKCR0 register.
(3) I_{DDA33} includes current into V_{DDA2} and V_{DDAIO} pins.
(4) The TYP numbers are applicable over room temperature and nominal voltage.
(5) When the identical code is run off SARAM, I_{DD} would increase as the code operates with zero wait states.
(6) The following is done in a loop:

- Data is continuously transmitted out of the SCI-A, SCI-B, SPI-A, McBSP-A, and eCAN-A ports.
- Floating-point multiplication and addition are performed.
- Watchdog is reset.
- ADC is performing continuous conversion. Data from ADC is transferred to SARAM through the DMA.
- 32-bit read/write of the XINTF is performed.
- GPIO19 is toggled.

NOTE

The peripheral - I/O multiplexing implemented in the F2833x device prevents all available peripherals from being used at the same time. This is because more than one peripheral function may share an I/O pin. It is, however, possible to turn on the clocks to all the peripherals at the same time, although such a configuration is not useful. If this is done, the current drawn by the device will be more than the numbers specified in the current consumption tables.

Table 6-2. TMS320F28334 Current Consumption by Power-Supply Pins at 150-MHz SYSCLKOUT

MODE	TEST CONDITIONS	I_{DD}		I_{DDIO} (1)		I_{DD3VFL}		I_{DDA18} (2)		I_{DDA33} (3)	
		TYP (4)	MAX	TYP (4)	MAX	TYP	MAX	TYP (4)	MAX	TYP (4)	MAX
Operational (Flash) (5)	The following peripheral clocks are enabled: • ePWM1/2/3/4/5/6 • eCAP1/2/3/4/5/6 • eQEP1/2 • eCAN-A • SCI-A/B (FIFO mode) • SPI-A (FIFO mode) • ADC • I2C • CPU Timer 0/1/2 All PWM pins are toggled at 150 kHz. All I/O pins are left unconnected. (6)	290 mA		25 mA		35 mA	40 mA	30 mA	38 mA	1.5 mA	2 mA
IDLE	Flash is powered down. XCLKOUT is turned off. The following peripheral clocks are enabled: • eCAN-A • SCI-A • SPI-A • I2C	75 mA	90 mA	500 µA	2 mA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA
STANDBY	Flash is powered down. Peripheral clocks are off.	6 mA	12 mA	100 µA	500 µA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA
HALT	Flash is powered down. Peripheral clocks are off. Input clock is disabled.	70 µA		60 µA	120 µA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA

(1) I_{DDIO} current is dependent on the electrical loading on the I/O pins.
(2) I_{DDA18} includes current into V_{DD1A18} and V_{DD2A18} pins. In order to realize the I_{DDA18} currents shown for IDLE, STANDBY, and HALT, clock to the ADC module must be turned off explicitly by writing to the PCLKCR0 register.
(3) I_{DDA33} includes current into V_{DDA2} and V_{DDAIO} pins.
(4) The TYP numbers are applicable over room temperature and nominal voltage.
(5) When the identical code is run off SARAM, I_{DD} would increase as the code operates with zero wait states.
(6) The following is done in a loop:
- Data is continuously transmitted out of the SCI-A, SCI-B, SPI-A, McBSP-A, and eCAN-A ports.
- Floating-point multiplication and addition are performed.
- Watchdog is reset.
- ADC is performing continuous conversion. Data from ADC is transferred to SARAM through the DMA.
- 32-bit read/write of the XINTF is performed.
- GPIO19 is toggled.

Table 6-3. TMS320F28332 Current Consumption by Power-Supply Pins at 100-MHz SYSCLKOUT

MODE	TEST CONDITIONS	I_{DD}		I_{DDIO} (1)		I_{DD3VFL}		I_{DDA18} (2)		I_{DDA33} (3)	
		TYP (4)	MAX	TYP (4)	MAX	TYP (4)	MAX	TYP (4)	MAX	TYP (4)	MAX
Operational (Flash) (5)	The following peripheral clocks are enabled: • ePWM1/2/3/4/5/6 • eCAP1/2/3/4 • eQEP1/2 • eCAN-A • SCI-A/B (FIFO mode) • SPI-A (FIFO mode) • ADC • I2C • CPU Timer 0/1/2 All PWM pins are toggled at 150 kHz. All I/O pins are left unconnected. (6)	205 mA		15 mA		35 mA	40 mA	30 mA	38 mA	1.5 mA	2 mA
IDLE	Flash is powered down. XCLKOUT is turned off. The following peripheral clocks are enabled: • eCAN-A • SCI-A • SPI-A • I2C	75 mA	90 mA	500 µA	2 mA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA
STANDBY	Flash is powered down. Peripheral clocks are off.	6 mA	12 mA	100 µA	500 µA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA
HALT	Flash is powered down. Peripheral clocks are off. Input clock is disabled.	70 µA		60 µA	120 µA	2 µA	10 µA	5 µA	50 µA	15 µA	30 µA

(1) I_{DDIO} current is dependent on the electrical loading on the I/O pins.
(2) I_{DDA18} includes current into V_{DD1A18} and V_{DD2A18} pins. In order to realize the I_{DDA18} currents shown for IDLE, STANDBY, and HALT, clock to the ADC module must be turned off explicitly by writing to the PCLKCR0 register.
(3) I_{DDA33} includes current into V_{DDA2} and V_{DDAIO} pins.
(4) The TYP numbers are applicable over room temperature and nominal voltage.
(5) When the identical code is run off SARAM, I_{DD} would increase as the code operates with zero wait states.
(6) The following is done in a loop:

- Data is continuously transmitted out of the SCI-A, SCI-B, SPI-A, McBSP-A, and eCAN-A ports.
- Floating-point multiplication and addition are performed.
- Watchdog is reset.
- ADC is performing continuous conversion. Data from ADC is transferred to SARAM through the DMA.
- 32-bit read/write of the XINTF is performed.
- GPIO19 is toggled.

6.4.1 Reducing Current Consumption

Like 280x and 281x, F2833x DSPs incorporate a unique method to reduce the device current consumption. Since each peripheral unit has an individual clock-enable bit, significant reduction in current consumption can be achieved by turning off the clock to any peripheral module that is not used in a given application. Furthermore, any one of the three low-power modes could be taken advantage of to reduce the current consumption even further. Table 6-4 indicates the typical reduction in current consumption achieved by turning off the clocks.

Table 6-4. Typical Current Consumption by Various Peripherals (at 150 MHz)(1)

PERIPHERAL MODULE	I_{DD} CURRENT REDUCTION (mA)
ADC	8(2)
I2C	2.5
eQEP	5
ePWM	5
eCAP	2
SCI	5
SPI	4
eCAN	8
McBSP	7
CPU - Timer	2
XINTF	10(3)
DMA	10
FPU	15

(1) All peripheral clocks are disabled upon reset. Writing to/reading from peripheral registers is possible only after the peripheral clocks are turned on.

(2) This number represents the current drawn by the digital portion of the ADC module. Turning off the clock to the ADC module results in the elimination of the current drawn by the analog portion of the ADC (I_{DDA18}) as well.

(3) Operating the XINTF bus has a significant effect on IDDIO current. It will increase considerably based on the following:

- How many address/data pins toggle from one cycle to another
- How fast they toggle
- Whether 16-bit or 32-bit interface is used and
- The load on these pins.

NOTE

I_{DDIO} current consumption is reduced by 15 mA (typical) when XCLKOUT is turned off.

NOTE

The baseline I_{DD} current (current when the core is executing a dummy loop with no peripherals enabled) is 165 mA, typical. To arrive at the I_{DD} current for a given application, the current-drawn by the peripherals (enabled by that application) must be added to the baseline I_{DD} current.

6.4.2 Current Consumption Graphs

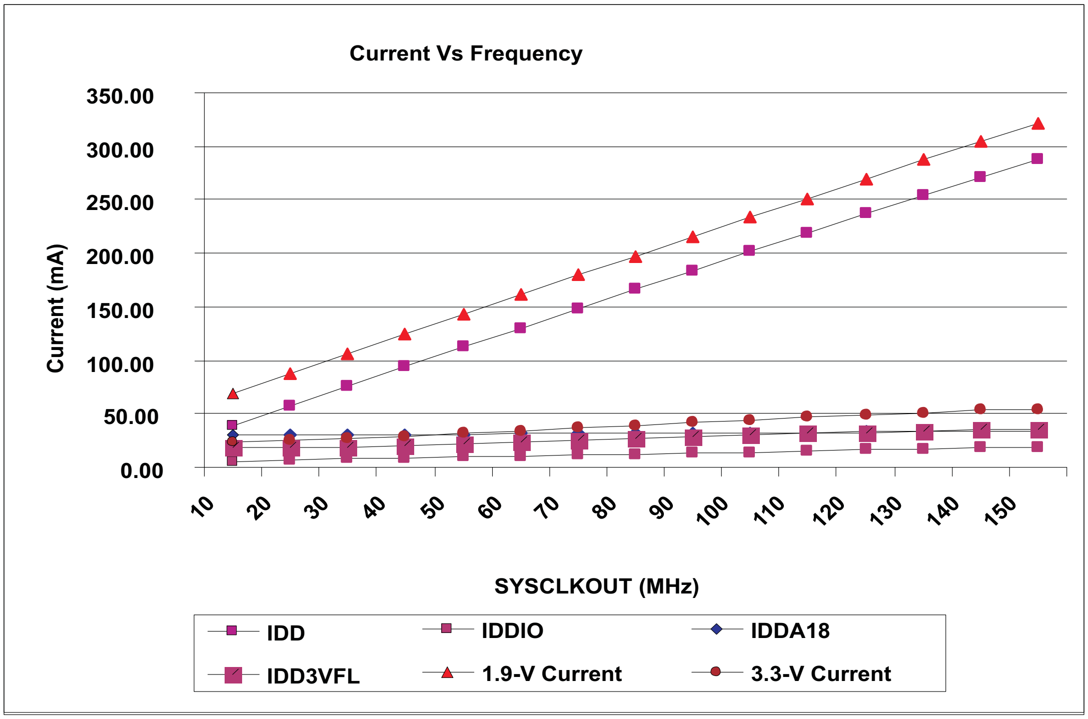


Figure 6-1. Typical Operational Current Versus Frequency (F28335/F28334)

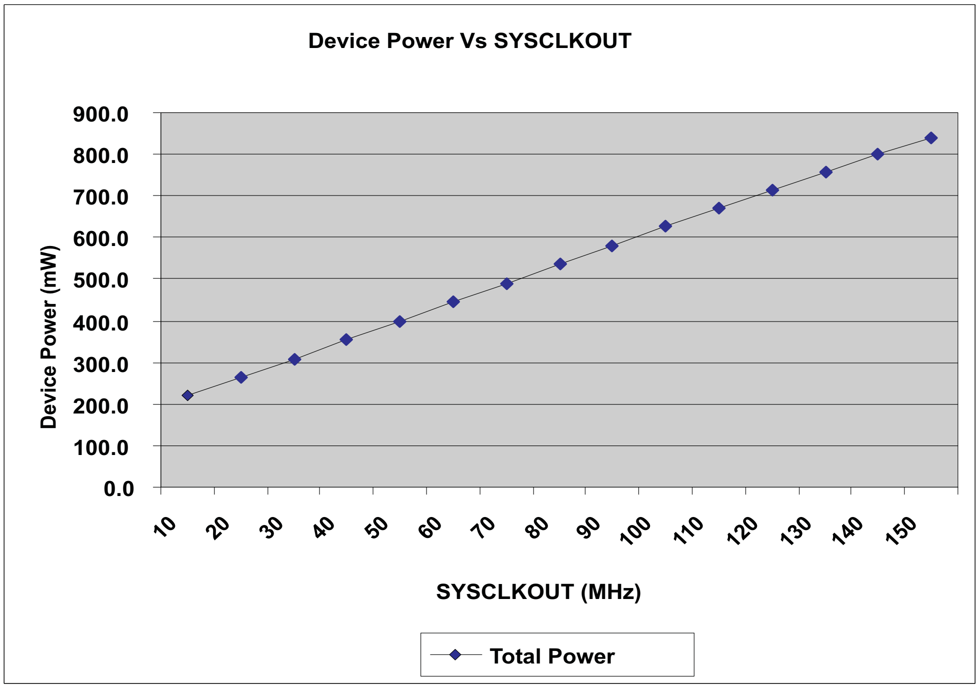


Figure 6-2. Typical Operational Power Versus Frequency (F28335/F28334)

NOTE

Typical operational current for 100-MHz devices can be estimated from Figure 6-1. For I_{dd} current alone, subtract the current contribution of non-existent peripherals after scaling the peripheral currents for 100 MHz. For example, to compute the current of F2833x-100 device, the contribution by the following peripherals must be subtracted from I_{dd} : eCAP5, eCAP6.

6.4.2.1 Thermal Design Considerations

Based on the end application design and operational profile, the I_{DD} and I_{DDIO} currents could vary. Systems with more than 1 Watt power dissipation may require a product level thermal design. Care should be taken to keep T_j within specified limits. In the end applications, T_{case} should be measured to estimate the operating junction temperature T_j. T_{case} is normally measured at the center of the package top side surface. The thermal application notes *IC Package Thermal Metrics* (literature number SPRA953) and *Reliability Data for TMS320LF24x and TMS320F281x Devices* (literature number SPRA963) help to understand the thermal metrics and definitions.

6.5 Emulator Connection Without Signal Buffering for the DSP

Figure 6-3 shows the connection between the DSP and JTAG header for a single-processor configuration. If the distance between the JTAG header and the DSP is greater than 6 inches, the emulation signals must be buffered. If the distance is less than 6 inches, buffering is typically not needed. Figure 6-3 shows the simpler, no-buffering situation. For the pullup/pulldown resistor values, see the pin description section. For details on buffering JTAG signals and multiple processor connections, see *TMS320F/C24x DSP Controllers CPU and Instruction Set Reference Guide* (literature number SPRU160).

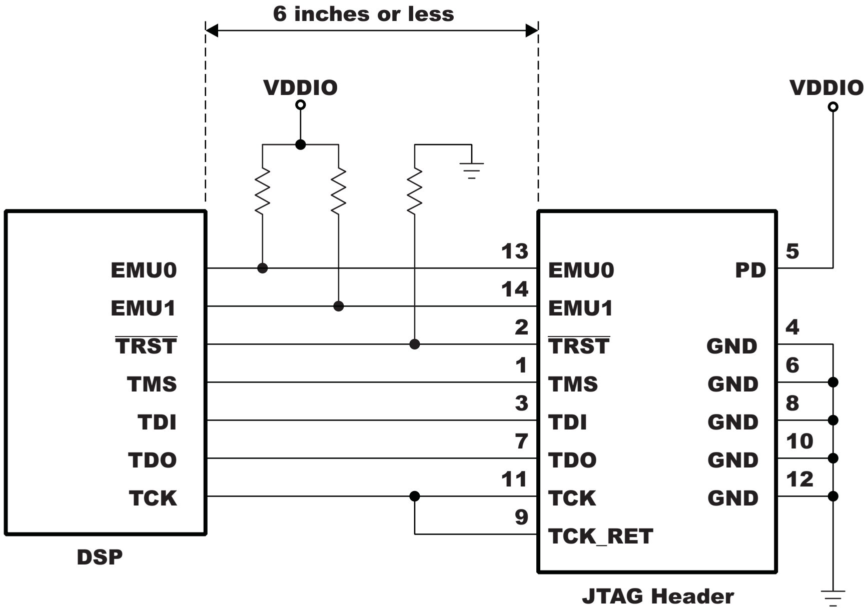


Figure 6-3. Emulator Connection Without Signal Buffering for the DSP

ADVANCE INFORMATION

6.6 Timing Parameter Symbology

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

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

Letters and symbols and their meanings:	
H	High
L	Low
V	Valid
X	Unknown, changing, or don't care level
Z	High impedance

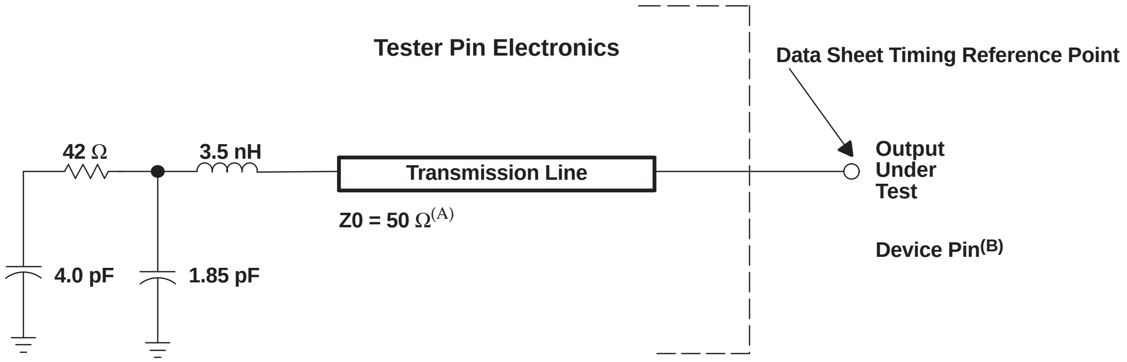
6.6.1 General Notes on Timing Parameters

All output signals from the 28x devices (including XCLKOUT) 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 document.

6.6.2 Test Load Circuit

This test load circuit is used to measure all switching characteristics provided in this document.



A. Input requirements in this data sheet are tested with an input slew rate of < 4 Volts per nanosecond (4 V/ns) at the device pin.

B. The data sheet provides timing at the device pin. For output timing analysis, the tester pin electronics and its transmission line effects must be taken into account. A transmission line with a delay of 2 ns or longer can be used to produce the desired transmission line effect. The transmission line is intended as a load only. It is not necessary to add or subtract the transmission line delay (2 ns or longer) from the data sheet timing.

Figure 6-4. 3.3-V Test Load Circuit

6.6.3 Device Clock Table

This section provides the timing requirements and switching characteristics for the various clock options available on the F2833x DSPs. Table 6-5 and Table 6-6 list the cycle times of various clocks.

Table 6-5. Clocking and Nomenclature (150-MHz devices)

		MIN	NOM	MAX	UNIT
On-chip oscillator clock	$t_{c(OSC)}$, Cycle time	28.6		50	ns
	Frequency	20		35	MHz
XCLKIN(1)	$t_{c(CI)}$, Cycle time	6.67		250	ns
	Frequency	4		150	MHz
SYSCLKOUT	$t_{c(SCO)}$, Cycle time	6.67		500	ns
	Frequency	2		150	MHz
XCLKOUT	$t_{c(XCO)}$, Cycle time	6.67		2000	ns
	Frequency	0.5		150	MHz
HSPCLK(2)	$t_{c(HCO)}$, Cycle time	6.67	13.3(3)		ns
	Frequency		75(3)	150	MHz
LSPCLK(2)	$t_{c(LCO)}$, Cycle time	13.3	26.7(3)		ns
	Frequency		37.5(3)	75	MHz
ADC clock	$t_{c(ADCCLK)}$, Cycle time	40			ns
	Frequency			25	MHz

(1) This also applies to the X1 pin if a 1.9-V oscillator is used.
(2) Lower LSPCLK and HSPCLK will reduce device power consumption.
(3) This is the default reset value if SYSCLKOUT = 150 MHz.

Table 6-6. Clocking and Nomenclature (100-MHz devices)

		MIN	NOM	MAX	UNIT
On-chip oscillator clock	$t_{c(OSC)}$, Cycle time	28.6		50	ns
	Frequency	20		35	MHz
XCLKIN(1)	$t_{c(CI)}$, Cycle time	10		250	ns
	Frequency	4		100	MHz
SYSCLKOUT	$t_{c(SCO)}$, Cycle time	10		500	ns
	Frequency	2		100	MHz
XCLKOUT	$t_{c(XCO)}$, Cycle time	10		2000	ns
	Frequency	0.5		100	MHz
HSPCLK(2)	$t_{c(HCO)}$, Cycle time	10	20(3)		ns
	Frequency		50(3)	100	MHz
LSPCLK(2)	$t_{c(LCO)}$, Cycle time	20	40(3)		ns
	Frequency		25(3)	50	MHz
ADC clock	$t_{c(ADCCLK)}$, Cycle time	40			ns
	Frequency			25	MHz

(1) This also applies to the X1 pin if a 1.9-V oscillator is used.
(2) Lower LSPCLK and HSPCLK will reduce device power consumption.
(3) This is the default reset value if SYSCLKOUT = 100 MHz.

6.7 Clock Requirements and Characteristics

Table 6-7. Input Clock Frequency

PARAMETER				MIN	TYP	MAX	UNIT
f_x	Input clock frequency	Resonator (X1/X2)		20		35	MHz
		Crystal (X1/X2)		20		35	
		External oscillator/clock source (XCLKIN or X1 pin)	150-MHz device	4		150	
			100-MHz device	4		100	
f_l	Limp mode SYSCLKOUT frequency range (with /2 enabled)				1 - 5		MHz

Table 6-8. XCLKIN(1) Timing Requirements - PLL Enabled

NO.			MIN	MAX	UNIT
C8	$t_{c(CI)}$	Cycle time, XCLKIN	33.3	200	ns
C9	$t_{f(CI)}$	Fall time, XCLKIN		6	ns
C10	$t_{r(CI)}$	Rise time, XCLKIN		6	ns
C11	$t_{w(CIL)}$	Pulse duration, XCLKIN low as a percentage of $t_{c(OSCCLK)}$	45	55	%
C12	$t_{w(CIH)}$	Pulse duration, XCLKIN high as a percentage of $t_{c(OSCCLK)}$	45	55	%

(1) This applies to the X1 pin also.

Table 6-9. XCLKIN(1) Timing Requirements - PLL Disabled

NO.				MIN	MAX	UNIT
C8	$t_{c(CI)}$	Cycle time, XCLKIN	150-MHz device	6.67	250	ns
			100-MHz device	10	250	
C9	$t_{f(CI)}$	Fall time, XCLKIN	Up to 30 MHz		6	ns
			30 MHz to 150 MHz		2	ns
C10	$t_{r(CI)}$	Rise time, XCLKIN	Up to 30 MHz		6	ns
			30 MHz to 150 MHz		2	ns
C11	$t_{w(CIL)}$	Pulse duration, XCLKIN low as a percentage of $t_{c(OSCCLK)}$		45	55	%
C12	$t_{w(CIH)}$	Pulse duration, XCLKIN high as a percentage of $t_{c(OSCCLK)}$		45	55	%

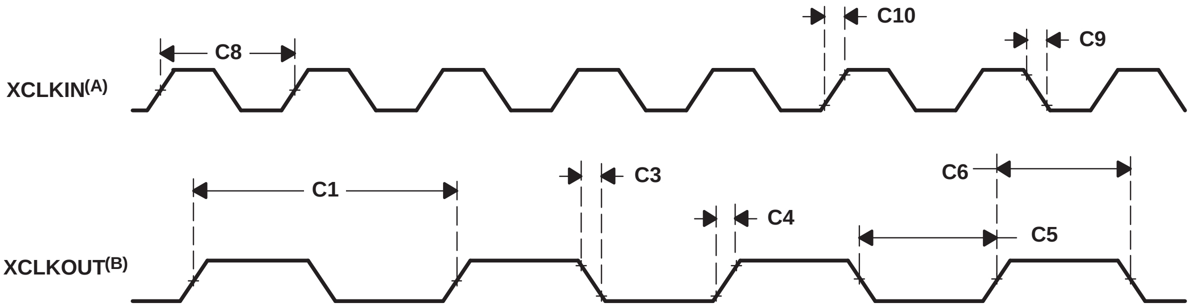
(1) This applies to the X1 pin also.

The possible configuration modes are shown in Table 3-18.

Table 6-10. XCLKOUT Switching Characteristics (PLL Bypassed or Enabled)(1)(2)

NO.	PARAMETER			MIN	TYP	MAX	UNIT
C1	$t_{c(XCO)}$	Cycle time, XCLKOUT	150-MHz device	6.67			ns
			100-MHz device	10			
C3	$t_{f(XCO)}$	Fall time, XCLKOUT			2		ns
C4	$t_{r(XCO)}$	Rise time, XCLKOUT			2		ns
C5	$t_{w(XCOL)}$	Pulse duration, XCLKOUT low		H – 2		H + 2	ns
C6	$t_{w(XCOH)}$	Pulse duration, XCLKOUT high		H – 2		H + 2	ns
	t_p	PLL lock time				$131072t_{c(OSCCLK)}$ (3)	cycles

(1) A load of 40 pF is assumed for these parameters.
(2) $H = 0.5t_{c(XCO)}$
(3) OSCCLK is either the output of the on-chip oscillator or the output from an external oscillator.



A. The relationship of XCLKIN to XCLKOUT depends on the divide factor chosen. The waveform relationship shown is intended to illustrate the timing parameters only and may differ based on actual configuration.

B. XCLKOUT configured to reflect SYSCLKOUT.

Figure 6-5. Clock Timing

6.8 Power Sequencing

No requirements are placed on the power up/down sequence of the various power pins to ensure the correct reset state for all the modules. However, if the 3.3-V transistors in the level shifting output buffers of the I/O pins are powered prior to the 1.9-V transistors, it is possible for the output buffers to turn on, causing a glitch to occur on the pin during power up. To avoid this behavior, power the V_{DD} pins prior to or simultaneously with the V_{DDIO} pins, ensuring that the V_{DD} pins have reached 0.7 V before the V_{DDIO} pins reach 0.7 V.

There are some requirements on the $\overline{XRS}$ pin:

1. During power up, the $\overline{XRS}$ pin must be held low for $t_{w(RSL1)}$ after the input clock is stable (see Table 6-12). This is to enable the entire device to start from a known condition.
2. During power down, the $\overline{XRS}$ pin must be pulled low at least 8 μs prior to V_{DD} reaching 1.5 V. This is to enhance flash reliability.

Additionally it is recommended that no voltage larger than a diode drop (0.7 V) should be applied to any pin prior to powering up the device. Voltages applied to pins on an unpowered device can bias internal p-n junctions in unintended ways and produce unpredictable results.

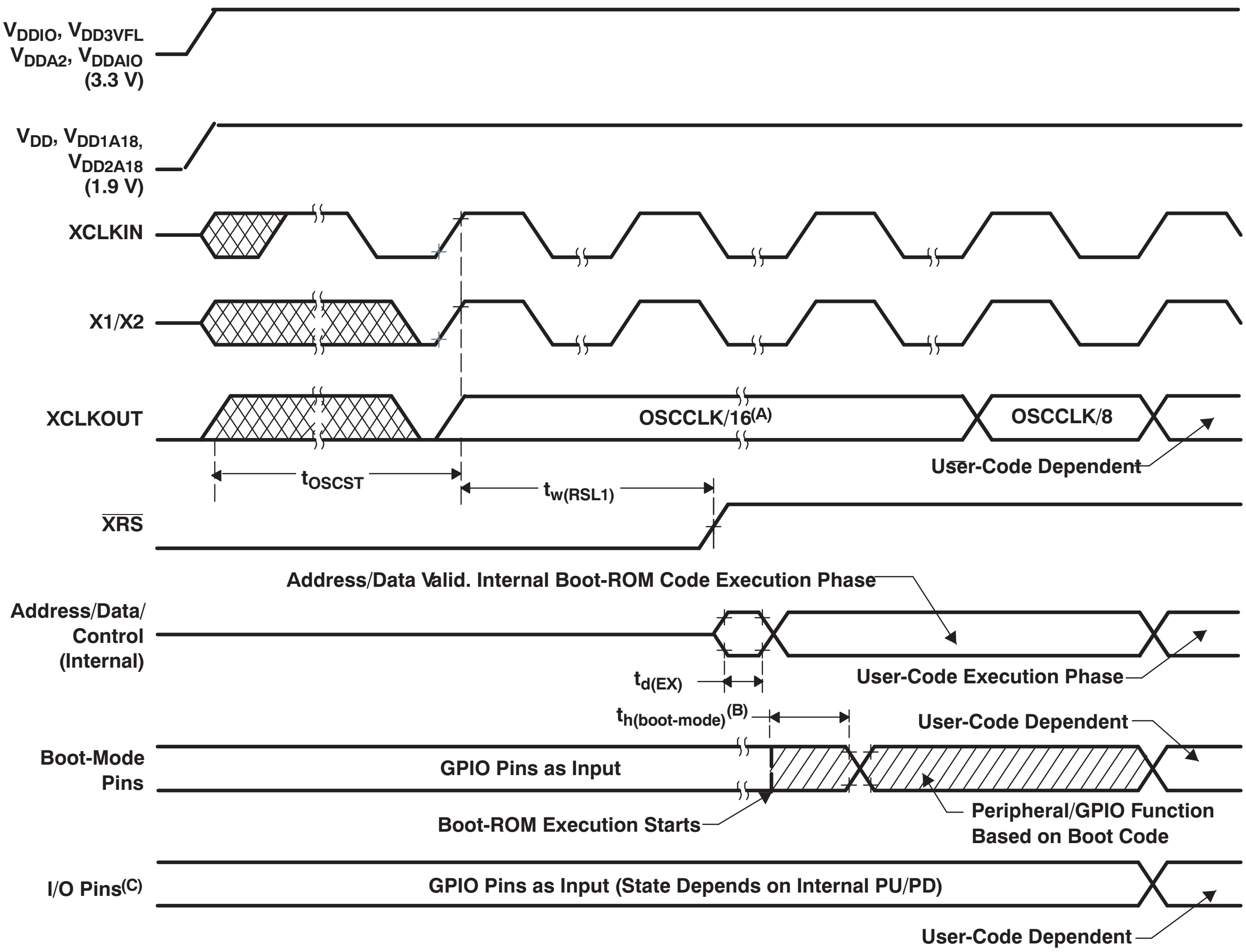
6.8.1 Power Management and Supervisory Circuit Solutions

Table 6-11 lists the power management and supervisory circuit solutions for 280x DSPs. LDO selection depends on the total power consumed in the end application. Go to www.power.ti.com for a complete list of TI power ICs or select TI DSP Power Solutions for links to the *DSP Power Selection Guide* (slub006a.pdf) and links to specific power reference designs.

Table 6-11. Power Management and Supervisory Circuit Solutions

SUPPLIER	TYPE	PART	DESCRIPTION
Texas Instruments	LDO	TPS767D301	Dual 1-A low-dropout regulator (LDO) with supply voltage supervisor (SVS)
Texas Instruments	LDO	TPS70202	Dual 500/250-mA LDO with SVS
Texas Instruments	LDO	TPS766xx	250-mA LDO with PG
Texas Instruments	SVS	TPS3808	Open Drain SVS with programmable delay
Texas Instruments	SVS	TPS3803	Low-cost Open-drain SVS with 5 μS delay
Texas Instruments	LDO	TPS799xx	200-mA LDO in WCSP package
Texas Instruments	LDO	TPS736xx	400-mA LDO with 40 mV of V_{DO}
Texas Instruments	DC/DC	TPS62110	High V_{in} 1.2-A dc/dc converter in 4x4 QFN package
Texas Instruments	DC/DC	TPS6230x	500-mA converter in WCSP package

ADVANCE INFORMATION



A. Upon power up, SYSCLKOUT is OSCCLK/4. Since both the XTIMCLK and CLKMODE bits in the XINTCNF2 register come up with a reset state of 1, SYSCLKOUT is further divided by 4 before it appears at XCLKOUT. This explains why XCLKOUT = OSCCLK/16 during this phase. Subsequently, boot ROM changes SYSCLKOUT to OSCCLK/2. Because the XTIMCLK register is unchanged by the boot ROM, XCLKOUT is OSCCLK/8 during this phase.

B. After reset, the boot ROM code samples Boot Mode pins. Based on the status of the Boot Mode pin, the boot code branches to destination memory or boot code function. If boot ROM code executes after power-on conditions (in debugger environment), the boot code execution time is based on the current SYSCLKOUT speed. The SYSCLKOUT will be based on user environment and could be with or without PLL enabled.

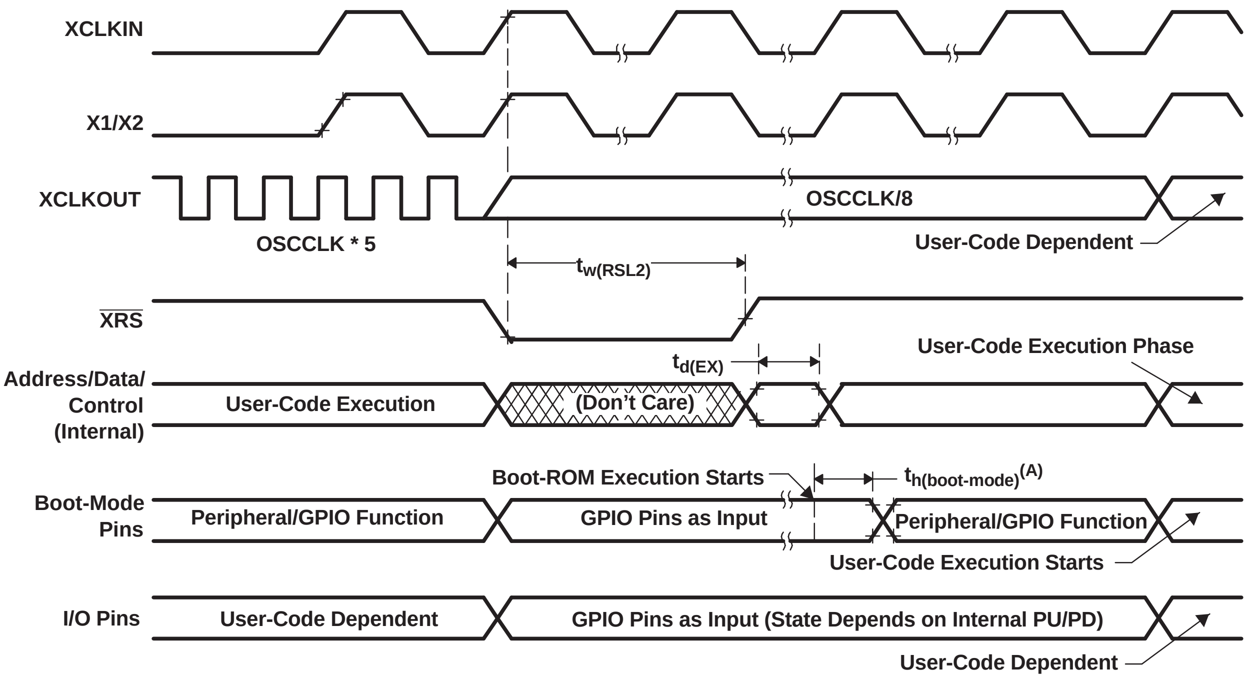
C. See Section 6.8 for requirements to ensure a high-impedance state for GPIO pins during power-up.

Figure 6-6. Power-on Reset

Table 6-12. Reset ($\overline{\text{XRS}}$) Timing Requirements

			MIN	NOM	MAX	UNIT
$t_{w(RSL1)}$ [1]	Pulse duration, stable XCLKIN to $\overline{\text{XRS}}$ high		$8t_{c(OSCCLK)}$			cycles
$t_{w(RSL2)}$	Pulse duration, $\overline{\text{XRS}}$ low	Warm reset	$8t_{c(OSCCLK)}$			cycles
$t_{w(WDRS)}$	Pulse duration, reset pulse generated by watchdog			$512t_{c(OSCCLK)}$		cycles
$t_{d(EX)}$	Delay time, address/data valid after $\overline{\text{XRS}}$ high			$32t_{c(OSCCLK)}$		cycles
t_{OSCST} [2]	Oscillator start-up time		1	10		ms
$t_{h(boot\text{-}mode)}$	Hold time for boot-mode pins		$200t_{c(OSCCLK)}$			cycles

(1) In addition to the $t_{w(RSL1)}$ requirement, $\overline{\text{XRS}}$ has to be low at least for 1 ms after V_{DD} reaches 1.5 V.
(2) Dependent on crystal/resonator and board design.



A. After reset, the Boot ROM code samples BOOT Mode pins. Based on the status of the Boot Mode pin, the boot code branches to destination memory or boot code function. If Boot ROM code executes after power-on conditions (in debugger environment), the Boot code execution time is based on the current SYSCLKOUT speed. The SYSCLKOUT will be based on user environment and could be with or without PLL enabled.

Figure 6-7. Warm Reset

Figure 6-8 shows an example for the effect of writing into PLLCR register. In the first phase, PLLCR = 0x0004 and SYSCLKOUT = OSCCLK x 2. The PLLCR is then written with 0x0008. Right after the PLLCR register is written, the PLL lock-up phase begins. During this phase, SYSCLKOUT = OSCCLK/2. After the PLL lock-up is complete (which takes 131072 OSCCLK cycles), SYSCLKOUT reflects the new operating frequency, OSCCLK x 4.

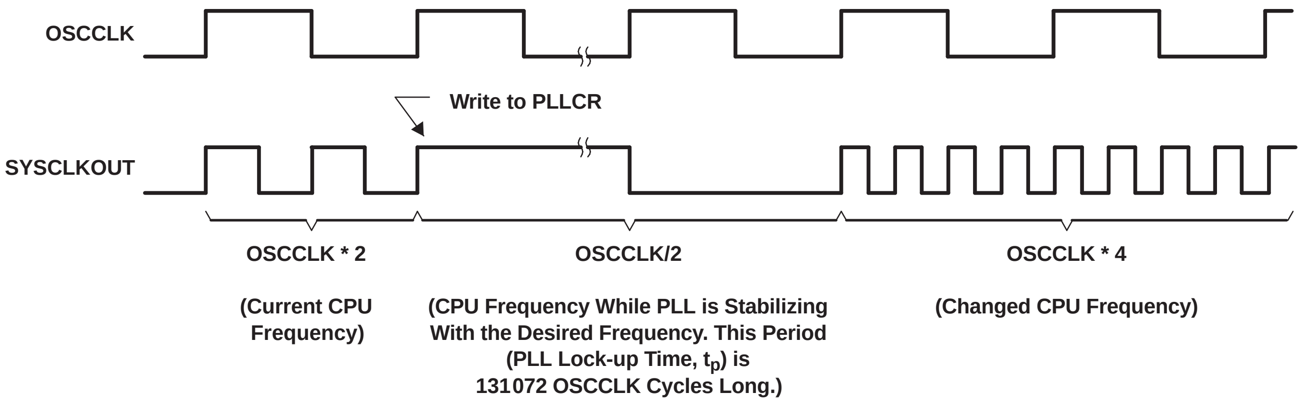


Figure 6-8. Example of Effect of Writing Into PLLCR Register

6.9 General-Purpose Input/Output (GPIO)

6.9.1 GPIO - Output Timing

Table 6-13. General-Purpose Output Switching Characteristics

PARAMETER			MIN	MAX	UNIT
$t_{r(GPO)}$	Rise time, GPIO switching low to high	All GPIOs		8	ns
$t_{f(GPO)}$	Fall time, GPIO switching high to low	All GPIOs		8	ns
t_{fGPO}	Toggling frequency, GPO pins			25	MHz

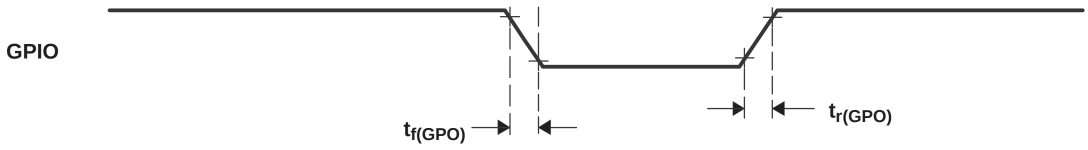
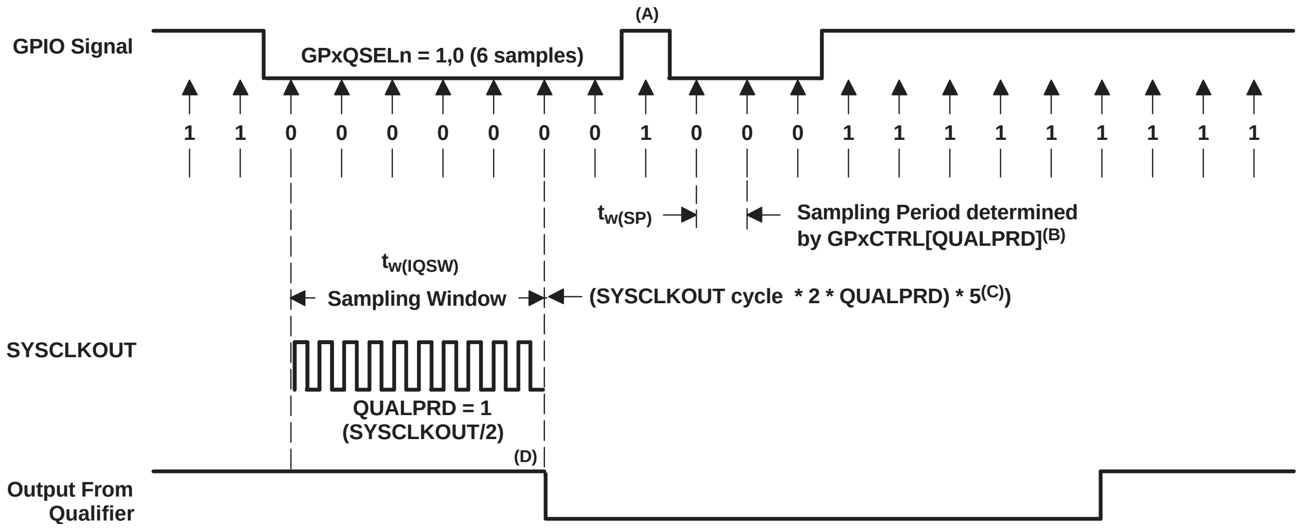


Figure 6-9. General-Purpose Output Timing

6.9.2 GPIO - Input Timing



A. This glitch will be ignored by the input qualifier. The QUALPRD bit field specifies the qualification sampling period. It can vary from 00 to 0xFF. If QUALPRD = 00, then the sampling period is 1 SYSCLKOUT cycle. For any other value "n", the qualification sampling period in 2n SYSCLKOUT cycles (i.e., at every 2n SYSCLKOUT cycles, the GPIO pin will be sampled).

B. The qualification period selected via the GPxCTRL register applies to groups of 8 GPIO pins.

C. The qualification block can take either three or six samples. The GPxQSELn Register selects which sample mode is used.

D. In the example shown, for the qualifier to detect the change, the input should be stable for 10 SYSCLKOUT cycles or greater. In other words, the inputs should be stable for (5 x QUALPRD x 2) SYSCLKOUT cycles. This would ensure 5 sampling periods for detection to occur. Since external signals are driven asynchronously, an 13-SYSCLKOUT-wide pulse ensures reliable recognition.

Figure 6-10. Sampling Mode

Table 6-14. General-Purpose Input Timing Requirements

			MIN	MAX	UNIT
$t_{w(SP)}$	Sampling period	QUALPRD = 0	$1t_{c(SCO)}$		cycles
		QUALPRD ≠ 0	$2t_{c(SCO)}$ * QUALPRD		cycles
$t_{w(IQSW)}$	Input qualifier sampling window		$t_{w(SP)}$ * (n[1] – 1)		cycles
$t_{w(GPI)}$ [2]	Pulse duration, GPIO low/high	Synchronous mode	$2t_{c(SCO)}$		cycles
		With input qualifier	$t_{w(IQSW)} + t_{w(SP)} + 1t_{c(SCO)}$		cycles

(1) "n" represents the number of qualification samples as defined by GPxQSELn register.
(2) For $t_{w(GPI)}$, pulse width is measured from V_{IL} to V_{IL} for an active low signal and V_{IH} to V_{IH} for an active high signal.

6.9.3 Sampling Window Width for Input Signals

The following section summarizes the sampling window width for input signals for various input qualifier configurations.

Sampling frequency denotes how often a signal is sampled with respect to SYSCLKOUT.

Sampling frequency = SYSCLKOUT/(2 * QUALPRD), if QUALPRD ≠ 0

Sampling frequency = SYSCLKOUT, if QUALPRD = 0

Sampling period = SYSCLKOUT cycle x 2 x QUALPRD, if QUALPRD ≠ 0

In the above equations, SYSCLKOUT cycle indicates the time period of SYSCLKOUT.

Sampling period = SYSCLKOUT cycle, if QUALPRD = 0

In a given sampling window, either 3 or 6 samples of the input signal are taken to determine the validity of the signal. This is determined by the value written to GPxQSELn register.

Case 1:

Qualification using 3 samples

Sampling window width = (SYSCLKOUT cycle x 2 x QUALPRD) x 2, if QUALPRD ≠ 0

Sampling window width = (SYSCLKOUT cycle) x 2, if QUALPRD = 0

Case 2:

Qualification using 6 samples

Sampling window width = (SYSCLKOUT cycle x 2 x QUALPRD) x 5, if QUALPRD ≠ 0

Sampling window width = (SYSCLKOUT cycle) x 5, if QUALPRD = 0

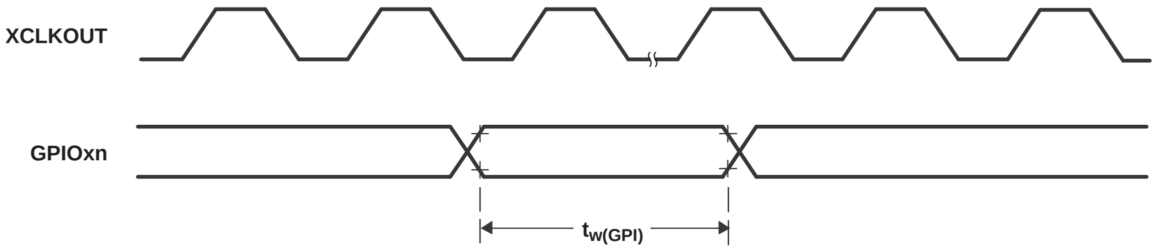


Figure 6-11. General-Purpose Input Timing

NOTE

The pulse-width requirement for general-purpose input is applicable for the XINT2_ADCSOC signal as well.

6.9.4 Low-Power Mode Wakeup Timing

Table 6-15 shows the timing requirements, Table 6-16 shows the switching characteristics, and Figure 6-12 shows the timing diagram for IDLE mode.

Table 6-15. IDLE Mode Timing Requirements(1)

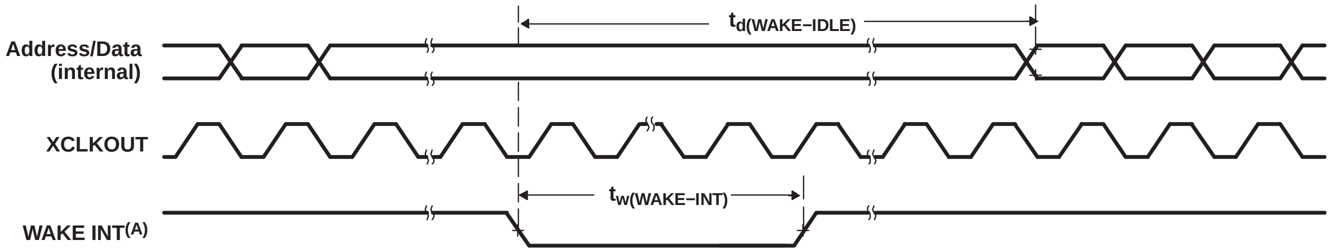
			MIN	NOM	MAX	UNIT
$t_{w(WAKE-INT)}$	Pulse duration, external wake-up signal	Without input qualifier	$2t_{c(SCO)}$			cycles
		With input qualifier	$5t_{c(SCO)} + t_{w(IQSW)}$			

(1) For an explanation of the input qualifier parameters, see Table 6-14.

Table 6-16. IDLE Mode Switching Characteristics(1)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{d(WAKE-IDLE)}$	Delay time, external wake signal to program execution resume (2)					
	• Wake-up from Flash – Flash module in active state	Without input qualifier			$20t_{c(SCO)}$	cycles
		With input qualifier			$20t_{c(SCO)} + t_{w(IQSW)}$	
	• Wake-up from Flash – Flash module in sleep state	Without input qualifier			$1050t_{c(SCO)}$	cycles
		With input qualifier			$1050t_{c(SCO)} + t_{w(IQSW)}$	
	• Wake-up from SARAM	Without input qualifier			$20t_{c(SCO)}$	cycles
		With input qualifier			$20t_{c(SCO)} + t_{w(IQSW)}$	

(1) For an explanation of the input qualifier parameters, see Table 6-14.
(2) This is the time taken to begin execution of the instruction that immediately follows the IDLE instruction. execution of an ISR (triggered by the wake up) signal involves additional latency.



A. WAKE INT can be any enabled interrupt, $\overline{\text{WDINT}}$, XNMI, or $\overline{\text{XRS}}$.

Figure 6-12. IDLE Entry and Exit Timing

Table 6-17. STANDBY Mode Timing Requirements

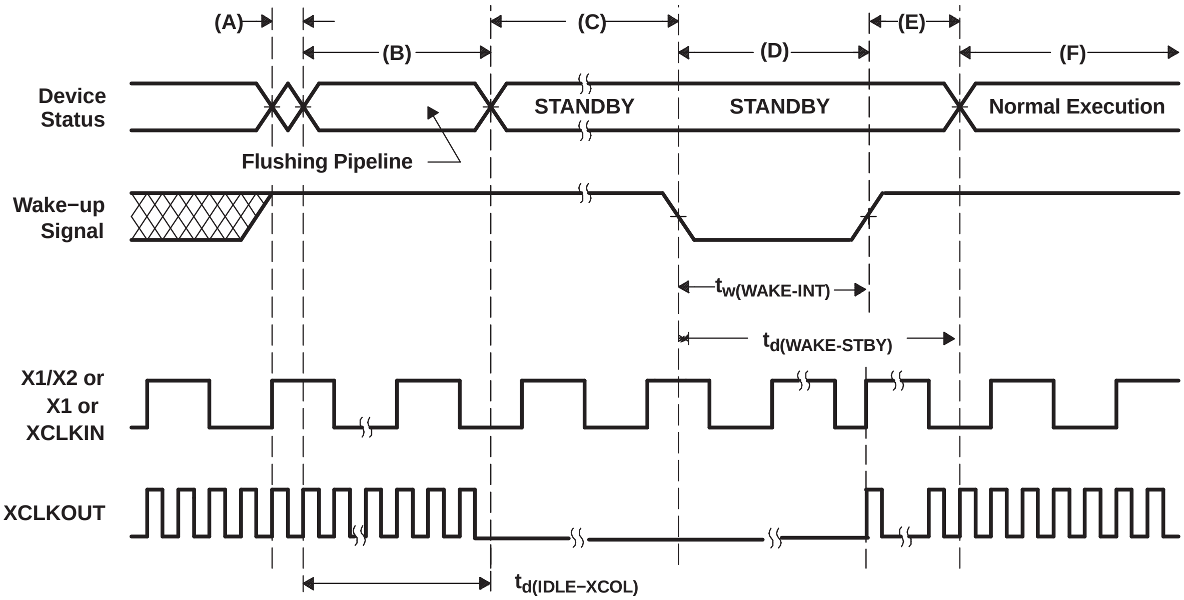
		TEST CONDITIONS	MIN	NOM	MAX	UNIT
$t_{w(WAKE-INT)}$	Pulse duration, external wake-up signal	Without input qualification	$3t_{c(OSCCLK)}$			cycles
		With input qualification(1)	$(2 + \text{QUALSTDBY}) * t_{c(OSCCLK)}$			

(1) QUALSTDBY is a 6-bit field in the LPMCR0 register.

Table 6-18. STANDBY Mode Switching Characteristics

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$t_{d(IDLE-XCOL)}$	Delay time, IDLE instruction executed to XCLKOUT low		$32t_{c(SCO)}$		$45t_{c(SCO)}$	cycles
$t_{d(WAKE-STBY)}$	Delay time, external wake signal to program execution resume[1]					cycles
	• Wake up from flash – Flash module in active state	Without input qualifier			$100t_{c(SCO)}$	cycles
		With input qualifier			$100t_{c(SCO)} + t_{w(WAKE-INT)}$	
	• Wake up from flash – Flash module in sleep state	Without input qualifier			$1125t_{c(SCO)}$	cycles
		With input qualifier			$1125t_{c(SCO)} + t_{w(WAKE-INT)}$	
	• Wake up from SARAM	Without input qualifier			$100t_{c(SCO)}$	cycles
		With input qualifier			$100t_{c(SCO)} + t_{w(WAKE-INT)}$	

(1) This is the time taken to begin execution of the instruction that immediately follows the IDLE instruction. execution of an ISR (triggered by the wake up signal) involves additional latency.



A. IDLE instruction is executed to put the device into STANDBY mode.

B. The PLL block responds to the STANDBY signal. SYSCLKOUT is held for approximately 32 cycles before being turned off. This 32-cycle delay enables the CPU pipe and any other pending operations to flush properly.

C. Clock to the peripherals are turned off. However, the PLL and watchdog are not shut down. The device is now in STANDBY mode.

D. The external wake-up signal is driven active.

E. After a latency period, the STANDBY mode is exited.

F. Normal execution resumes. The device will respond to the interrupt (if enabled).

Figure 6-13. STANDBY Entry and Exit Timing Diagram

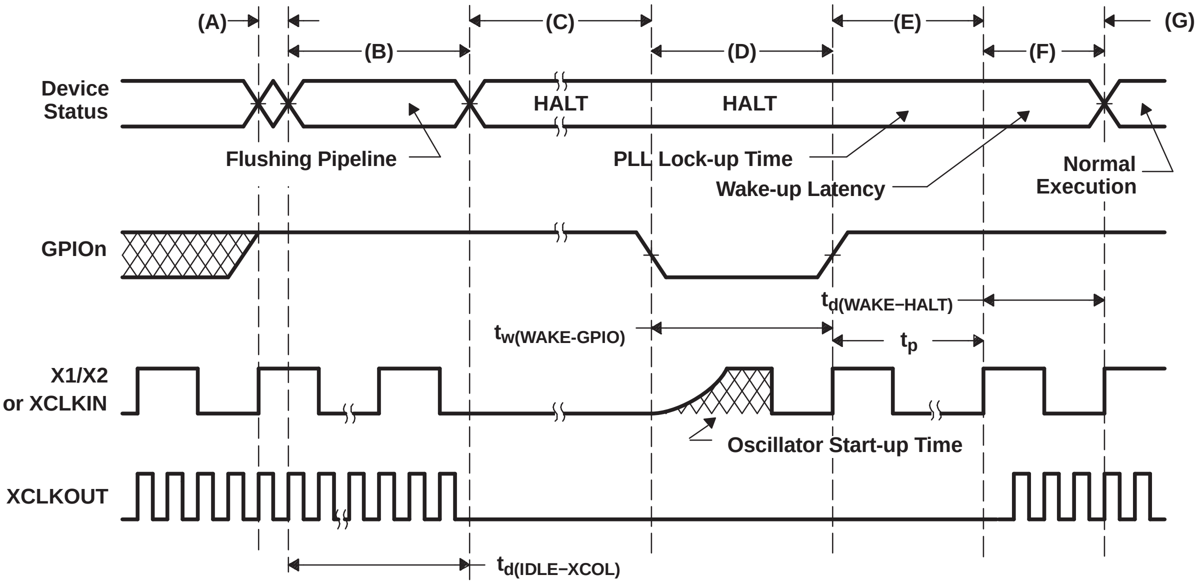
Table 6-19. HALT Mode Timing Requirements

		MIN	NOM	MAX	UNIT
$t_{w(WAKE-GPIO)}$	Pulse duration, GPIO wake-up signal	$t_{oscst} + 2t_{c(OSCCLK)}$[1]			cycles
$t_{w(WAKE-XRS)}$	Pulse duration, $\overline{XRS}$ wakeup signal	$t_{oscst} + 8t_{c(OSCCLK)}$			cycles

(1) See Table 6-12 for an explanation of t_{oscst}.

Table 6-20. HALT Mode Switching Characteristics

PARAMETER		MIN	TYP	MAX	UNIT
$t_{d(IDLE\text{-}XCOL)}$	Delay time, IDLE instruction executed to XCLKOUT low	$32t_{c(SCO)}$		$45t_{c(SCO)}$	cycles
t_p	PLL lock-up time			$131072t_{c(OSCCLK)}$	cycles
$t_{d(WAKE\text{-}HALT)}$	Delay time, PLL lock to program execution resume • Wake up from flash – Flash module in sleep state			$1125t_{c(SCO)}$	cycles
	• Wake up from SARAM			$35t_{c(SCO)}$	cycles



A. IDLE instruction is executed to put the device into HALT mode.

B. The PLL block responds to the HALT signal. SYSCLKOUT is held for approximately 32 cycles before the oscillator is turned off and the CLKIN to the core is stopped. This 32-cycle delay enables the CPU pipe and any other pending operations to flush properly.

C. Clocks to the peripherals are turned off and the PLL is shut down. If a quartz crystal or ceramic resonator is used as the clock source, the internal oscillator is shut down as well. The device is now in HALT mode and consumes absolute minimum power.

D. When the GPIOn pin is driven low, the oscillator is turned on and the oscillator wake-up sequence is initiated. The GPIO pin should be driven high only after the oscillator has stabilized. This enables the provision of a clean clock signal during the PLL lock sequence. Since the falling edge of the GPIO pin asynchronously begins the wakeup procedure, care should be taken to maintain a low noise environment prior to entering and during HALT mode.

E. When GPIOn is deactivated, it initiates the PLL lock sequence, which takes 131,072 OSCCLK (X1/X2 or X1 or XCLKIN) cycles.

F. When CLKIN to the core is enabled, the device will respond to the interrupt (if enabled), after a latency. The HALT mode is now exited.

G. Normal operation resumes.

Figure 6-14. HALT Wake-Up Using GPIOn

ADVANCE INFORMATION

6.10 Enhanced Control Peripherals

6.10.1 Enhanced Pulse Width Modulator (ePWM) Timing

PWM refers to PWM outputs on ePWM1-6. Table 6-21 shows the PWM timing requirements and Table 6-22, switching characteristics.

Table 6-21. ePWM Timing Requirements[1]

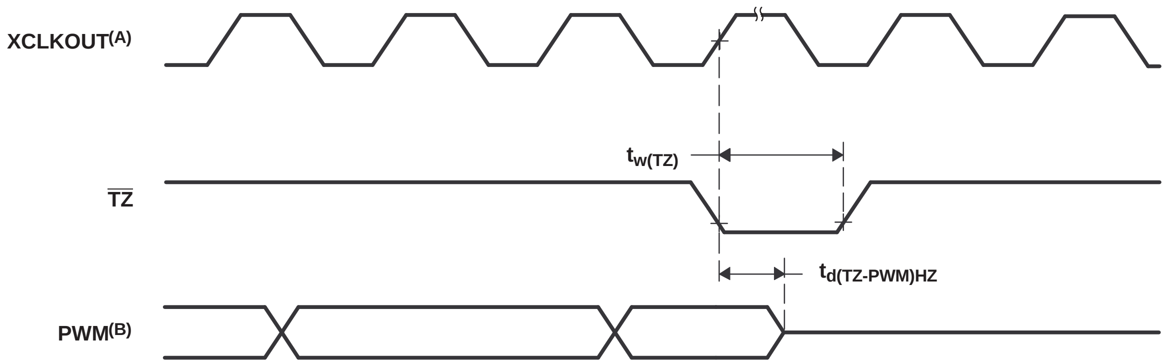
		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(SYCIN)}$	Sync input pulse width	Asynchronous	$2t_{c(SCO)}$		cycles
		Synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

[1] For an explanation of the input qualifier parameters, see Table 6-14.

Table 6-22. ePWM Switching Characteristics

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(PWM)}$	Pulse duration, PWMx output high/low		20		ns
$t_{w(SYNCOUT)}$	Sync output pulse width		$8t_{c(SCO)}$		cycles
$t_{d(PWM)tza}$	Delay time, trip input active to PWM forced high Delay time, trip input active to PWM forced low	no pin load		25	ns
$t_{d(TZ-PWM)HZ}$	Delay time, trip input active to PWM Hi-Z			20	ns

6.10.2 Trip-Zone Input Timing



A. $\overline{TZ}$ - $\overline{TZ1}$, $\overline{TZ2}$, $\overline{TZ3}$, $\overline{TZ4}$, $\overline{TZ5}$, $\overline{TZ6}$

B. PWM refers to all the PWM pins in the device. The state of the PWM pins after $\overline{TZ}$ is taken high depends on the PWM recovery software.

Figure 6-15. PWM Hi-Z Characteristics

Table 6-23. Trip-Zone input Timing Requirements[1]

			MIN	MAX	UNIT
$t_{w(TZ)}$	Pulse duration, $\overline{TZx}$ input low	Asynchronous	$1t_{c(SCO)}$		cycles
		Synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

[1] For an explanation of the input qualifier parameters, see Table 6-14.

Table 6-24 shows the high-resolution PWM switching characteristics.

Table 6-24. High Resolution PWM Characteristics at SYSCLKOUT = (60 - 150 MHz)

	MIN	TYP	MAX	UNIT
Micro Edge Positioning (MEP) step size[1]		150	310	ps

(1) Maximum MEP step size is based on worst-case process, maximum temperature and maximum voltage. MEP step size will increase with low voltage and high temperature and decrease with voltage and cold temperature.
Applications that use the HRPWM feature should use MEP Scale Factor Optimizer (SFO) estimation software functions. See the TI software libraries for details of using SFO function in end applications. SFO functions help to estimate the number of MEP steps per SYSCLKOUT period dynamically while the HRPWM is in operation.

Table 6-25 shows the eCAP timing requirement and Table 6-26 shows the eCAP switching characteristics.

Table 6-25. Enhanced Capture (eCAP) Timing Requirement[1]

		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(CAP)}$	Capture input pulse width	Asynchronous	$2t_{c(SCO)}$		cycles
		Synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

(1) For an explanation of the input qualifier parameters, see Table 6-14.

Table 6-26. eCAP Switching Characteristics

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(APWM)}$	Pulse duration, APWMx output high/low		20		ns

Table 6-27 shows the eQEP timing requirement and Table 6-28 shows the eQEP switching characteristics.

Table 6-27. Enhanced Quadrature Encoder Pulse (eQEP) Timing Requirements[1]

		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(QEPP)}$	QEP input period	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2(1t_{c(SCO)} + t_{w(IQSW)})$		cycles
$t_{w(INDEXH)}$	QEP Index Input High time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles
$t_{w(INDEXL)}$	QEP Index Input Low time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles
$t_{w(STROBH)}$	QEP Strobe High time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles
$t_{w(STROBL)}$	QEP Strobe Input Low time	Asynchronous/synchronous	$2t_{c(SCO)}$		cycles
		With input qualifier	$2t_{c(SCO)} + t_{w(IQSW)}$		cycles

(1) For an explanation of the input qualifier parameters, see Table 6-14.

Table 6-28. eQEP Switching Characteristics

PARAMETER		TEST CONDITIONS	MIN	MAX	UNIT
$t_{d(CNTR)xin}$	Delay time, external clock to counter increment			$4t_{c(SCO)}$	cycles
$t_{d(PCS-OUT)QEP}$	Delay time, QEP input edge to position compare sync output			$6t_{c(SCO)}$	cycles

Table 6-29. External ADC Start-of-Conversion Switching Characteristics

PARAMETER		MIN	MAX	UNIT
$t_{w(ADCSOCAL)}$	Pulse duration, $\overline{\text{ADCSOCAO}}$ low	$32t_{c(HCO)}$		cycles

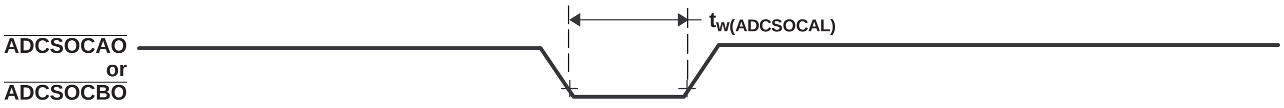


Figure 6-16. $\overline{\text{ADCSOCAO}}$ or $\overline{\text{ADCSOCBO}}$ Timing

6.10.3 External Interrupt Timing

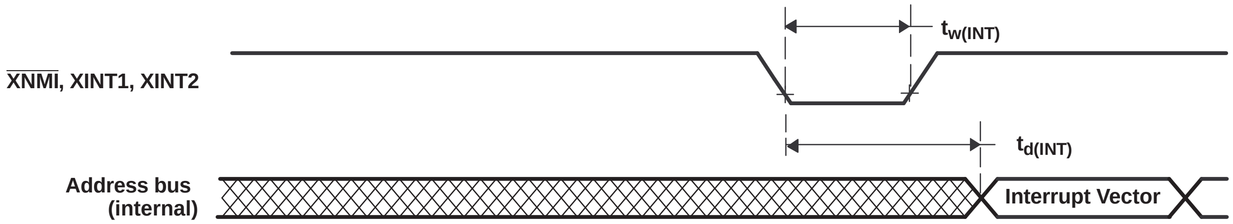


Figure 6-17. External Interrupt Timing

Table 6-30. External Interrupt Timing Requirements(1)

		TEST CONDITIONS	MIN	MAX	UNIT
$t_{w(INT)}$(2)	Pulse duration, INT input low/high	Synchronous	$1t_{c(SCO)}$		cycles
		With qualifier	$1t_{c(SCO)} + t_{w(IQSW)}$		cycles

(1) For an explanation of the input qualifier parameters, see Table 6-14.
(2) This timing is applicable to any GPIO pin configured for ADCSOC functionality.

Table 6-31. External Interrupt Switching Characteristics(1)

PARAMETER		MIN	MAX	UNIT
$t_{d(INT)}$	Delay time, INT low/high to interrupt-vector fetch		$t_{w(IQSW)} + 12t_{c(SCO)}$	cycles

(1) For an explanation of the input qualifier parameters, see Table 6-14.

6.10.4 I2C Electrical Specification and Timing

Table 6-32. I2C Timing

		TEST CONDITIONS	MIN	MAX	UNIT
f_{SCL}	SCL clock frequency	I2C clock module frequency is between 7 MHz and 12 MHz and I2C prescaler and clock divider registers are configured appropriately		400	kHz
v_{il}	Low level input voltage			0.3 V_{DDIO}	V
V_{ih}	High level input voltage		0.7 V_{DDIO}		V
V_{hys}	Input hysteresis		0.05 V_{DDIO}		V
V_{ol}	Low level output voltage	3-mA sink current	0	0.4	V
t_{LOW}	Low period of SCL clock	I2C clock module frequency is between 7 MHz and 12 MHz and I2C prescaler and clock divider registers are configured appropriately	1.3		µs
t_{HIGH}	High period of SCL clock	I2C clock module frequency is between 7 MHz and 12 MHz and I2C prescaler and clock divider registers are configured appropriately	0.6		µs
I_I	Input current with an input voltage between 0.1 V_{DDIO} and 0.9 V_{DDIO} MAX		-10	10	µA

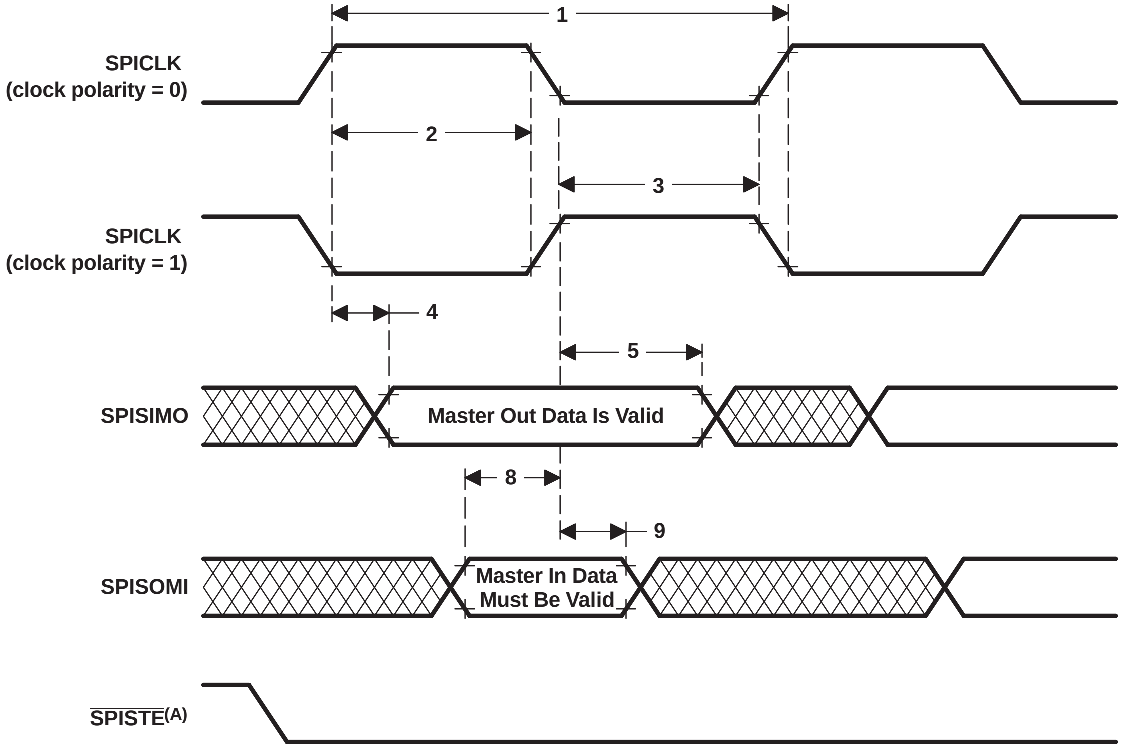
6.10.5 Serial Peripheral Interface (SPI) Master Mode Timing

Table 6-33 lists the master mode timing (clock phase = 0) and Table 6-34 lists the timing (clock phase = 1). Figure 6-18 and Figure 6-19 show the timing waveforms.

Table 6-33. SPI Master Mode External Timing (Clock Phase = 0)(1)(2)(3)(4)(5)

NO.			SPI WHEN (SPIBRR + 1) IS EVEN OR SPIBRR = 0 OR 2		SPI WHEN (SPIBRR + 1) IS ODD AND SPIBRR > 3		UNIT
			MIN	MAX	MIN	MAX	
1	$t_{c(SPC)M}$	Cycle time, SPICLK	$4t_{c(LCO)}$	$128t_{c(LCO)}$	$5t_{c(LCO)}$	$127t_{c(LCO)}$	ns
2	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}$ -10	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M} - 0.5t_{c(LCO)} - 10$	$0.5t_{c(SPC)}M - 0.5t_{c(LCO)}$	ns
	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}$ - 10	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M} - 0.5t_{c(LCO)} - 10$	$0.5t_{c(SPC)M} - 0.5t_{c(LCO)}$	
3	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)M}$ - 10	$0.5_{tc(SPC)M}$	$0.5t_{c(SPC)M} + 0.5t_{c(LCO)}$-10	$0.5t_{c(SPC)M} + 0.5t_{c(LCO)}$	ns
	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 1)	$0.5_{tc(SPC)M}$ - 10	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M} + 0.5t_{c(LCO)}$- 10	$0.5t_{c(SPC)M} + 0.5t_{c(LCO)}$	
4	$t_{d(SPCH-SIMO)M}$	Delay time, SPICLK high to SPISIMO valid (clock polarity = 0)		10		10	ns
	$t_{d(SPCL-SIMO)M}$	Delay time, SPICLK low to SPISIMO valid (clock polarity = 1)		10		10	
5	$t_{v(SPCL-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M} + 0.5t_{c(LCO)}$ -10		
	$t_{v(SPCH-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M} + 0.5t_{c(LCO)}$ -10		
8	$t_{su(SOMI-SPCL)M}$	Setup time, SPISOMI before SPICLK low (clock polarity = 0)	35		35		ns
	$t_{su(SOMI-SPCH)M}$	Setup time, SPISOMI before SPICLK high (clock polarity = 1)	35		35		ns
9	$t_{v(SPCL-SOMI)M}$	Valid time, SPISOMI data valid after SPICLK low (clock polarity = 0)	$0.25t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M}- 0.5t_{c(LCO)}$- 10		
	$t_{v(SPCH-SOMI)M}$	Valid time, SPISOMI data valid after SPICLK high (clock polarity = 1)	$0.25t_{c(SPC)M}$ - 10		$0.5t_{c(SPC)M}- 0.5t_{c(LCO)}$- 10		ns

(1) The MASTER / SLAVE bit (SPICTL.2) is set and the CLOCK PHASE bit (SPICTL.3) is cleared.
(2) $t_{c(SPC)}$ = SPI clock cycle time = LSPCLK/4 or LSPCLK/(SPIBRR +1)
(3) $t_{c(LCO)}$ = LSPCLK cycle time
(4) Internal clock prescalers must be adjusted such that the SPI clock speed is limited to the following SPI clock rate:
Master mode transmit 25-MHz MAX, master mode receive 12.5-MHz MAX
Slave mode transmit 12.5-MAX, slave mode receive 12.5-MHz MAX.
(5) The active edge of the SPICLK signal referenced is controlled by the clock polarity bit (SPICCR.6).



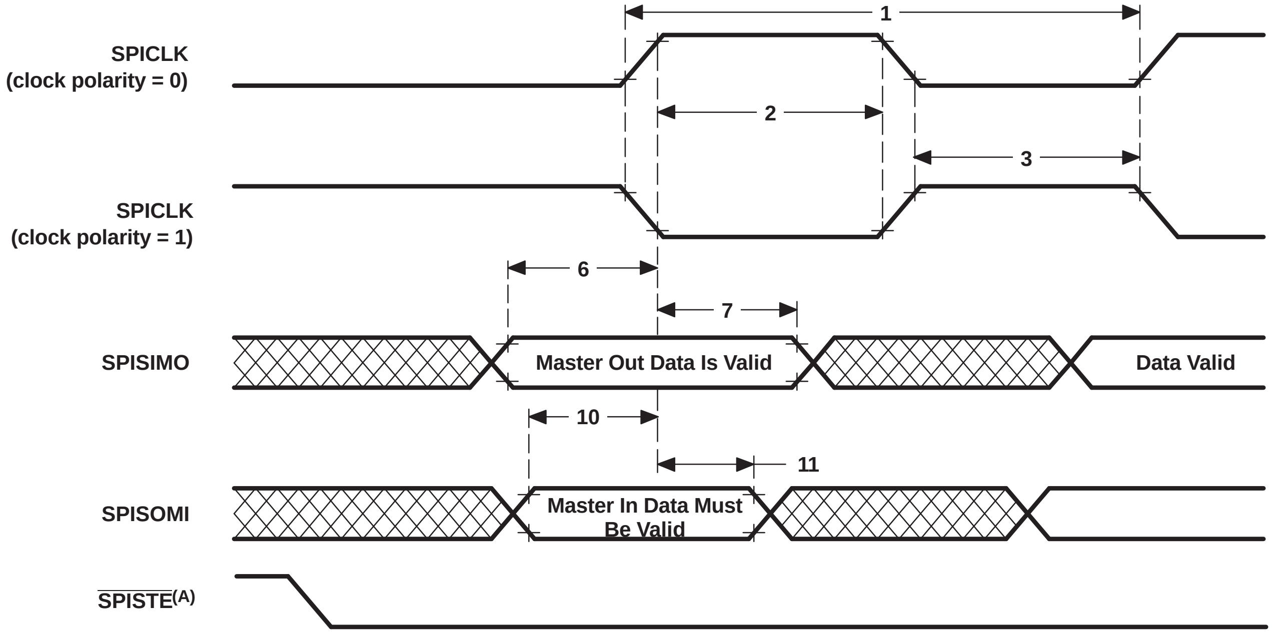
A. In the master mode, $\overline{\text{SPISTE}}$ goes active 0.5$t_{c(SPC)}$ (minimum) before valid SPI clock edge. On the trailing end of the word, the $\overline{\text{SPISTE}}$ will go inactive 0.5$t_{c(SPC)}$ after the receiving edge (SPICLK) of the last data bit, except that $\overline{\text{SPISTE}}$ stays active between back-to-back transmit words in both FIFO and nonFIFO modes.

Figure 6-18. SPI Master Mode External Timing (Clock Phase = 0)

Table 6-34. SPI Master Mode External Timing (Clock Phase = 1)(1)(2)(3)(4)(5)

NO.			SPI WHEN (SPIBRR + 1) IS EVEN OR SPIBRR = 0 OR 2		SPI WHEN (SPIBRR + 1) IS ODD AND SPIBRR > 3		UNIT
			MIN	MAX	MIN	MAX	
1	$t_{c(SPC)M}$	Cycle time, SPICLK	$4t_{c(LCO)}$	$128t_{c(LCO)}$	$5t_{c(LCO)}$	$127t_{c(LCO)}$	ns
2	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}$ -10	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}$ - $0.5t_{c(LCO)}$-10	$0.5t_{c(SPC)M}$ - $0.5t_{c(LCO)}$	ns
	$t_{w(SPCL))M}$	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}$ -10	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}$ - $0.5t_{c(LCO)}$-10	$0.5t_{c(SPC)M}$ - $0.5t_{c(LCO}$	ns
3	$t_{w(SPCL)M}$	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)M}$ -10	$0.5t_{c(SPC)M}$	$0.5t_{c(SPC)M}$ + $0.5t_{c(LCO)}$ - 10	$0.5t_{c(SPC)M}$ + $0.5t_{c(LCO)}$	ns
	$t_{w(SPCH)M}$	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)M}$ -10	$0.5t_{c(SPC)M}$	$0.5_{tc(SPC)M}$ + $0.5t_{c(LCO)}$ -10	$0.5t_{c(SPC)M}$ + $0.5t_{c(LCO)}$	ns
6	$t_{su(SIMO-SPCH)M}$	Setup time, SPISIMO data valid before SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M}$ - 10		ns
	$t_{su(SIMO-SPCL)M}$	Setup time, SPISIMO data valid before SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M}$ - 10		ns
7	$t_{v(SPCH-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M}$ - 10		ns
	$t_{v(SPCL-SIMO)M}$	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M}$ -10		ns
10	$t_{su(SOMI-SPCH)M}$	Setup time, SPISOMI before SPICLK high (clock polarity = 0)	35		35		ns
	$t_{su(SOMI-SPCL)M}$	Setup time, SPISOMI before SPICLK low (clock polarity = 1)	35		35		ns
11	$t_{v(SPCH-SOMI)M}$	Valid time, SPISOMI data valid after SPICLK high (clock polarity = 0)	$0.25t_{c(SPC)M}$ -10		$0.5t_{c(SPC)M}$ -10		ns
	$t_{v(SPCL-SOMI)M}$	Valid time, SPISOMI data valid after SPICLK low (clock polarity = 1)	$0.25_{tc(SPC)M}$ -10		$0.5_{tc(SPC)M}$ -10		ns

(1) The MASTER/SLAVE bit (SPICTL.2) is set and the CLOCK PHASE bit (SPICTL.3) is set.
(2) $t_{c(SPC)}$ = SPI clock cycle time = LSPCLK/4 or LSPCLK/(SPIBRR + 1)
(3) Internal clock prescalers must be adjusted such that the SPI clock speed is limited to the following SPI clock rate:
Master mode transmit 25 MHz MAX, master mode receive 12.5 MHz MAX
Slave mode transmit 12.5 MHz MAX, slave mode receive 12.5 MHz MAX.
(4) $t_{c(LCO)}$ = LSPCLK cycle time
(5) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).



A. In the master mode, $\overline{\text{SPISTE}}$ goes active $0.5t_{c(SPC)}$ (minimum) before valid SPI clock edge. On the trailing end of the word, the $\overline{\text{SPISTE}}$ will go inactive $0.5t_{c(SPC)}$ after the receiving edge (SPICLK) of the last data bit, except that $\overline{\text{SPISTE}}$ stays active between back-to-back transmit words in both FIFO and nonFIFO modes.

Figure 6-19. SPI Master Mode External Timing (Clock Phase = 1)

6.10.6 SPI Slave Mode Timing

Table 6-35 lists the slave mode external timing (clock phase = 0) and Table 6-36 (clock phase = 1). Figure 6-20 and Figure 6-21 show the timing waveforms.

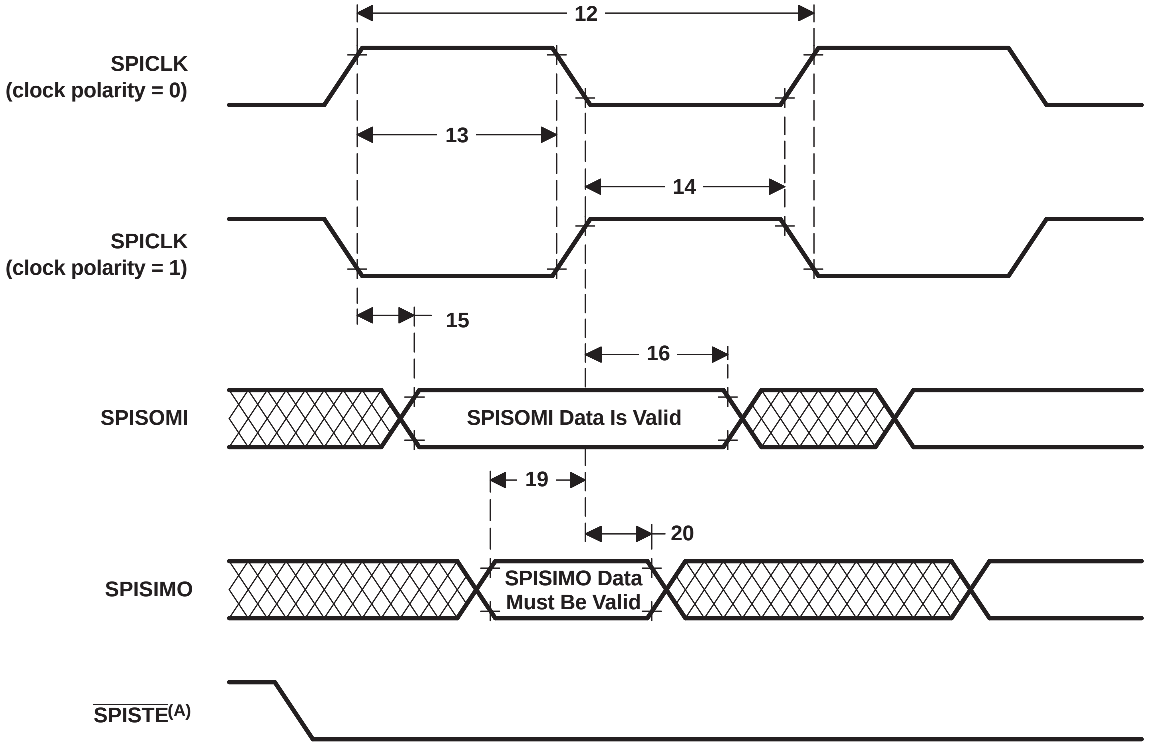
Table 6-35. SPI Slave Mode External Timing (Clock Phase = 0)(1) (2) (3) (4) (5)

NO.			MIN	MAX	UNIT
12	$t_{c(SPC)S}$	Cycle time, SPICLK	$4t_{c(LCO)}$		ns
13	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
14	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
15	$t_{d(SPCH-SOMI)S}$	Delay time, SPICLK high to SPISOMI valid (clock polarity = 0)		35	ns
	$t_{d(SPCL-SOMI)S}$	Delay time, SPICLK low to SPISOMI valid (clock polarity = 1)		35	ns
16	$t_{v(SPCL-SOMI)S}$	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)	$0.75t_{c(SPC)S}$		ns

(1) The MASTER / SLAVE bit (SPICTL.2) is cleared and the CLOCK PHASE bit (SPICTL.3) is cleared.
(2) $t_{c(SPC)}$ = SPI clock cycle time = LSPCLK/4 or LSPCLK/(SPIBRR + 1)
(3) Internal clock prescalers must be adjusted such that the SPI clock speed is limited to the following SPI clock rate:
Master mode transmit 25-MHz MAX, master mode receive 12.5-MHz MAX
Slave mode transmit 12.5-MHz MAX, slave mode receive 12.5-MHz MAX.
(4) $t_{c(LCO)}$ = LSPCLK cycle time
(5) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).

Table 6-35. SPI Slave Mode External Timing (Clock Phase = 0) (continued)

NO.			MIN	MAX	UNIT
19	$t_{su(SIMO-SPCL)S}$	Setup time, SPISIMO before SPICLK low (clock polarity = 0)	35		ns
	$t_{su(SIMO-SPCH)S}$	Setup time, SPISIMO before SPICLK high (clock polarity = 1)	35		ns
20	$t_{v(SPCL-SIMO)S}$	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)S-10}$		ns
	$t_{v(SPCH-SIMO)S}$	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)S-10}$		ns



A. In the slave mode, the SPISTE signal should be asserted low at least $0.5t_{c(SPC)}$ (minimum) before the valid SPI clock edge and remain low for at least $0.5t_{c(SPC)}$ after the receiving edge (SPICLK) of the last data bit.

Figure 6-20. SPI Slave Mode External Timing (Clock Phase = 0)

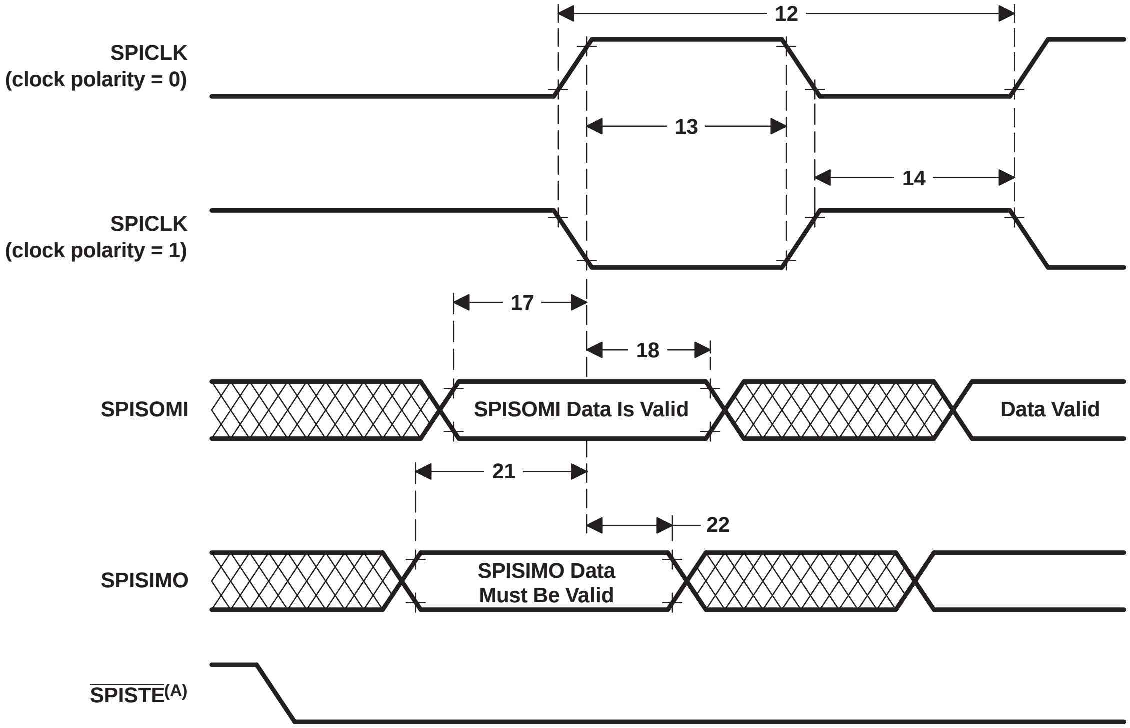
Table 6-36. SPI Slave Mode External Timing (Clock Phase = 1)(1)(2)(3)(4)

NO.			MIN	MAX	UNIT
12	$t_{c(SPC)S}$	Cycle time, SPICLK	$8t_{c(LCO)}$		ns
13	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 0)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 1)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
14	$t_{w(SPCL)S}$	Pulse duration, SPICLK low (clock polarity = 0)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
	$t_{w(SPCH)S}$	Pulse duration, SPICLK high (clock polarity = 1)	$0.5t_{c(SPC)S}$ - 10	$0.5t_{c(SPC)S}$	ns
17	$t_{su(SOMI-SPCH)S}$	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	$0.125t_{c(SPC)S}$		ns
18	$t_{v(SPCH-SOMI)S}$	Valid time, SPISOMI data valid after SPICLK low (clock polarity = 0)	$0.75t_{c(SPC)S}$		ns

(1) The MASTER / SLAVE bit (SPICTL.2) is cleared and the CLOCK PHASE bit (SPICTL.3) is cleared.
(2) $t_{c(SPC)}$ = SPI clock cycle time = LSPCLK/4 or LSPCLK/(SPIBRR + 1)
(3) Internal clock prescalers must be adjusted such that the SPI clock speed is limited to the following SPI clock rate:
Master mode transmit 25-MHz MAX, master mode receive 12.5-MHz MAX
Slave mode transmit 12.5-MHz MAX, slave mode receive 12.5-MHz MAX.
(4) The active edge of the SPICLK signal referenced is controlled by the CLOCK POLARITY bit (SPICCR.6).

Table 6-36. SPI Slave Mode External Timing (Clock Phase = 1) (continued)

NO.			MIN	MAX	UNIT
	$t_{v(SPCL-SOMI)S}$	Valid time, SPISOMI data valid after SPICLK high (clock polarity = 1)	0.75$t_{c(SPC)S}$		ns
21	$t_{su(SIMO-SPCH)S}$	Setup time, SPISIMO before SPICLK high (clock polarity = 0)	35		ns
	$t_{su(SIMO-SPCL)S}$	Setup time, SPISIMO before SPICLK low (clock polarity = 1)	35		ns
22	$t_{v(SPCH-SIMO)S}$	Valid time, SPISIMO data valid after SPICLK high (clock polarity = 0)	0.5$t_{c(SPC)S-10}$		ns
	$t_{v(SPCL-SIMO)S}$	Valid time, SPISIMO data valid after SPICLK low (clock polarity = 1)	0.5$t_{c(SPC)S-10}$		ns



A. In the slave mode, the $\overline{\text{SPISTE}}$ signal should be asserted low at least 0.5$t_{c(SPC)}$ before the valid SPI clock edge and remain low for at least 0.5$t_{c(SPC)}$ after the receiving edge (SPICLK) of the last data bit.

Figure 6-21. SPI Slave Mode External Timing (Clock Phase = 1)

6.10.7 External Interface (XINTF) Timing

Each XINTF access consists of three parts: Lead, Active, and Trail. The user configures the Lead/Active/Trail wait states in the XTIMING registers. There is one XTIMING register for each XINTF zone. Table 6-37 shows the relationship between the parameters configured in the XTIMING register and the duration of the pulse in terms of XTIMCLK cycles.

Table 6-37. Relationship Between Parameters Configured in XTIMING and Duration of Pulse

	DESCRIPTION	DURATION (ns)[1][2]	
		X2TIMING = 0	X2TIMING = 1
LR	Lead period, read access	XRDLEAD × $t_{c(XTIM)}$	(XRDLEAD × 2) × $t_{c(XTIM)}$
AR	Active period, read access	(XRDACTIVE + WS + 1) × $t_{c(XTIM)}$	(XRDACTIVE × 2 + WS + 1) × $t_{c(XTIM)}$
TR	Trail period, read access	XRDTRAIL × $t_{c(XTIM)}$	(XRDTRAIL × 2) × $t_{c(XTIM)}$

(1) $t_{c(XTIM)}$ – Cycle time, XTIMCLK
(2) WS refers to the number of wait states inserted by hardware when using XREADY. If the zone is configured to ignore XREADY (USEREADY = 0), then WS = 0.

Table 6-37. Relationship Between Parameters Configured in XTIMING and Duration of Pulse (continued)

	DESCRIPTION	DURATION (ns)[1][2]	
LW	Lead period, write access	XWRLEAD × $t_{c(XTIM)}$	(XWRLEAD × 2) × $t_{c(XTIM)}$
AW	Active period, write access	(XWRACTIVE + WS + 1) × $t_{c(XTIM)}$	(XWRACTIVE × 2 + WS + 1) × $t_{c(XTIM)}$
TW	Trail period, write access	XWRTRAIL × $t_{c(XTIM)}$	(XWRTRAIL × 2) × $t_{c(XTIM)}$

Minimum wait state requirements must be met when configuring each zone's XTIMING register. These requirements are in addition to any timing requirements as specified by that device's data sheet. No internal device hardware is included to detect illegal settings.

6.10.7.1 USEREADY = 0

If the XREADY signal is ignored (USEREADY = 0), then:

Lead: $LR \geq t_{c(XTIM)}$

$LW \geq t_{c(XTIM)}$

These requirements result in the following XTIMING register configuration restrictions:

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 0	≥ 0	≥ 1	≥ 0	≥ 0	0, 1

Examples of valid and invalid timing when not sampling XREADY:

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Invalid(1)	0	0	0	0	0	0	0, 1
Valid	1	0	0	1	0	0	0, 1

(1) No hardware to detect illegal XTIMING configurations

6.10.7.2 Synchronous Mode (USEREADY = 1, READYMODE = 0)

If the XREADY signal is sampled in the synchronous mode (USEREADY = 1, READYMODE = 0), then:

1 Lead: $LR \geq \times t_{c(XTIM)}$

$LW \geq t_{c(XTIM)}$

2 Active: $AR \geq 2 \times t_{c(XTIM)}$

$AW \geq 2 \times t_{c(XTIM)}$

NOTE

Restriction does not include external hardware wait states.

These requirements result in the following XTIMING register configuration restrictions:

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 1	≥ 0	≥ 1	≥ 1	≥ 0	0, 1

Examples of valid and invalid timing when using synchronous XREADY:

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Invalid(1)	0	0	0	0	0	0	0, 1
Invalid(1)	1	0	0	1	0	0	0, 1
Valid	1	1	0	1	1	0	0, 1

(1) No hardware to detect illegal XTIMING configurations

6.10.7.3 Asynchronous Mode (USEREADY = 1, READYMODE = 1)

If the XREADY signal is sampled in the asynchronous mode (USEREADY = 1, READYMODE = 1), then:

1 Lead: $LR \ge \times t_{c(XTIM)}$
$LW \ge t_{c(XTIM)}$
2 Active: $AR \ge 2 \times t_{c(XTIM)}$
$AW \ge 2 \times t_{c(XTIM)}$

3 Lead + Active: $LR + AR \ge 4 \times t_{c(XTIM)}$
$LW + AW \ge 4 \times t_{c(XTIM)}$

NOTE

Restrictions do not include external hardware wait states.

These requirements result in the following XTIMING register configuration restrictions:

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 2	0	≥ 1	≥ 2	0	0, 1

or

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 2	≥ 1	0	≥ 2	≥ 1	0	0, 1

Examples of valid and invalid timing when using asynchronous XREADY:

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Invalid (1)	0	0	0	0	0	0	0, 1
Invalid (1)	1	0	0	1	0	0	0, 1
Invalid (1)	1	1	0	1	1	0	0
Valid	1	1	0	1	1	0	1
Valid	1	2	0	1	2	0	0, 1
Valid	2	1	0	2	1	0	0, 1

(1) No hardware to detect illegal XTIMING configurations

Unless otherwise specified, all XINTF timing is applicable for the clock configurations shown in Table 6-38.

Table 6-38. XINTF Clock Configurations

MODE	SYSCLKOUT	XTIMCLK	XCLKOUT
1		SYSCLKOUT	SYSCLKOUT
Example:	150 MHz	150 MHz	150 MHz
2		SYSCLKOUT	1/2 SYSCLKOUT
Example:	150 MHz	150 MHz	75 MHz
3		1/2 SYSCLKOUT	1/2 SYSCLKOUT
Example:	150 MHz	75 MHz	75 MHz
4		1/2 SYSCLKOUT	1/4 SYSCLKOUT
Example:	150 MHz	75 MHz	37.5 MHz

The relationship between SYSCLKOUT and XTIMCLK is shown in Figure 6-22.

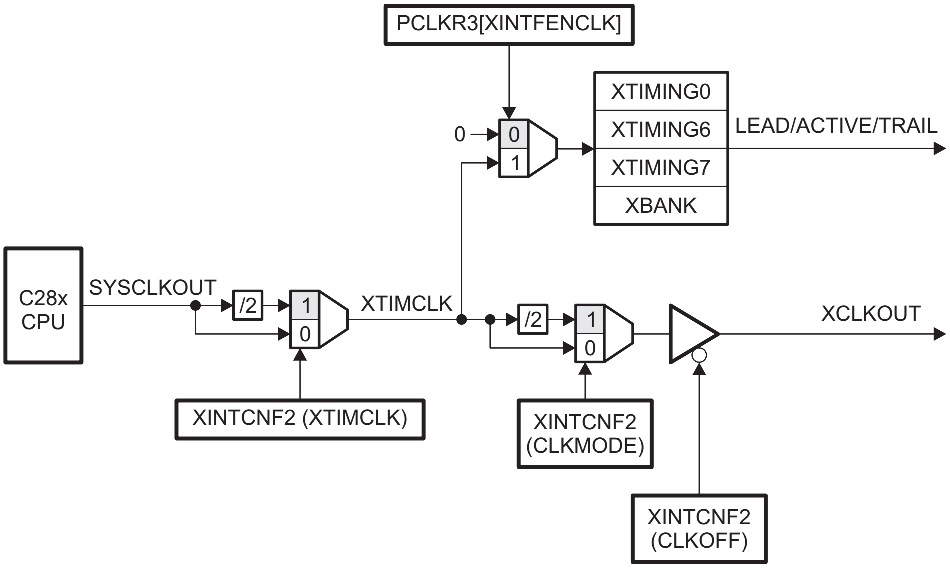


Figure 6-22. Relationship Between XTIMCLK and SYSCLKOUT

6.10.7.4 XINTF Signal Alignment to XCLKOUT

For each XINTF access, the number of lead, active, and trail cycles is based on the internal clock XTIMCLK. Strobes such as $\overline{\text{XRD}}$, $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$, and zone chip-select ($\overline{\text{XZCS}}$) change state in relationship to the rising edge of XTIMCLK. The external clock, XCLKOUT, can be configured to be either equal to or one-half the frequency of XTIMCLK.

For the case where XCLKOUT = XTIMCLK, all of the XINTF strobes will change state with respect to the rising edge of XCLKOUT. For the case where XCLKOUT = one-half XTIMCLK, some strobes will change state either on the rising edge of XCLKOUT or the falling edge of XCLKOUT. In the XINTF timing tables, the notation XCOHL is used to indicate that the parameter is with respect to either case; XCLKOUT rising edge (high) or XCLKOUT falling edge (low). If the parameter is always with respect to the rising edge of XCLKOUT, the notation XCOH is used.

For the case where XCLKOUT = one-half XTIMCLK, the XCLKOUT edge with which the change will be aligned can be determined based on the number of XTIMCLK cycles from the start of the access to the point at which the signal changes. If this number of XTIMCLK cycles is even, the alignment will be with respect to the rising edge of XCLKOUT. If this number is odd, then the signal will change with respect to the falling edge of XCLKOUT. Examples include the following:

- Strobes that change at the beginning of an access always align to the rising edge of XCLKOUT. This is because all XINTF accesses begin with respect to the rising edge of XCLKOUT.

Examples:	XZCSL	Zone chip-select active low
	XRNWL	XR/$\overline{\text{W}}$ active low

- Strobes that change at the beginning of the active period will align to the rising edge of XCLKOUT if the total number of lead XTIMCLK cycles for the access is even. If the number of lead XTIMCLK cycles is odd, then the alignment will be with respect to the falling edge of XCLKOUT.

Examples:	XRDL	$\overline{\text{XRD}}$ active low
	XWEL	$\overline{\text{XWE1}}$ or $\overline{\text{XWE0}}$ active low

- Strobes that change at the beginning of the trail period will align to the rising edge of XCLKOUT if the total number of lead + active XTIMCLK cycles (including hardware waitstates) for the access is even. If the number of lead + active XTIMCLK cycles (including hardware waitstates) is odd, then the alignment will be with respect to the falling edge of XCLKOUT.

Examples:	XRDH	$\overline{\text{XRD}}$ inactive high
	XWEH	$\overline{\text{XWE1}}$ or $\overline{\text{XWE0}}$ inactive high

- Strobes that change at the end of the access will align to the rising edge of XCLKOUT if the total number of lead + active + trail XTIMCLK cycles (including hardware waitstates) is even. If the number of lead + active + trail XTIMCLK cycles (including hardware waitstates) is odd, then the alignment will be with respect to the falling edge of XCLKOUT.

Examples:	XZCSH	Zone chip-select inactive high
	XRNWH	XR/$\overline{\text{W}}$ inactive high

6.10.7.5 External Interface Read Timing

Table 6-39. External Interface Read Timing Requirements

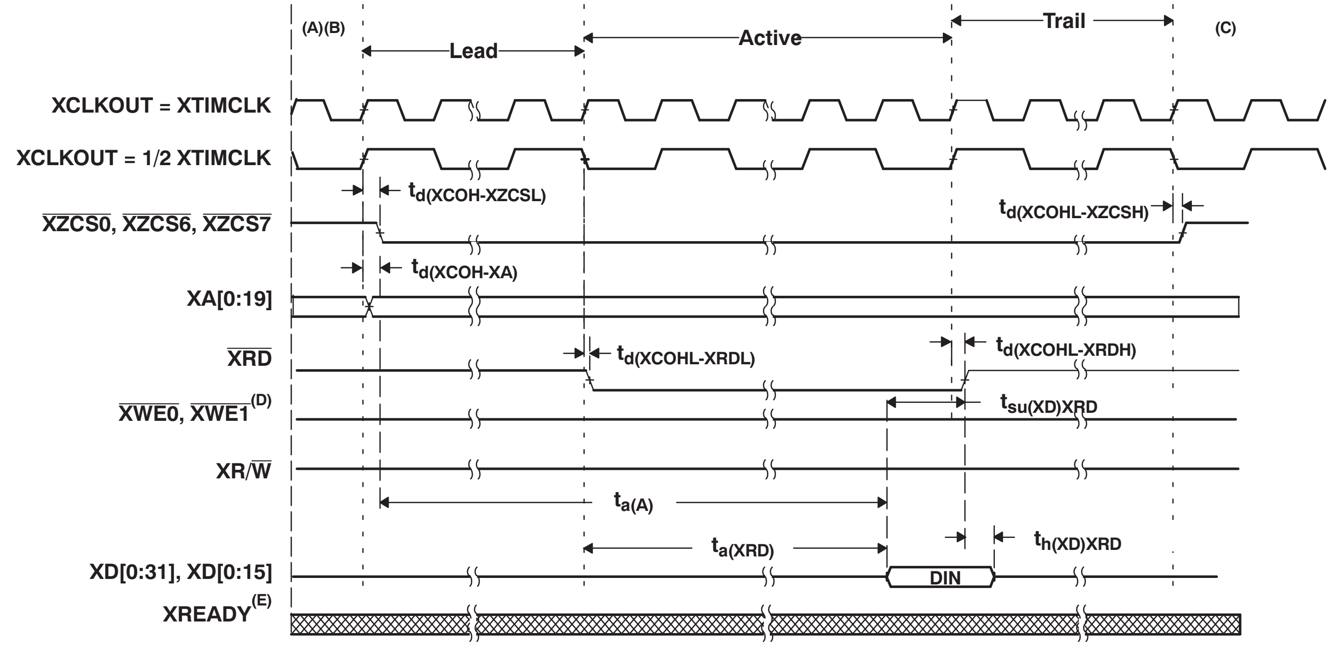
		MIN	MAX	UNIT
$t_{a(A)}$	Access time, read data from address valid		(LR + AR) –16 (1)	ns
$t_{a(XRD)}$	Access time, read data valid from $\overline{\text{XRD}}$ active low		AR –14 (1)	ns
$t_{su(XD)XRD}$	Setup time, read data valid before $\overline{\text{XRD}}$ strobe inactive high	14		ns
$t_{h(XD)XRD}$	Hold time, read data valid after $\overline{\text{XRD}}$ inactive high	0		ns

(1) LR = Lead period, read access. AR = Active period, read access. See Table 6-37.

Table 6-40. External Interface Read Switching Characteristics

	PARAMETER	MIN	MAX	UNIT
$t_{d(XCOH-XZCSL)}$	Delay time, XCLKOUT high to zone chip-select active low		1	ns
$t_{d(XCOHL-XZCSH)}$	Delay time, XCLKOUT high/low to zone chip-select inactive high	–2	3	ns
$t_{d(XCOH-XA)}$	Delay time, XCLKOUT high to address valid		2	ns
$t_{d(XCOHL-XRDL)}$	Delay time, XCLKOUT high/low to $\overline{\text{XRD}}$ active low		1	ns
$t_{d(XCOHL-XRDH}$	Delay time, XCLKOUT high/low to $\overline{\text{XRD}}$ inactive high	–2	1	ns
$t_{h(XA)XZCSH}$	Hold time, address valid after zone chip-select inactive high	(1)		ns
$t_{h(XA)XRD}$	Hold time, address valid after $\overline{\text{XRD}}$ inactive high	(1)		ns

(1) During inactive cycles, the XINTF address bus always holds the last address put out on the bus. This includes alignment cycles.



A. All XINTF accesses (lead period) begin on the rising edge of XCLKOUT. When necessary, the device inserts an alignment cycle before an access to meet this requirement.

B. During alignment cycles, all signals transition to their inactive state.

C. XA[0:19] holds the last address put on the bus during inactive cycles, including alignment cycles.

D. $\overline{\text{XWE1}}$ is used in 32-bit data bus mode. In 16-bit mode, this signal is XA0.

E. For USEREADY = 0, the external XREADY input signal is ignored.

Figure 6-23. Example Read Access

XTIMING register parameters used for this example:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
≥ 1	≥ 0	≥ 0	0	0	N/A (1)	N/A (1)	N/A (1)	N/A (1)

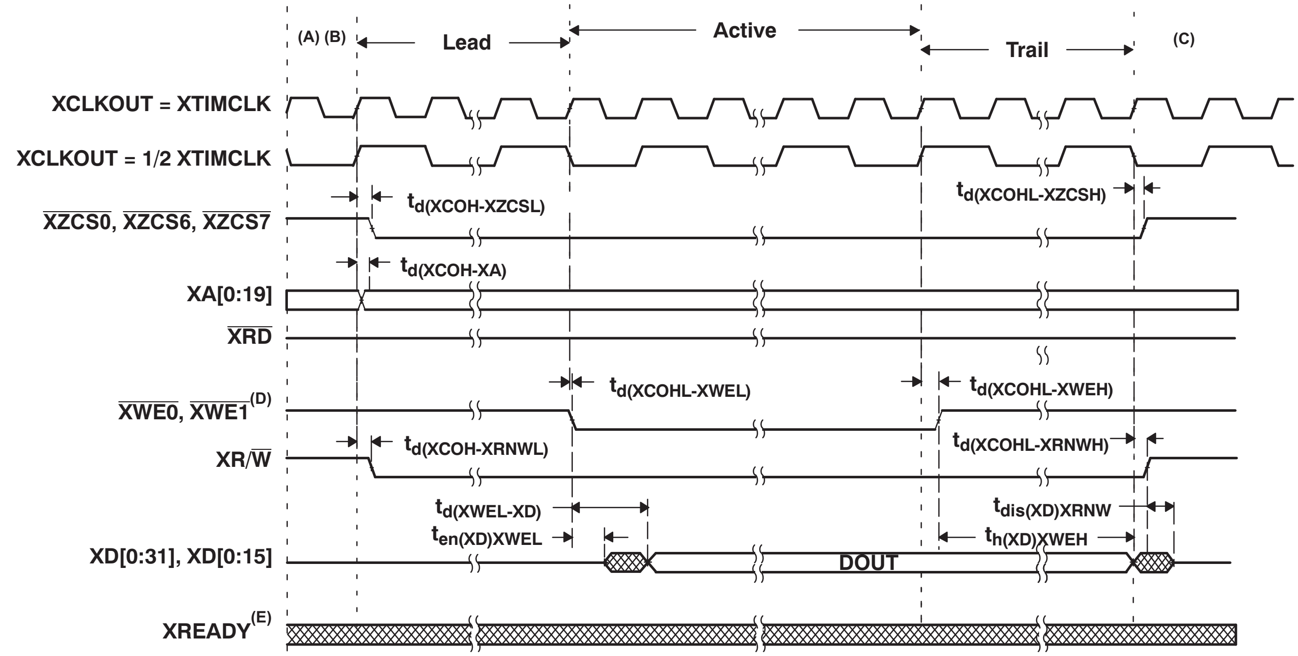
(1) N/A = Not applicable (or "Don't care") for this example

6.10.7.6 External Interface Write Timing

Table 6-41. External Interface Write Switching Characteristics

	PARAMETER	MIN	MAX	UNIT
$t_{d(XCOH-XZCSL)}$	Delay time, XCLKOUT high to zone chip-select active low		1	ns
$t_{d(XCOHL-XZCSH)}$	Delay time, XCLKOUT high or low to zone chip-select inactive high	- 2	3	ns
$t_{d(XCOH-XA)}$	Delay time, XCLKOUT high to address valid		2	ns
$t_{d(XCOHL-XWEL)}$	Delay time, XCLKOUT high/low to $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$ (1) low		2	ns
$t_{d(XCOHL-XWEH)}$	Delay time, XCLKOUT high/low to $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$ high		2	ns
$t_{d(XCOH-XRNWL)}$	Delay time, XCLKOUT high to XR/$\overline{\text{W}}$ low		1	ns
$t_{d(XCOHL-XRNWH)}$	Delay time, XCLKOUT high/low to XR/$\overline{\text{W}}$ high	- 2	1	ns
$t_{en(XD)XWEL}$	Enable time, data bus driven from $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$ low	0		ns
$t_{d(XWEL-XD)}$	Delay time, data valid after $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$ active low		4	ns
$t_{h(XA)XZCSH}$	Hold time, address valid after zone chip-select inactive high	(2)		ns
$t_{h(XD)XWE}$	Hold time, write data valid after $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$ inactive high	TW-2 (3)		ns
$t_{dis(XD)XRNW}$	Maximum time for DSP to release the data bus after XR/$\overline{\text{W}}$ inactive high		4	ns

(1) $\overline{\text{XWE1}}$ is used in 32-bit data bus mode only. In 16-bit mode, this signal is XA0.
(2) During inactive cycles, the XINTF address bus will always hold the last address put out on the bus. This includes alignment cycles.
(3) TW = Trail period, write access. See Table 6-37.



A. All XINTF accesses (lead period) begin on the rising edge of XCLKOUT. When necessary, the device inserts an alignment cycle before an access to meet this requirement.

B. During alignment cycles, all signals transition to their inactive state.

C. XA[0:19] holds the last address put on the bus during inactive cycles, including alignment cycles.

D. $\overline{\text{XWE1}}$ is used in 32-bit data bus mode. In 16-bit mode, this signal is XA0.

E. For USEREADY = 0, the external XREADY input signal is ignored.

Figure 6-24. Example Write Access

XTIMING register parameters used for this example:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
N/A(1)	N/A(1)	N/A(1)	0	0	≥ 1	≥ 0	≥ 0	N/A(1)

(1) N/A = Not applicable (or “Don’t care”) for this example

6.10.7.7 External Interface Ready-on-Read Timing With One External Wait State

Table 6-42. External Interface Read Switching Characteristics (Ready-on-Read, 1 Wait State)

	PARAMETER	MIN	MAX	UNIT
$t_{d(XCOH-XZCSL)}$	Delay time, XCLKOUT high to zone chip-select active low		1	ns
$t_{d(XCOHL-XZCSH)}$	Delay time, XCLKOUT high/low to zone chip-select inactive high	- 2	3	ns
$t_{d(XCOH-XA)}$	Delay time, XCLKOUT high to address valid		2	ns
$t_{d(XCOHL-XRDL)}$	Delay time, XCLKOUT high/low to $\overline{\text{XRD}}$ active low		1	ns
$t_{d(XCOHL-XRDH)}$	Delay time, XCLKOUT high/low to $\overline{\text{XRD}}$ inactive high	- 2	1	ns
$t_{h(XA)XZCSH}$	Hold time, address valid after zone chip-select inactive high	(1)		ns
$t_{h(XA)XRD}$	Hold time, address valid after $\overline{\text{XRD}}$ inactive high	(1)		ns

(1) During inactive cycles, the XINTF address bus always holds the last address put out on the bus. This includes alignment cycles.

Table 6-43. External Interface Read Timing Requirements (Ready-on-Read, 1 Wait State)

		MIN	MAX	UNIT
$t_{a(A)}$	Access time, read data from address valid		(LR + AR) - 16(1)	ns

(1) LR = Lead period, read access. AR = Active period, read access. See Table 6-37.

Table 6-43. External Interface Read Timing Requirements (Ready-on-Read, 1 Wait State) (continued)

		MIN	MAX	UNIT
$t_{a(XRD)}$	Access time, read data valid from $\overline{\text{XRD}}$ active low		AR - 14(1)	ns
$t_{su(XD)XRD}$	Setup time, read data valid before $\overline{\text{XRD}}$ strobe inactive high	14		ns
$t_{h(XD)XRD}$	Hold time, read data valid after $\overline{\text{XRD}}$ inactive high	0		ns

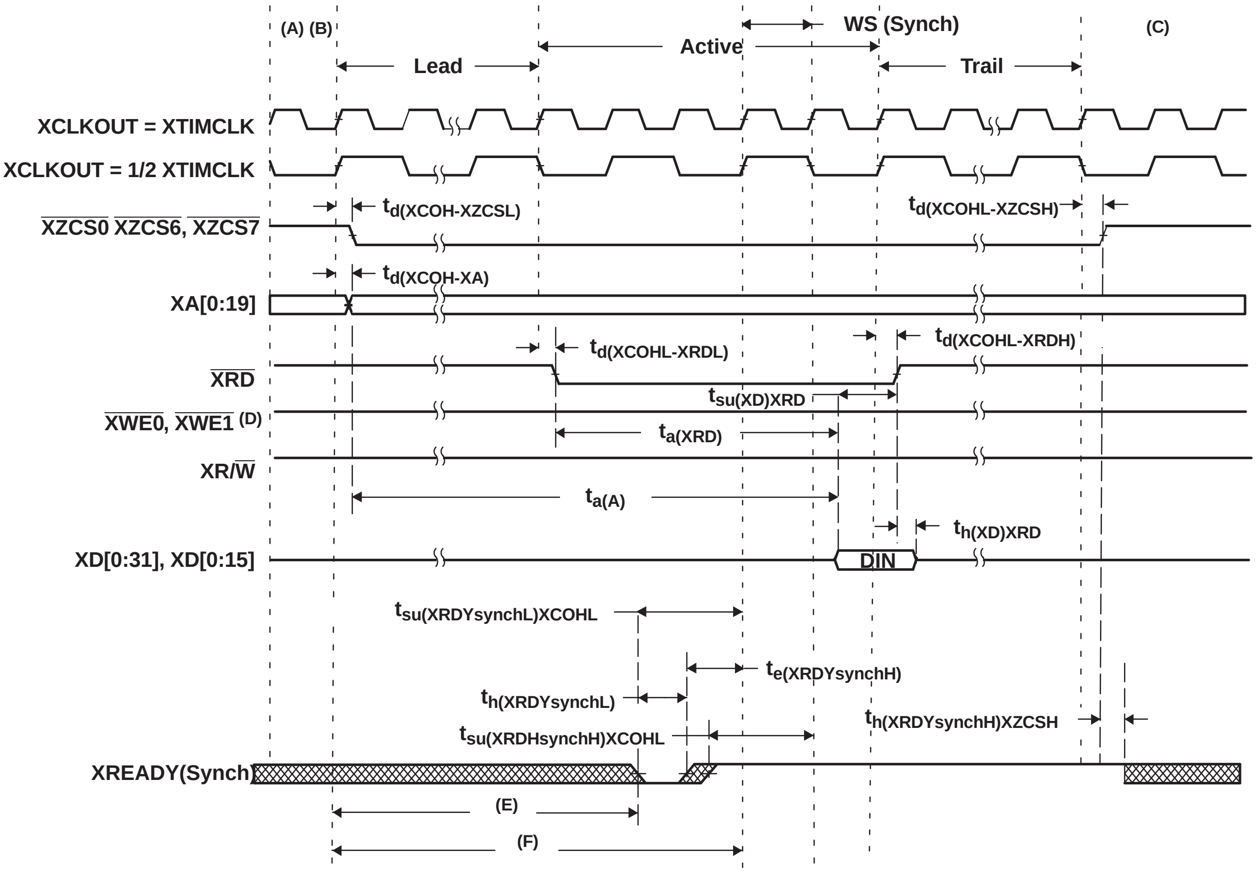
Table 6-44. Synchronous XREADY Timing Requirements (Ready-on-Read, 1 Wait State)(1)

		MIN	MAX	UNIT
$t_{su(XRDYsynchL)XCOHL}$	Setup time, XREADY (synchronous) low before XCLKOUT high/low	15		ns
$t_{h(XRDYsynchL)}$	Hold time, XREADY (synchronous) low	12		ns
$t_{e(XRDYsynchH)}$	Earliest time XREADY (synchronous) can go high before the sampling XCLKOUT edge		3	ns
$t_{su(XRDYsynchH)XCOHL}$	Setup time, XREADY (synchronous) high before XCLKOUT high/low	15		ns
$t_{h(XRDYsynchH)XZCSH}$	Hold time, XREADY (synchronous) held high after zone chip select high	0		ns

(1) The first XREADY (synchronous) sample occurs with respect to E in Figure 6-25:
E = (XRDLEAD + XRDACTIVE) $t_{c(XTIM)}$
When first sampled, if XREADY (synchronous) is found to be high, then the access will complete. If XREADY (synchronous) is found to be low, it will be sampled again each $t_{c(XTIM)}$ until it is found to be high.
For each sample (n) the setup time (F) with respect to the beginning of the access can be calculated as:
F = (XRDLEAD + XRDACTIVE +n – 1) $t_{c(XTIM)} - t_{su(XRDYsynchL)XCOHL}$
where n is the sample number: n = 1, 2, 3, and so forth.

Table 6-45. Asynchronous XREADY Timing Requirements (Ready-on-Read, 1 Wait State)

		MIN	MAX	UNIT
$t_{su(XRDYAsynchL)XCOHL}$	Setup time, XREADY (asynchronous) low before XCLKOUT high/low	11		ns
$t_{h(XRDYAsynchL)}$	Hold time, XREADY (asynchronous) low	8		ns
$t_{e(XRDYAsynchH)}$	Earliest time XREADY (asynchronous) can go high before the sampling XCLKOUT edge		3	ns
$t_{su(XRDYAsynchH)XCOHL}$	Setup time, XREADY (asynchronous) high before XCLKOUT high/low	11		ns
$t_{h(XRDYasynchH)XZCSH}$	Hold time, XREADY (asynchronous) held high after zone chip select high	0		ns



Legend:

▨ = Don't care. Signal can be high or low during this time.

A. All XINTF accesses (lead period) begin on the rising edge of XCLKOUT. When necessary, the device inserts an alignment cycle before an access to meet this requirement.

B. During alignment cycles, all signals transition to their inactive state.

C. During inactive cycles, the XINTF address bus always holds the last address put out on the bus. This includes alignment cycles.

D. $\overline{XWE1}$ is valid only in 32-bit data bus mode. In 16-bit mode, this signal is XA0.

E. For each sample, setup time from the beginning of the access (E) can be calculated as: D = (XRDLEAD + XRDACTIVE +n - 1) $t_{c(XTIM)} - t_{su(XRDYsynchL)XCOHL}$

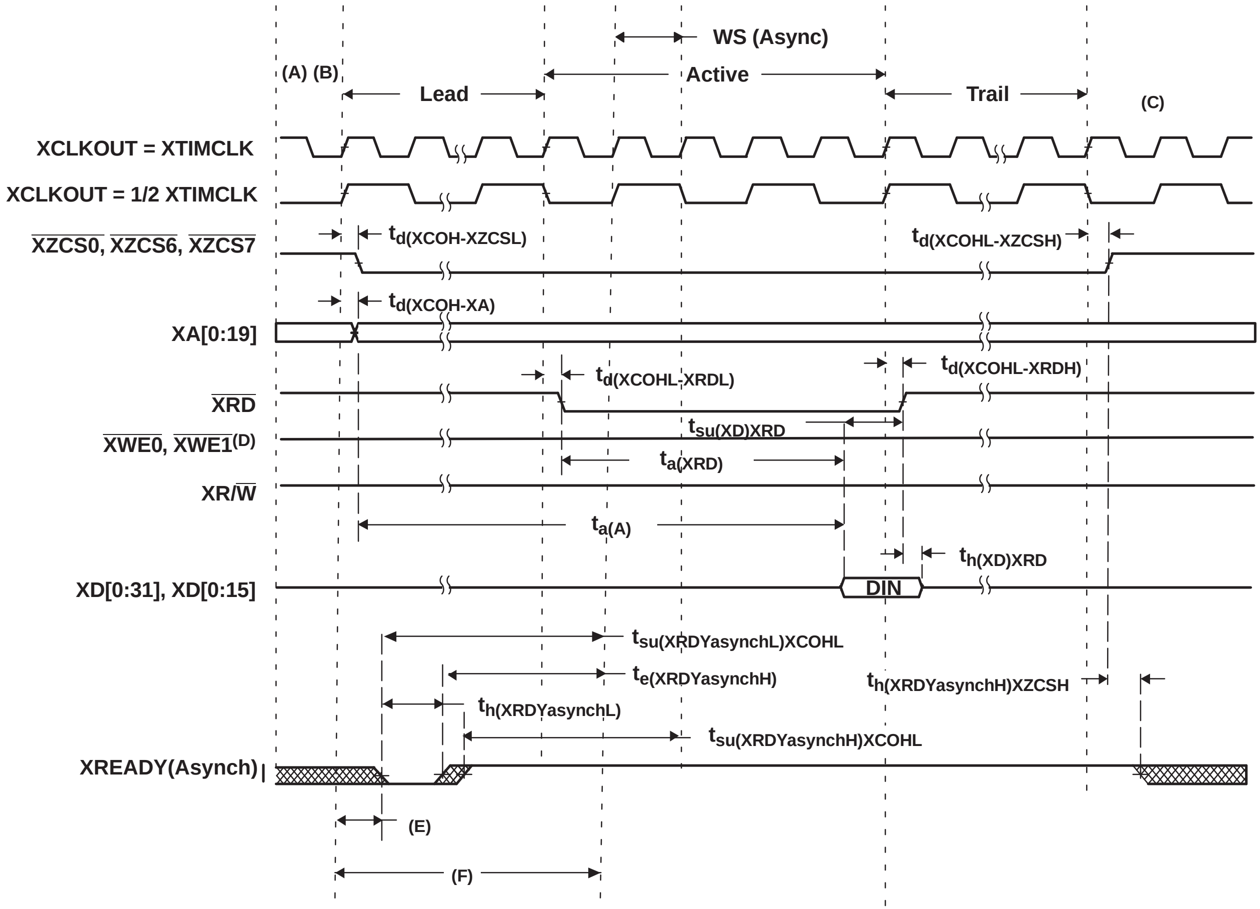
F. Reference for the first sample is with respect to this point: F = (XRDLEAD + XRDACTIVE) $t_{c(XTIM)}$ where n is the sample number: n = 1, 2, 3, and so forth.

Figure 6-25. Example Read With Synchronous XREADY Access

XTIMING register parameters used for this example:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
≥ 1	3	≥ 1	1	0	N/A (1)	N/A (1)	N/A (1)	0 = XREADY (Synch)

(1) N/A = "Don't care" for this example



Legend:

▨ = Don't care. Signal can be high or low during this time.

A. All XINTF accesses (lead period) begin on the rising edge of XCLKOUT. When necessary, the device will insert an alignment cycle before an access to meet this requirement.

B. During alignment cycles, all signals will transition to their inactive state.

C. During inactive cycles, the XINTF address bus will always hold the last address put out on the bus. This includes alignment cycles.

D. XWE1 is valid only in 32-bit data bus mode. In 16-bit mode, this signal is XA0.

E. For each sample, setup time from the beginning of the access can be calculated as: E = (XRDLEAD + XRDACTIVE -3 +n) $t_{c(XTIM)}$ – $t_{su(XRDYasynchL)XCOHL}$ where n is the sample number: n = 1, 2, 3, and so forth.

F. Reference for the first sample is with respect to this point: F = (XRDLEAD + XRDACTIVE –2) $t_{c(XTIM)}$

Figure 6-26. Example Read With Asynchronous XREADY Access

XTIMING register parameters used for this example:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
≥ 1	3	≥ 1	1	0	N/A (1)	N/A (1)	N/A (1)	1 = XREADY (Async)

(1) N/A = "Don't care" for this example

6.10.7.8 External Interface Ready-on-Write Timing With One External Wait State

Table 6-46. External Interface Write Switching Characteristics (Ready-on-Write, 1 Wait State)

	PARAMETER	MIN	MAX	UNIT
$t_{d(XCOH-XZCSL)}$	Delay time, XCLKOUT high to zone chip-select active low		1	ns
$t_{d(XCOHL-XZCSH)}$	Delay time, XCLKOUT high or low to zone chip-select inactive high	– 2	3	ns
$t_{d(XCOH-XA)}$	Delay time, XCLKOUT high to address valid		2	ns
$t_{d(XCOHL-XWEL)}$	Delay time, XCLKOUT high/low to $\overline{XWE0}$, $\overline{XWE1}$ low(1)		2	ns
$t_{d(XCOHL-XWEH)}$	Delay time, XCLKOUT high/low to $\overline{XWE0}$, $\overline{XWE1}$ high(1)		2	ns
$t_{d(XCOH-XRNWL)}$	Delay time, XCLKOUT high to XR/$\overline{W}$ low		1	ns
$t_{d(XCOHL-XRNWH)}$	Delay time, XCLKOUT high/low to XR/$\overline{W}$ high	– 2	1	ns
$t_{en(XD)XWEL}$	Enable time, data bus driven from $\overline{XWE0}$, $\overline{XWE1}$ low(1)	0		ns
$t_{d(XWEL-XD)}$	Delay time, data valid after $\overline{XWE0}$, $\overline{XWE1}$ active low(1)		4	ns
$t_{h(XA)XZCSH}$	Hold time, address valid after zone chip-select inactive high	(2)		ns
$t_{h(XD)XWE}$	Hold time, write data valid after $\overline{XWE0}$, $\overline{XWE1}$ inactive high(1)	TW-2(3)		ns
$t_{dis(XD)XRNW}$	Maximum time for DSP to release the data bus after XR/$\overline{W}$ inactive high		4	ns

(1) $\overline{XWE1}$ is used in 32-bit data bus mode only. In 16-bit, this signal is XA0.
(2) During inactive cycles, the XINTF address bus always holds the last address put out on the bus. This includes alignment cycles.
(3) TW = trail period, write access (see Table 6-37)

Table 6-47. Synchronous XREADY Timing Requirements (Ready-on-Write, 1 Wait State)(1)

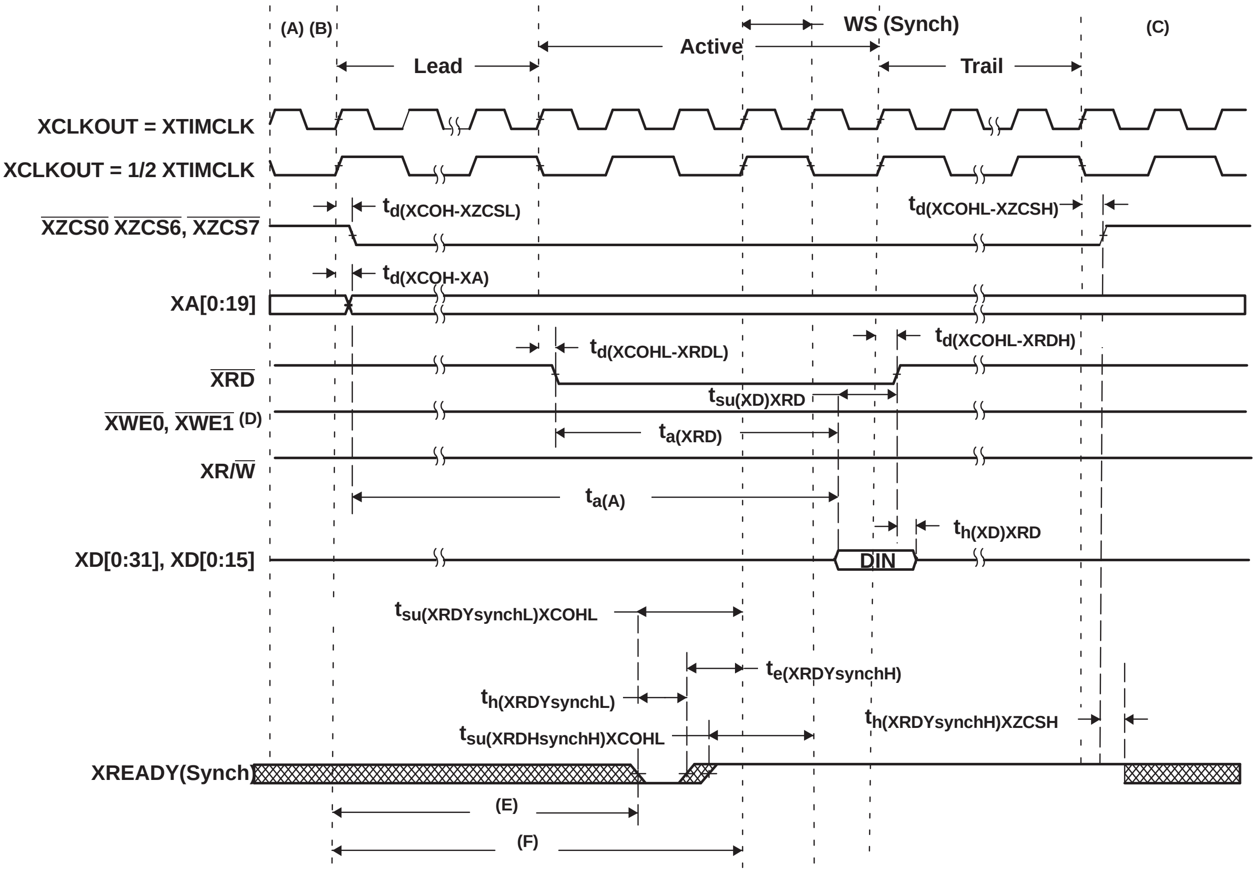
		MIN	MAX	UNIT
$t_{su(XRDYsynchL)XCOHL}$	Setup time, XREADY (synchronous) low before XCLKOUT high/low	15		ns
$t_{h(XRDYsynchL)}$	Hold time, XREADY (synchronous) low	12		ns
$t_{e(XRDYsynchH)}$	Earliest time XREADY (synchronous) can go high before the sampling XCLKOUT edge		3	ns
$t_{su(XRDYsynchH)XCOHL}$	Setup time, XREADY (synchronous) high before XCLKOUT high/low	15		ns
$t_{h(XRDYsynchH)XZCSH}$	Hold time, XREADY (synchronous) held high after zone chip select high	0		ns

(1) The first XREADY (synchronous) sample occurs with respect to E in Figure 6-27:
$E = (XWRLEAD + XWRACTIVE)\ t_{c(XTIM)}$
When first sampled, if XREADY (synchronous) is high, then the access will complete. If XREADY (synchronous) is low, it is sampled again each $t_{c(XTIM)}$ until it is high.
For each sample, setup time from the beginning of the access can be calculated as:
$F = (XWRLEAD + XWRACTIVE + n - 1)\ t_{c(XTIM)} - t_{su(XRDYsynchL)XCOHL}$
where n is the sample number: n = 1, 2, 3, and so forth.

Table 6-48. Asynchronous XREADY Timing Requirements (Ready-on-Write, 1 Wait State)(1)

		MIN	MAX	UNIT
$t_{su(XRDYasynchL)XCOHL}$	Setup time, XREADY (asynchronous) low before XCLKOUT high/low	11		ns
$t_{h(XRDYasynchL)}$	Hold time, XREADY (asynchronous) low	8		ns
$t_{e(XRDYasynchH)}$	Earliest time XREADY (asynchronous) can go high before the sampling XCLKOUT edge		3	ns
$t_{su(XRDYasynchH)XCOHL}$	Setup time, XREADY (asynchronous) high before XCLKOUT high/low	11		ns
$t_{h(XRDYasynchH)XZCSH}$	Hold time, XREADY (asynchronous) held high after zone chip select high	0		ns

(1) The first XREADY (synchronous) sample occurs with respect to E in Figure 6-27:
$E = (XWRLEAD + XWRACTIVE - 2)\ t_{c(XTIM)}$. When first sampled, if XREADY (asynchronous) is high, then the access will complete. If XREADY (asynchronous) is low, it is sampled again each $t_{c(XTIM)}$ until it is high.
For each sample, setup time from the beginning of the access can be calculated as:
$F = (XWRLEAD + XWRACTIVE - 3 + n)\ t_{c(XTIM)} - t_{su(XRDYasynchL)XCOHL}$
where n is the sample number: n = 1, 2, 3, and so forth.



Legend:

▨ = Don't care. Signal can be high or low during this time.

A. All XINTF accesses (lead period) begin on the rising edge of XCLKOUT. When necessary, the device inserts an alignment cycle before an access to meet this requirement.

B. During alignment cycles, all signals will transition to their inactive state.

C. During inactive cycles, the XINTF address bus always holds the last address put out on the bus. This includes alignment cycles.

D. $\overline{\text{XWE1}}$ is used in 32-bit data bus mode only. In 16-bit, this signal is XA0

E. For each sample, setup time from the beginning of the access can be calculated as E = (XWRLEAD + XWRACTIVE + n –1) $t_{c(XTIM)}$ – $t_{su(XRDYsynchL)XCOH}$ where n is the sample number: n = 1, 2, 3, and so forth.

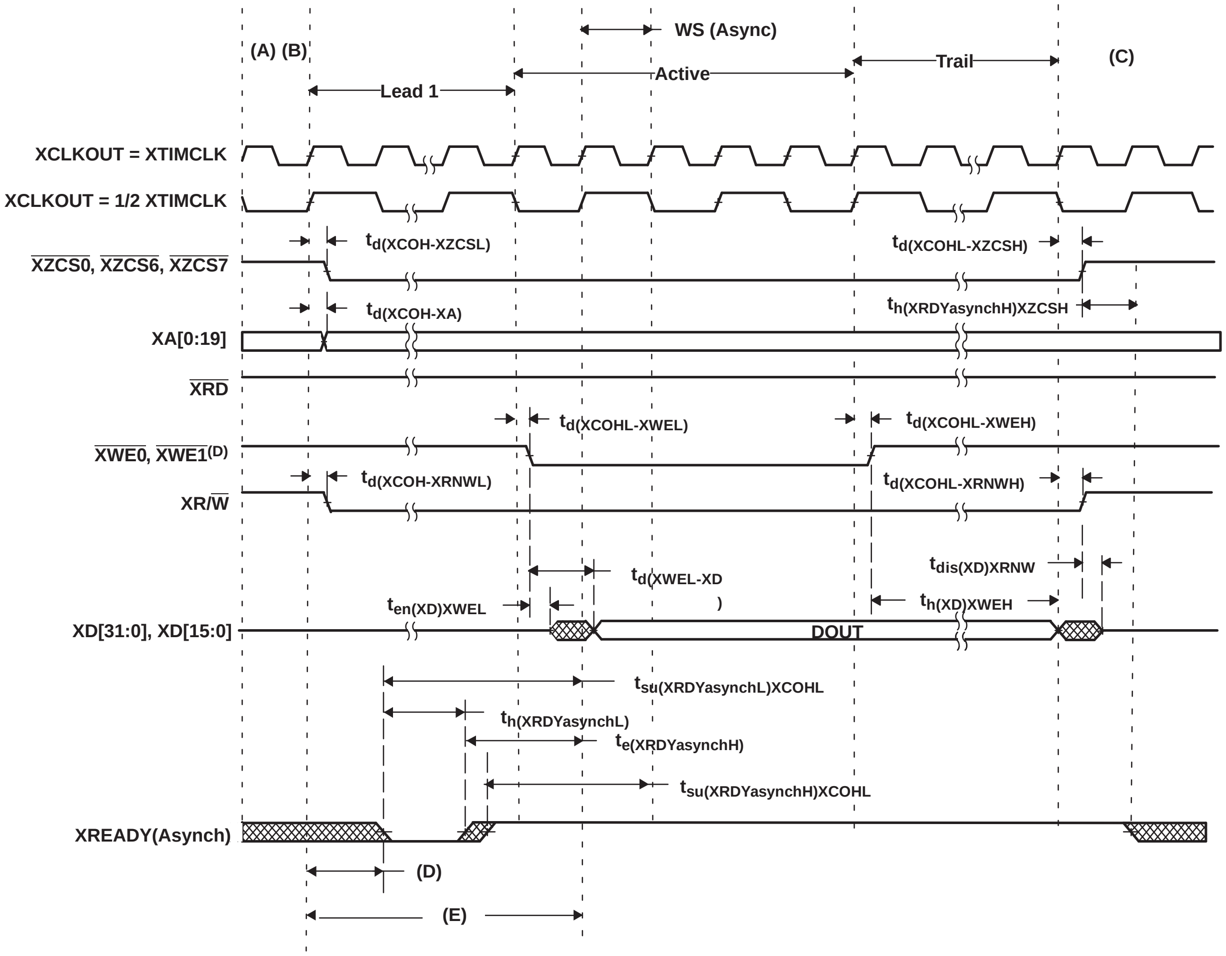
F. Reference for the first sample is with respect to this point: F = (XWRLEAD + XWRACTIVE) $t_{c(XTIM)}$

Figure 6-27. Write With Synchronous XREADY Access

XTIMING register parameters used for this example:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
N/A (1)	N/A (1)	N/A (1)	1	0	≥ 1	3	≥ 1	0 = XREADY (Synch)

(1) N/A = "Don't care" for this example.



Legend:

▨ = Don't care. Signal can be high or low during this time.

A. All XINTF accesses (lead period) begin on the rising edge of XCLKOUT. When necessary, the device inserts an alignment cycle before an access to meet this requirement.

B. During alignment cycles, all signals transition to their inactive state.

C. During inactive cycles, the XINTF address bus always holds the last address put out on the bus. This includes alignment cycles.

D. $\overline{\text{XWE1}}$ is used in 32-bit data bus mode only. In 16-bit, this signal is XA0.

E. For each sample, set up time from the beginning of the access can be calculated as: E = (XWRLEAD + XWRACTIVE -3 + n) $t_{c(XTIM)} - t_{su(XRDYasynchL)XCOHL}$ where n is the sample number: n = 1, 2, 3, and so forth.

F. Reference for the first sample is with respect to this point: F = (XWRLEAD + XWRACTIVE – 2) $t_{c(XTIM)}$

Figure 6-28. Write With Asynchronous XREADY Access

XTIMING register parameters used for this example:

XRDLEAD	XRDACTIVE	XRDTRAIL	USEREADY	X2TIMING	XWRLEAD	XWRACTIVE	XWRTRAIL	READYMODE
N/A(1)	N/A(1)	N/A(1)	1	0	≥ 1	3	≥ 1	1 = XREADY (Async)

(1) N/A = "Don't care" for this example

6.10.8 $\overline{\text{XHOLD}}$ and $\overline{\text{XHOLDA}}$ Timing

If the HOLD mode bit is set while $\overline{\text{XHOLD}}$ and $\overline{\text{XHOLDA}}$ are both low (external bus accesses granted), the $\overline{\text{XHOLDA}}$ signal is forced high (at the end of the current cycle) and the external interface is taken out of high-impedance mode.

On a reset ($\overline{\text{XRS}}$), the HOLD mode bit is set to 0. If the $\overline{\text{XHOLD}}$ signal is active low on a system reset, the bus and all signal strobes must be in high-impedance mode, and the $\overline{\text{XHOLDA}}$ signal is also driven active low.

When HOLD mode is enabled and $\overline{\text{XHOLDA}}$ is active low (external bus grant active), the CPU can still execute code from internal memory. If an access is made to the external interface, the CPU is stalled until the $\overline{\text{XHOLD}}$ signal is removed.

An external DMA request, when granted, places the following signals in a high-impedance mode:

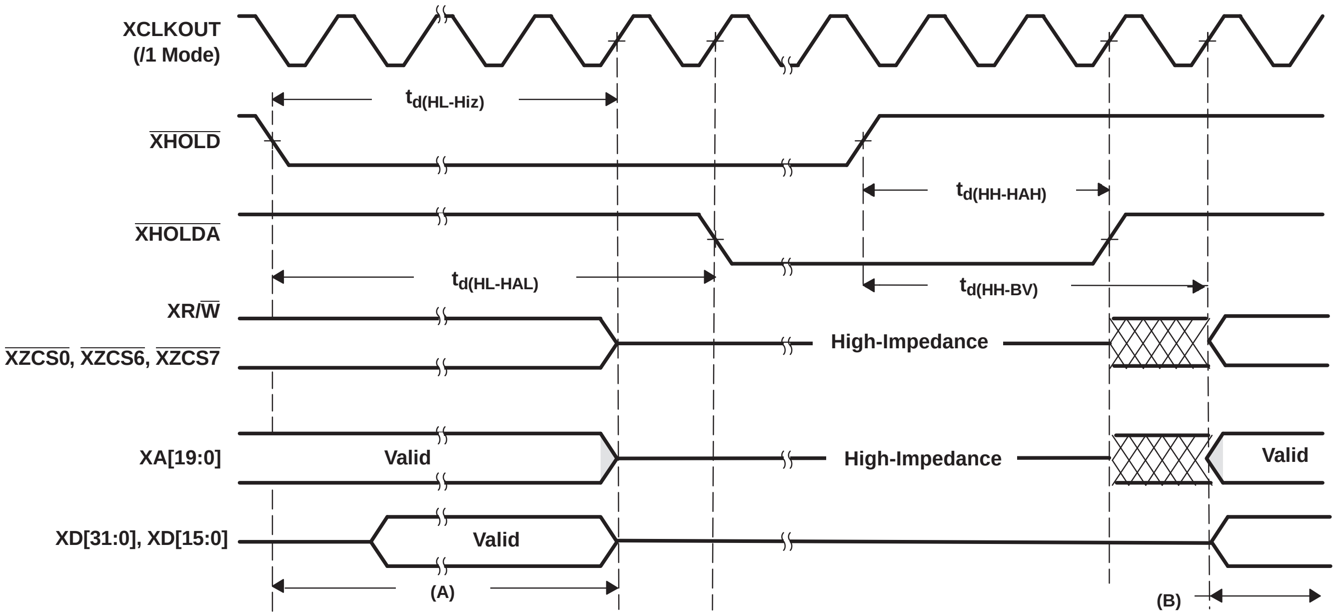
- XA[19:0]
- XD[31:0], XD[15:0]
- $\overline{\text{XWE0}}$, $\overline{\text{XWE1}}$, $\overline{\text{XRD}}$
- XR/$\overline{\text{W}}$
- $\overline{\text{XZCS0}}$
- $\overline{\text{XZCS6}}$
- $\overline{\text{XZCS7}}$

All other signals not listed in this group remain in their default or functional operational modes during these signal events.

Table 6-49. $\overline{\text{XHOLD}}$/$\overline{\text{XHOLDA}}$ Timing Requirements (XCLKOUT = XTIMCLK)(1)(2)

		MIN	MAX	UNIT
$t_{d(HL-HiZ)}$	Delay time, $\overline{\text{XHOLD}}$ low to Hi-Z on all address, data, and control		$4t_{c(XTIM)}$	ns
$t_{d(HL-HAL)}$	Delay time, $\overline{\text{XHOLD}}$ low to $\overline{\text{XHOLDA}}$ low		$5t_{c(XTIM)}$	ns
$t_{d(HH-HAH)}$	Delay time, $\overline{\text{XHOLD}}$ high to $\overline{\text{XHOLDA}}$ high		$3t_{c(XTIM)}$	ns
$t_{d(HH-BV)}$	Delay time, $\overline{\text{XHOLD}}$ high to bus valid		$4t_{c(XTIM)}$	ns

(1) When a low signal is detected on $\overline{\text{XHOLD}}$, all pending XINTF accesses will be completed before the bus is placed in a high-impedance state.
(2) The state of $\overline{\text{XHOLD}}$ is latched on the rising edge of XTIMCLK.



A. All pending XINTF accesses are completed.

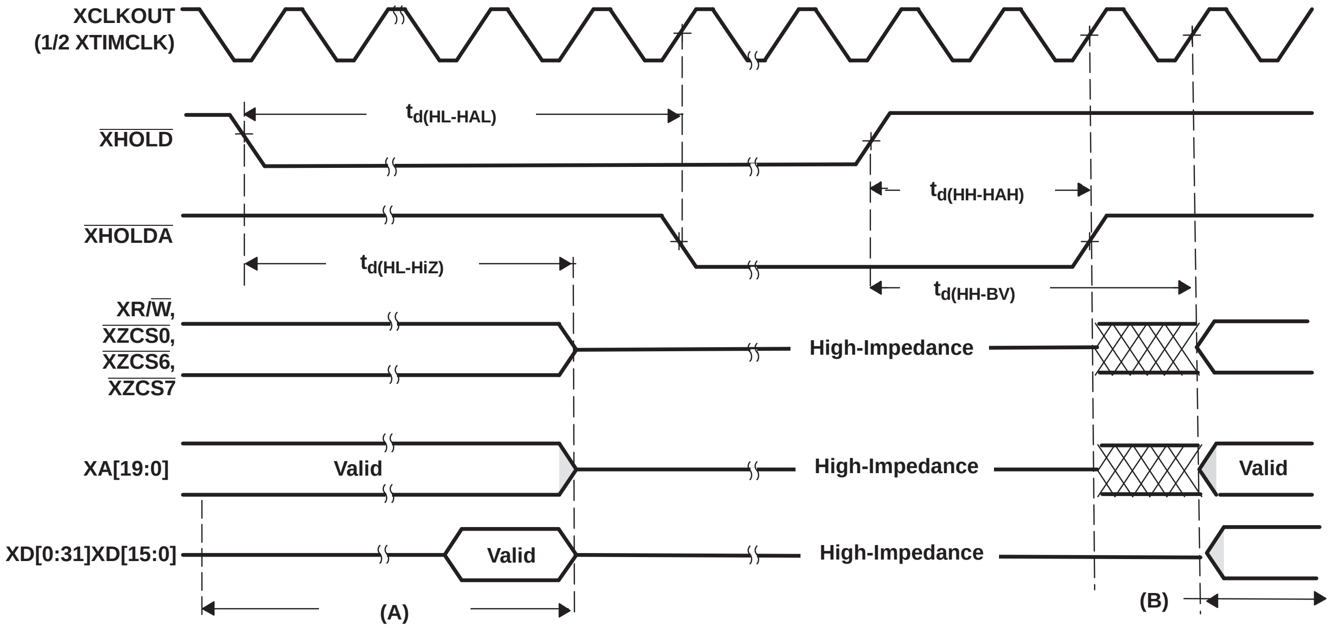
B. Normal XINTF operation resumes.

Figure 6-29. External Interface Hold Waveform

Table 6-50. $\overline{\text{XHOLD}}$/$\overline{\text{XHOLDA}}$ Timing Requirements (XCLKOUT = 1/2 XTIMCLK)(1)(2)(3)

		MIN	MAX	UNIT
$t_{d(HL-HiZ)}$	Delay time, $\overline{\text{XHOLD}}$ low to Hi-Z on all address, data, and control		$4t_{c(XTIM)} + t_{c(XCO)}$	ns
$t_{d(HL-HAL)}$	Delay time, $\overline{\text{XHOLD}}$ low to $\overline{\text{XHOLDA}}$ low		$4t_{c(XTIM + 2tc(XCO))}$	ns
$t_{d(HH-HAH)}$	Delay time, $\overline{\text{XHOLD}}$ high to $\overline{\text{XHOLDA}}$ high		$4t_{c(XTIM)}$	ns
$t_{d(HH-BV)}$	Delay time, $\overline{\text{XHOLD}}$ high to bus valid		$6t_{c(XTIM)}$	ns

(1) When a low signal is detected on $\overline{\text{XHOLD}}$, all pending XINTF accesses will be completed before the bus is placed in a high-impedance state.
(2) The state of $\overline{\text{XHOLD}}$ is latched on the rising edge of XTIMCLK.
(3) After the $\overline{\text{XHOLD}}$ is detected low or high, all bus transitions and $\overline{\text{XHOLDA}}$ transitions occur with respect to the rising edge of XCLKOUT. Thus, for this mode where XCLKOUT = 1/2 XTIMCLK, the transitions can occur up to 1 XTIMCLK cycle earlier than the maximum value specified.



A. All pending XINTF accesses are completed.

B. Normal XINTF operation resumes.

Figure 6-30. $\overline{\text{XHOLD}}$/$\overline{\text{XHOLDA}}$ Timing Requirements (XCLKOUT = 1/2 XTIMCLK)

ADVANCE INFORMATION

6.10.9 On-Chip Analog-to-Digital Converter

Table 6-51. ADC Electrical Characteristics (over recommended operating conditions)(1)(2)

PARAMETER		MIN	TYP	MAX	UNIT
DC SPECIFICATIONS(3)					
Resolution		12			Bits
ADC clock		0.001		25	MHz
ACCURACY					
INL (Integral nonlinearity)	1-12.5 MHz ADC clock (6.25 MSPS)			±1.5	LSB
	12.5-25 MHz ADC clock (12.5 MSPS)			±2	LSB
DNL (Differential nonlinearity)(4)				±1	LSB
Offset error (5)(3)			±15		LSB
Offset error with hardware trimming			±4		LSB
Overall gain error with internal reference (6)(3)			±30		LSB
Overall gain error with external reference (3)			±30		LSB
Channel-to-channel offset variation			±4		LSB
Channel-to-channel gain variation			±4		LSB
ANALOG INPUT					
Analog input voltage (ADCINx to ADCLO) (7)		0		3	V
ADCLO		–5	0	5	mV
Input capacitance			10		pF
Input leakage current				±5	μA
INTERNAL VOLTAGE REFERENCE (6)					
$V_{ADCREFP}$ - ADCREFP output voltage at the pin based on internal reference			1.275		V
$V_{ADCREFM}$ - ADCREFM output voltage at the pin based on internal reference			0.525		V
Voltage difference, ADCREFP - ADCREFM			0.75		V
Temperature coefficient			50		PPM/°C
EXTERNAL VOLTAGE REFERENCE(6) (8)					
$V_{ADCREFIN}$ - External reference voltage input on ADCREFIN pin 0.2% or better accurate reference recommended	ADCREFSEL[15:14] = 11b		1.024		V
	ADCREFSEL[15:14] = 10b		1.500		V
	ADCREFSEL[15:14] = 01b		2.048		V
AC SPECIFICATIONS					
SINAD (100 kHz) Signal-to-noise ratio + distortion			67.5		dB
SNR (100 kHz) Signal-to-noise ratio			68		dB
THD (100 kHz) Total harmonic distortion			–79		dB
ENOB (100 kHz) Effective number of bits			10.9		Bits
SFDR (100 kHz) Spurious free dynamic range			83		dB

(1) Tested at 25 MHz ADCCLK.

(2) All voltages listed in this table are with respect to V_{SSA2}.

(3) ADC parameters for gain error and offset error are only specified if the ADC calibration routine is executed from the Boot ROM. See Section 4.7.3 for more information.

(4) TI specifies that the ADC will have no missing codes.

(5) 1 LSB has the weighted value of 3.0/4096 = 0.732 mV.

(6) A single internal/external band gap reference sources both ADCREFP and ADCREFM signals, and hence, these voltages track together. The ADC converter uses the difference between these two as its reference. The total gain error listed for the internal reference is inclusive of the movement of the internal bandgap over temperature. Gain error over temperature for the external reference option will depend on the temperature profile of the source used.

(7) Voltages above V_{DDA} + 0.3 V or below V_{SS} - 0.3 V applied to an analog input pin may temporarily affect the conversion of another pin. To avoid this, the analog inputs should be kept within these limits.

(8) TI recommends using high precision external reference TI part REF3020/3120 or equivalent for 2.048-V reference.

6.10.9.1 ADC Power-Up Control Bit Timing

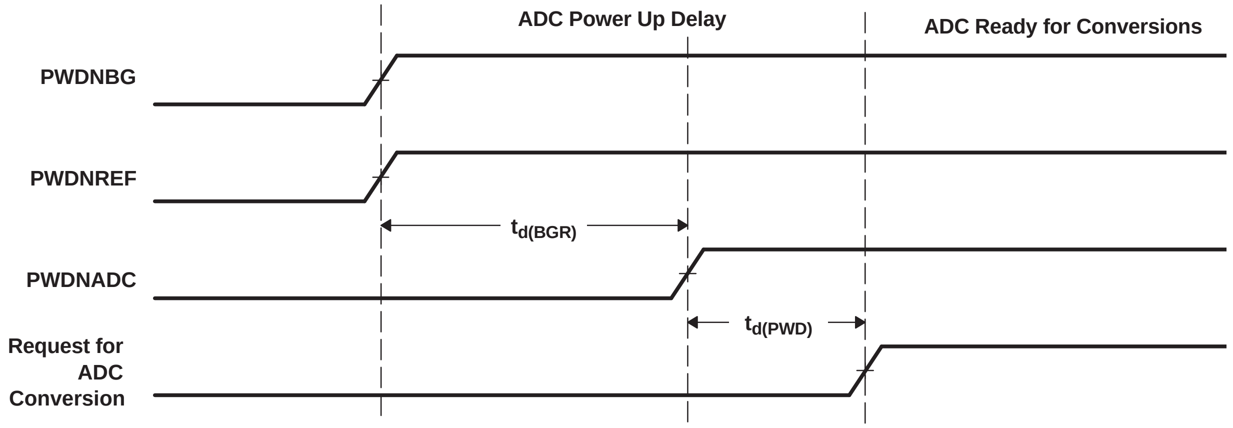


Figure 6-31. ADC Power-Up Control Bit Timing

Table 6-52. ADC Power-Up Delays

PARAMETER[1]		MIN	TYP	MAX	UNIT
$t_{d(BGR)}$	Delay time for band gap reference to be stable. Bits 7 and 6 of the ADCTRL3 register (ADCBGRFDN1/0) must be set to 1 before the PWDNADC bit is enabled.			5	ms
$t_{d(PWD)}$	Delay time for power-down control to be stable. Bit delay time for band-gap reference to be stable. Bits 7 and 6 of the ADCTRL3 register (ADCBGRFDN1/0) must be set to 1 before the PWDNADC bit is enabled. Bit 5 of the ADCTRL3 register (PWDNADC)must be set to 1 before any ADC conversions are initiated.	20	50		μs
				1	ms

(1) Timings maintain compatibility to the 281x ADC module. The F2833x ADC also supports driving all 3 bits at the same time and waiting $t_{d(BGR)}$ ms before first conversion.

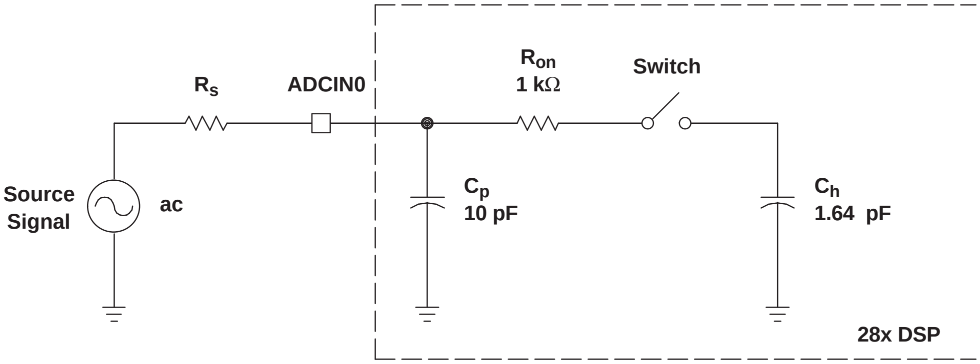
Table 6-53. Current Consumption for Different ADC Configurations (at 25-MHz ADCCLK)[1][2]

ADC OPERATING MODE	CONDITIONS	V_{DDA18}	$V_{DDA3.3}$	UNIT
Mode A (Operational Mode):	• BG and REF enabled • PWD disabled	30	2	mA
Mode B:	• ADC clock enabled • BG and REF enabled • PWD enabled	9	0.5	mA
Mode C:	• ADC clock enabled • BG and REF disabled • PWD enabled	5	20	μA
Mode D:	• ADC clock disabled • BG and REF disabled • PWD enabled	5	15	μA

(1) Test Conditions:
SYSCLKOUT = 150 MHz
ADC module clock = 25 MHz
ADC performing a continuous conversion of all 16 channels in Mode A

(2) V_{DDA18} includes current into V_{DD1A18} and V_{DD2A18}. $V_{DDA3.3}$ includes current into V_{DDA2} and V_{DDAIO}.

ADVANCE INFORMATION



Typical Values of the Input Circuit Components:

Switch Resistance (R_{on}): 1 kΩ
Sampling Capacitor (C_h): 1.64 pF
Parasitic Capacitance (C_p): 10 pF
Source Resistance (R_s): 50 Ω

Figure 6-32. ADC Analog Input Impedance Model

6.10.9.2 Definitions

Reference Voltage

The on-chip ADC has a built-in reference, which provides the reference voltages for the ADC.

Analog Inputs

The on-chip ADC consists of 16 analog inputs, which are sampled either one at a time or two channels at a time. These inputs are software-selectable.

Converter

The on-chip ADC uses a 12-bit four-stage pipeline architecture, which achieves a high sample rate with low power consumption.

Conversion Modes

The conversion can be performed in two different conversion modes:

- Sequential sampling mode (SMODE = 0)
- Simultaneous sampling mode (SMODE = 1)

6.10.9.3 Sequential Sampling Mode (Single-Channel) (SMODE = 0)

In sequential sampling mode, the ADC can continuously convert input signals on any of the channels (Ax to Bx). The ADC can start conversions on event triggers from the ePWM, software trigger, or from an external ADCSOC signal. If the SMODE bit is 0, the ADC will do conversions on the selected channel on every Sample/Hold pulse. The conversion time and latency of the Result register update are explained below. The ADC interrupt flags are set a few SYSCLKOUT cycles after the Result register update. The selected channels will be sampled at every falling edge of the Sample/Hold pulse. The Sample/Hold pulse width can be programmed to be 1 ADC clock wide (minimum) or 16 ADC clocks wide (maximum).

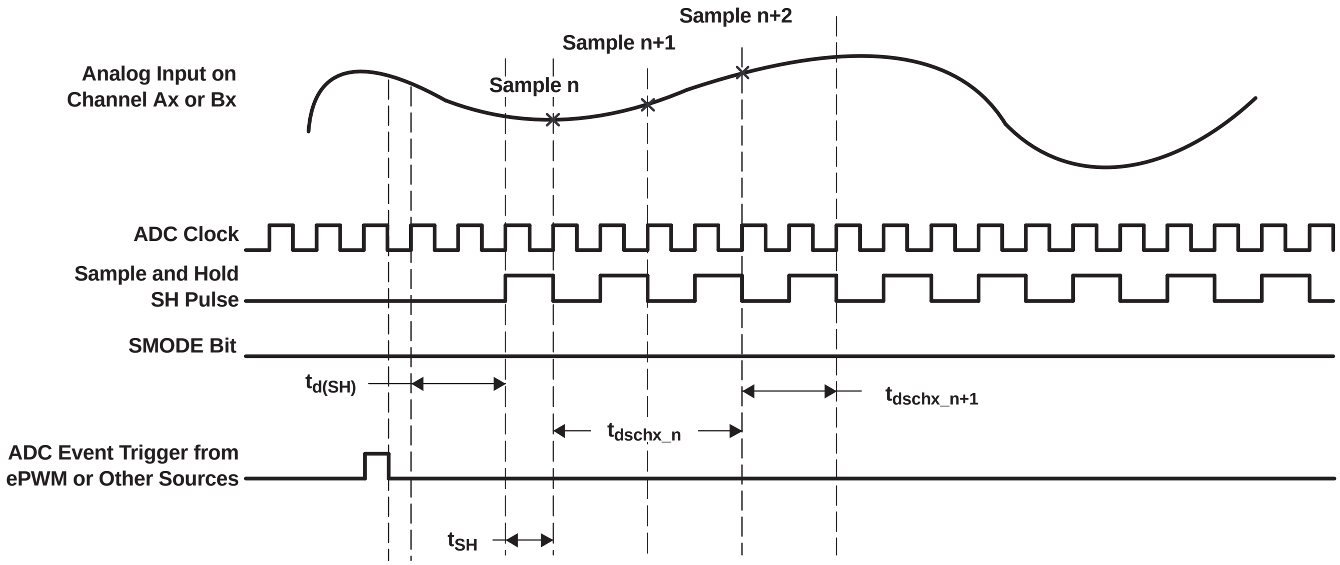


Figure 6-33. Sequential Sampling Mode (Single-Channel) Timing

Table 6-54. Sequential Sampling Mode Timing

		SAMPLE n	SAMPLE n + 1	AT 25 MHz ADC CLOCK, $t_{c(ADCCLK)}$ = 40 ns	REMARKS
$t_{d(SH)}$	Delay time from event trigger to sampling	$2.5t_{c(ADCCLK)}$			
t_{SH}	Sample/Hold width/Acquisition Width	(1 + Acqps) * $t_{c(ADCCLK)}$		40 ns with Acqps = 0	Acqps value = 0-15 ADCTRL1[8:11]
$t_{d(schx\_n)}$	Delay time for first result to appear in Result register	$4t_{c(ADCCLK)}$		160 ns	
$t_{d(schx\_n+1)}$	Delay time for successive results to appear in Result register		(2 + Acqps) * $t_{c(ADCCLK)}$	80 ns	

ADVANCE INFORMATION

6.10.9.4 Simultaneous Sampling Mode (Dual-Channel) (SMODE = 1)

In simultaneous mode, the ADC can continuously convert input signals on any one pair of channels (A0/B0 to A7/B7). The ADC can start conversions on event triggers from the ePWM, software trigger, or from an external ADCSOC signal. If the SMODE bit is 1, the ADC will do conversions on two selected channels on every Sample/Hold pulse. The conversion time and latency of the result register update are explained below. The ADC interrupt flags are set a few SYSCLKOUT cycles after the Result register update. The selected channels will be sampled simultaneously at the falling edge of the Sample/Hold pulse. The Sample/Hold pulse width can be programmed to be 1 ADC clock wide (minimum) or 16 ADC clocks wide (maximum).

NOTE

In simultaneous mode, the ADCIN channel pair select has to be A0/B0, A1/B1, ..., A7/B7, and not in other combinations (such as A1/B3, etc.).

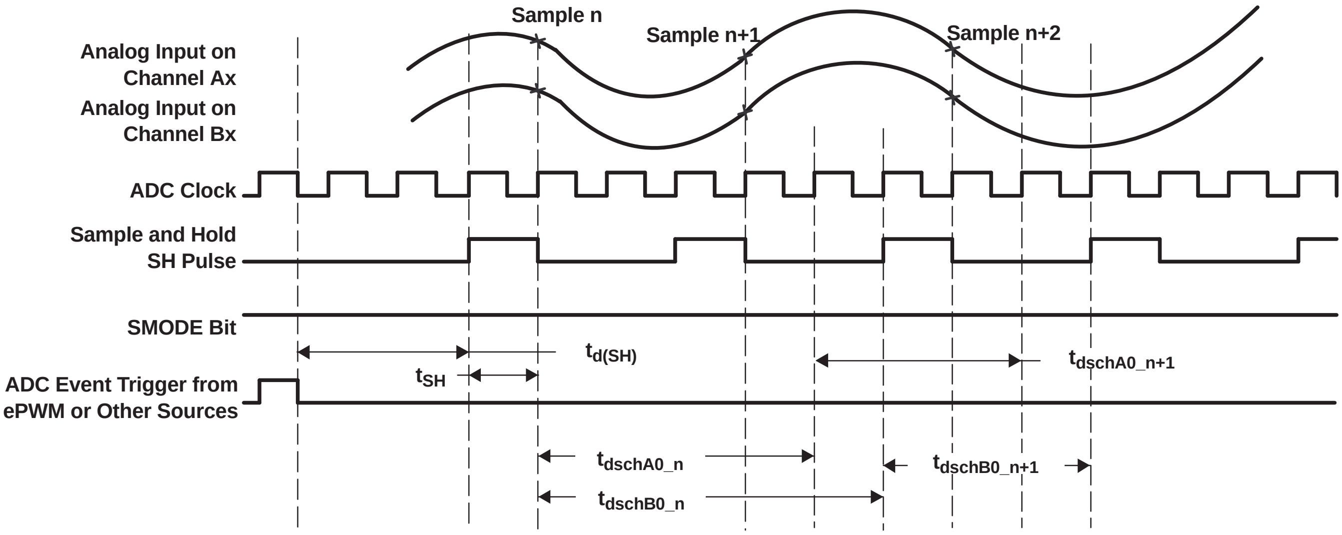


Figure 6-34. Simultaneous Sampling Mode Timing

Table 6-55. Simultaneous Sampling Mode Timing

		SAMPLE n	SAMPLE n + 1	AT 25 MHz ADC CLOCK, $t_{c(ADCCLK)}$ = 40 ns	REMARKS
$t_{d(SH)}$	Delay time from event trigger to sampling	$2.5t_{c(ADCCLK)}$			
t_{SH}	Sample/Hold width/Acquisition Width	(1 + Acqps) * $t_{c(ADCCLK)}$		40 ns with Acqps = 0	Acqps value = 0-15 ADCTRL1[8:11]
$t_{d(schA0\_n)}$	Delay time for first result to appear in Result register	$4t_{c(ADCCLK)}$		160 ns	
$t_{d(schB0\_n)}$	Delay time for first result to appear in Result register	$5t_{c(ADCCLK)}$		200 ns	
$t_{d(schA0\_n+1)}$	Delay time for successive results to appear in Result register		(3 + Acqps) * $t_{c(ADCCLK)}$	120 ns	
$t_{d(schB0\_n+1)}$	Delay time for successive results to appear in Result register		(3 + Acqps) * $t_{c(ADCCLK)}$	120 ns	

6.10.10 Detailed Descriptions

Integral Nonlinearity

Integral nonlinearity refers to the deviation of each individual code from a line drawn from zero through full scale. The point used as zero occurs one-half LSB before the first code transition. The full-scale point is defined as level one-half LSB beyond the last code transition. The deviation is measured from the center of each particular code to the true straight line between these two points.

Differential Nonlinearity

An ideal ADC exhibits code transitions that are exactly 1 LSB apart. DNL is the deviation from this ideal value. A differential nonlinearity error of less than ±1 LSB ensures no missing codes.

Zero Offset

The major carry transition should occur when the analog input is at zero volts. Zero error is defined as the deviation of the actual transition from that point.

Gain Error

The first code transition should occur at an analog value one-half LSB above negative full scale. The last transition should occur at an analog value one and one-half LSB below the nominal full scale. Gain error is the deviation of the actual difference between first and last code transitions and the ideal difference between first and last code transitions.

Signal-to-Noise Ratio + Distortion (SINAD)

SINAD is the ratio of the rms value of the measured input signal to the rms sum of all other spectral components below the Nyquist frequency, including harmonics but excluding dc. The value for SINAD is expressed in decibels.

Effective Number of Bits (ENOB)

For a sine wave, SINAD can be expressed in terms of the number of bits. Using the following formula, $N = \frac{(SINAD - 1.76)}{6.02}$ it is possible to get a measure of performance expressed as N, the effective number of bits. Thus, effective number of bits for a device for sine wave inputs at a given input frequency can be calculated directly from its measured SINAD.

Total Harmonic Distortion (THD)

THD is the ratio of the rms sum of the first nine harmonic components to the rms value of the measured input signal and is expressed as a percentage or in decibels.

Spurious Free Dynamic Range (SFDR)

SFDR is the difference in dB between the rms amplitude of the input signal and the peak spurious signal.

6.10.11 *Multichannel Buffered Serial Port (McBSP) Timing*

6.10.11.0.1 McBSP Transmit and Receive Timing

Table 6-56. McBSP Timing Requirements(1)(2)

NO.				MIN	MAX	UNIT
		McBSP module clock (CLKG, CLKX, CLKR) range		1		kHz
					20(3)	MHz
		McBSP module cycle time (CLKG, CLKX, CLKR) range		50		ns
					1	ms
M11	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X ext	2P		ns
M12	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X ext	P – 7		ns
M13	$t_{r(CKRX)}$	Rise time, CLKR/X	CLKR/X ext		7	ns
M14	$t_{f(CKRX)}$	Fall time, CLKR/X	CLKR/X ext		7	ns
M15	$t_{su(FRH-CKRL)}$	Setup time, external FSR high before CLKR low	CLKR int	18		ns
			CLKR ext	2		
M16	$t_{h(CKRL-FRH)}$	Hold time, external FSR high after CLKR low	CLKR int	0		ns
			CLKR ext	6		
M17	$t_{su(DRV-CKRL)}$	Setup time, DR valid before CLKR low	CLKR int	18		ns
			CLKR ext	2		
M18	$t_{h(CKRL-DRV)}$	Hold time, DR valid after CLKR low	CLKR int	0		ns
			CLKR ext	6		
M19	$t_{su(FXH-CKXL)}$	Setup time, external FSX high before CLKX low	CLKX int	18		ns
			CLKX ext	2		
M20	$t_{h(CKXL-FXH)}$	Hold time, external FSX high after CLKX low	CLKX int	0		ns
			CLKX ext	6		

(1) Polarity bits CLKRP = CLKXP = FSRP = FSXP = 0. If the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

(2) 2P = 1/CLKG in ns. CLKG is the output of sample rate generator mux. CLKG = $\frac{\text{CLKSRG}}{(1 + \text{CLKGDV})}$. CLKSRG can be LSPCLK, CLKX, CLKR as source. CLKSRG ≤ (SYSCLKOUT/2). McBSP performance is limited by I/O buffer switching speed.

(3) Internal clock prescalers must be adjusted such that the McBSP clock (CLKG, CLKX, CLKR) speeds are not greater than the I/O buffer speed limit (20 MHz).

Table 6-57. McBSP Switching Characteristics(1)(2)

NO.	PARAMETER			MIN	MAX	UNIT
M1	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X int	2P		ns
M2	$t_{w(CKRXH)}$	Pulse duration, CLKR/X high	CLKR/X int	D-5(3)	D+5(3)	ns
M3	$t_{w(CKRXL)}$	Pulse duration, CLKR/X low	CLKR/X int	C-5(3)	C+5(3)	ns
M4	$t_{d(CKRH-FRV)}$	Delay time, CLKR high to internal FSR valid	CLKR int	0	4	ns
			CLKR ext	3	27	
M5	$t_{d(CKXH-FXV)}$	Delay time, CLKX high to internal FSX valid	CLKX int	0	4	ns
			CLKX ext	3	27	
M6	$t_{dis(CKXH-DXHZ)}$	Disable time, CLKX high to DX high impedance following last data bit	CLKX int		8	ns
			CLKX ext		14	

(1) Polarity bits CLKRP = CLKXP = FSRP = FSXP = 0. If the polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

(2) 2P = 1/CLKG in ns.

(3) C=CLKRX low pulse width = P
D=CLKRX high pulse width = P

Table 6-57. McBSP Switching Characteristics (continued)

NO.	PARAMETER				MIN	MAX	UNIT
M7	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid. This applies to all bits except the first bit transmitted.		CLKX int		9	ns
				CLKX ext		28	
		Delay time, CLKX high to DX valid	DXENA = 0	CLKX int		8	
				CLKX ext		14	
		Only applies to first bit transmitted when in Data Delay 1 or 2 (XDATDLY=01b or 10b) modes	DXENA = 1	CLKX int		P + 8	
				CLKX ext		P + 14	
M8	$t_{en(CKXH-DX)}$	Enable time, CLKX high to DX driven	DXENA = 0	CLKX int	0		ns
				CLKX ext	6		
		Only applies to first bit transmitted when in Data Delay 1 or 2 (XDATDLY=01b or 10b) modes	DXENA = 1	CLKX int	P		
				CLKX ext	P + 6		
M9	$t_{d(FXH-DXV)}$	Delay time, FSX high to DX valid	DXENA = 0	FSX int		8	ns
				FSX ext		14	
		Only applies to first bit transmitted when in Data Delay 0 (XDATDLY=00b) mode.	DXENA = 1	FSX int		P + 8	
				FSX ext		P + 14	
M10	$t_{en(FXH-DX)}$	Enable time, FSX high to DX driven	DXENA = 0	FSX int	0		ns
				FSX ext	6		
		Only applies to first bit transmitted when in Data Delay 0 (XDATDLY=00b) mode	DXENA = 1	FSX int	P		
				FSX ext	P + 6		

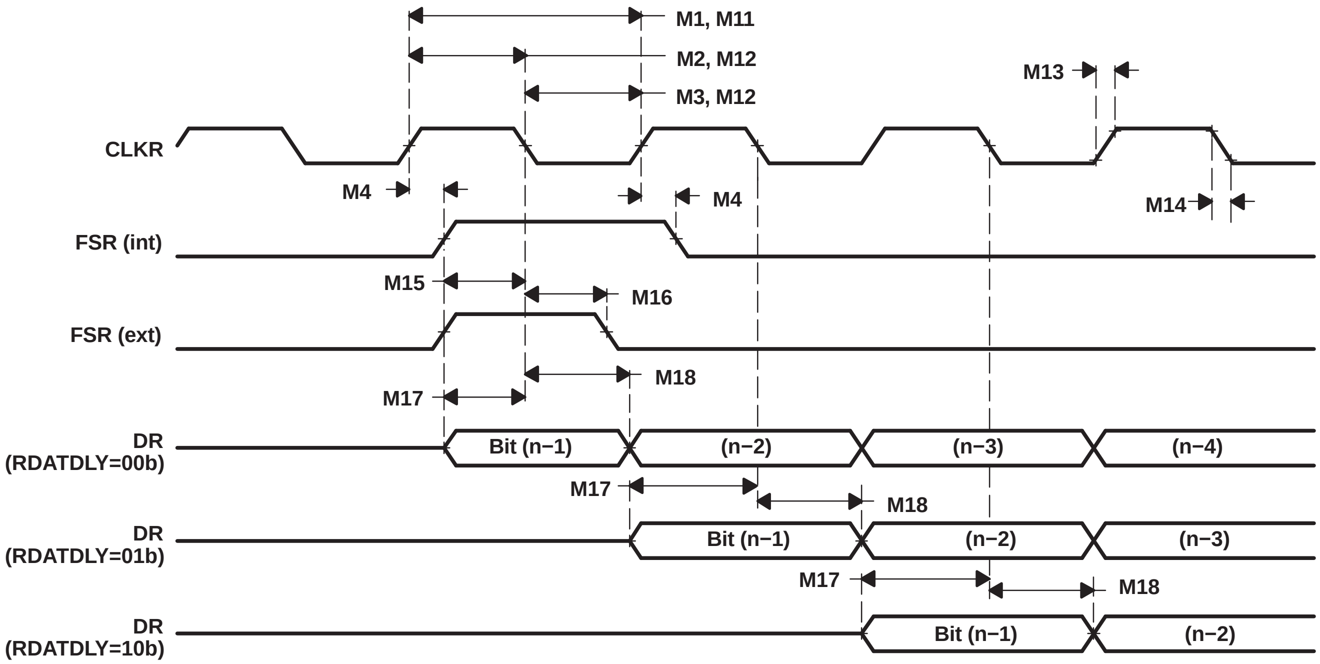


Figure 6-35. McBSP Receive Timing

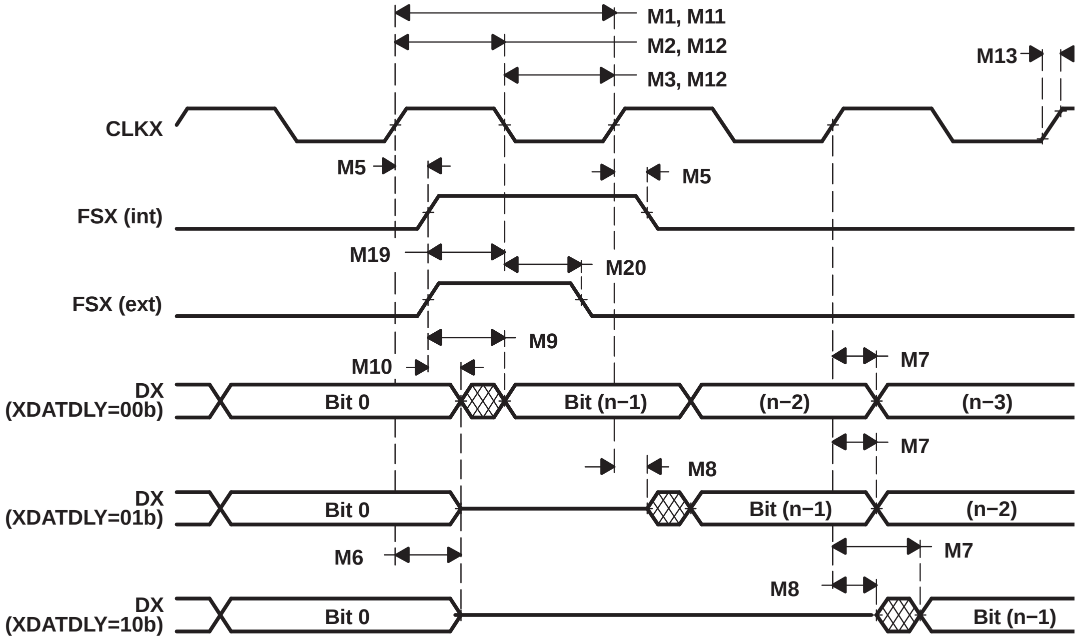


Figure 6-36. McBSP Transmit Timing

6.10.11.0.2 McBSP as SPI Master or Slave Timing

Table 6-58. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 10b, CLKXP = 0)

NO.			MASTER		SLAVE		UNIT
			MIN	MAX	MIN	MAX	
M30	$t_{su(DRV-CKXL)}$	Setup time, DR valid before CLKX low	30		8P – 10		ns
M31	$t_{h(CKXL-DRV)}$	Hold time, DR valid after CLKX low	1		8P –10		ns
M32	$t_{su(BFXL-CKXH)}$	Setup time, FSX low before CLKX high			8P + 10		ns
M33	$t_{c(CKX)}$	Cycle time, CLKX	2P (1)		16P		ns

(1) 2P = 1/CLKG

Table 6-59. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 10b, CLKXP = 0)

NO.	PARAMETER		MASTER		SLAVE		UNIT
			MIN	MAX	MIN	MAX	
M24	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low	2P (1)				ns
M25	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high	P				ns
M28	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high	6		6P + 6		ns
M29	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	6		4P + 6		ns

(1) 2P = 1/CLKG

For all SPI slave modes, CLKX has to be minimum 8 CLKG cycles. Also CLKG should be LSPCLK/2 by setting CLKSM = CLKGDV = 1. With maximum LSPCLK speed of 75 MHz, CLKX maximum frequency will be LSPCLK/16 , that is 4.6875 MHz and P = 13.3 ns.

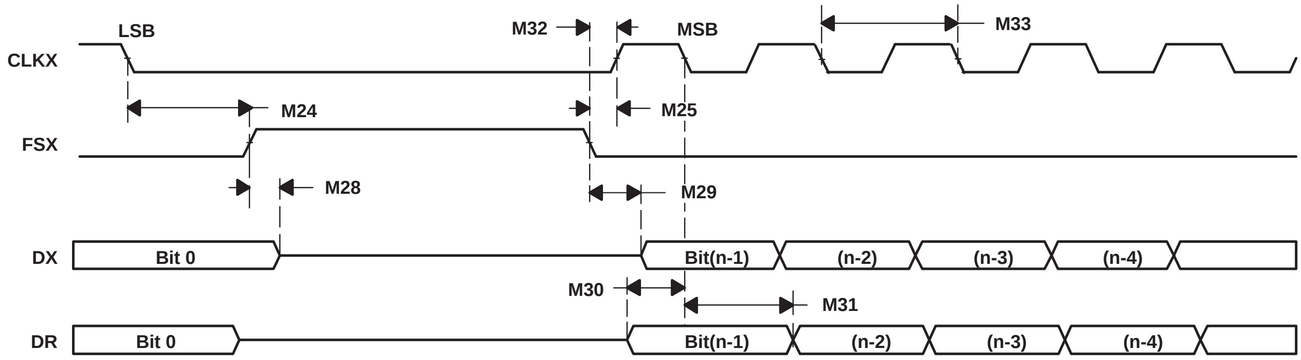


Figure 6-37. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

Table 6-60. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 11b, CLKXP = 0)

NO.			MASTER		SLAVE		UNIT
			MIN	MAX	MIN	MAX	
M39	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	30		8P – 10		ns
M40	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	1		8P – 10		ns
M41	$t_{su(FXL-CKXH)}$	Setup time, FSX low before CLKX high			16P + 10		ns
M42	$t_{c(CKX)}$	Cycle time, CLKX	2P(1)		16P		ns

(1) 2P = 1/CLKG

Table 6-61. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 11b, CLKXP = 0)

NO.	PARAMETER		MASTER		SLAVE		UNIT
			MIN	MAX	MIN	MAX	
M34	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low	P				ns
M35	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high	2P(1)				ns
M37	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low	P + 6		7P + 6		ns
M38	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	6		4P + 6		ns

(1) 2P = 1/CLKG

For all SPI slave modes, CLKX must be a minimum of 8 CLKG cycles. Also, CLKG should be LSPCLK/2 by setting CLKSM = CLKGDV = 1. With a maximum LSPCLK speed of 75 MHz, CLKX maximum frequency is LSPCLK/16; that is, 4.6875 MHz and P =13.3 ns.

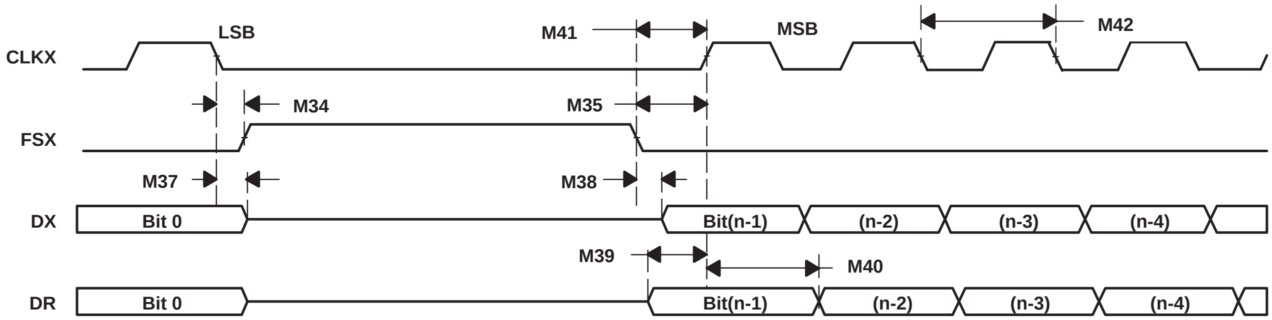


Figure 6-38. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0

ADVANCE INFORMATION

Table 6-62. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 10b, CLKXP = 1)

NO.			MASTER		SLAVE		
			MIN	MAX	MIN	MAX	UNIT
M49	$t_{su(DRV-CKXH)}$	Setup time, DR valid before CLKX high	30		8P –10		ns
M50	$t_{h(CKXH-DRV)}$	Hold time, DR valid after CLKX high	1		8P –10		ns
M51	$t_{su(FXL-CKXL)}$	Setup time, FSX low before CLKX low			8P + 10		ns
M52	$t_{c(CKX)}$	Cycle time, CLKX	2P (1)		16P		ns

(1) 2P = 1/CLKG

Table 6-63. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 10b, CLKXP = 1)

NO.	PARAMETER		MASTER		SLAVE		
			MIN	MAX	MIN	MAX	UNIT
M43	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high	2P (1)				ns
M44	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low	P				ns
M47	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high	6		6P + 6		ns
M48	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	6		4P + 6		ns

(1) 2P = 1/CLKG

For all SPI slave modes, CLKX must be a minimum of 8 CLKG cycles. Also, CLKG should be LSPCLK/2 by setting CLKSM = CLKGDV = 1. With maximum LSPCLK speed of 75 MHz, CLKX maximum frequency will be LSPCLK/16; that is, 4.6875 MHz and P = 13.3 ns.

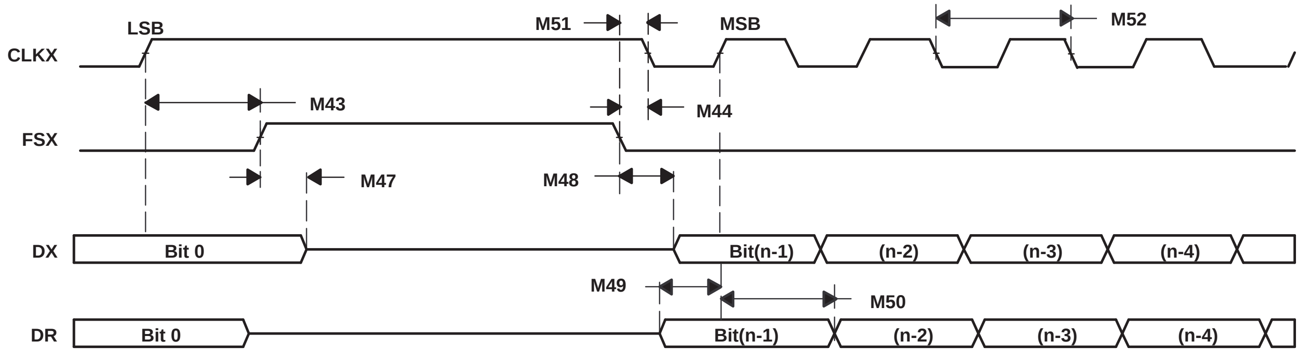


Figure 6-39. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1

Table 6-64. McBSP as SPI Master or Slave Timing Requirements (CLKSTP = 11b, CLKXP = 1)

NO.			MASTER		SLAVE		UNIT
			MIN	MAX	MIN	MAX	
M58	$t_{su(DRV-CKXL)}$	Setup time, DR valid before CLKX low	30		8P – 10		ns
M59	$t_{h(CKXL-DRV)}$	Hold time, DR valid after CLKX low	1		8P – 10		ns
M60	$t_{su(FXL-CKXL)}$	Setup time, FSX low before CLKX low			16P + 10		ns
M61	$t_{c(CKX)}$	Cycle time, CLKX	2P (1)		16P		ns

(1) 2P = 1/CLKG

For all SPI slave modes, CLKX must be a minimum of 8 CLKG cycles. Also CLKG should be LSPCLK/2 by setting CLKSM = CLKGDV = 1. With maximum LSPCLK speed of 75 MHz, CLKX maximum frequency is LSPCLK/16 , that is 4.6875 MHz and P = 13.3 ns.

Table 6-65. McBSP as SPI Master or Slave Switching Characteristics (CLKSTP = 11b, CLKXP = 1)(1)

NO.	PARAMETER		MASTER(2)		SLAVE		UNIT
			MIN	MAX	MIN	MAX	
M53	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high	P				ns
M54	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low	2P(1)				ns
M56	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high	P + 6		7P + 6		ns
M57	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	6		4P + 6		ns

(1) 2P = 1/CLKG
(2) C = CLKX low pulse width = P
D = CLKX high pulse width = P

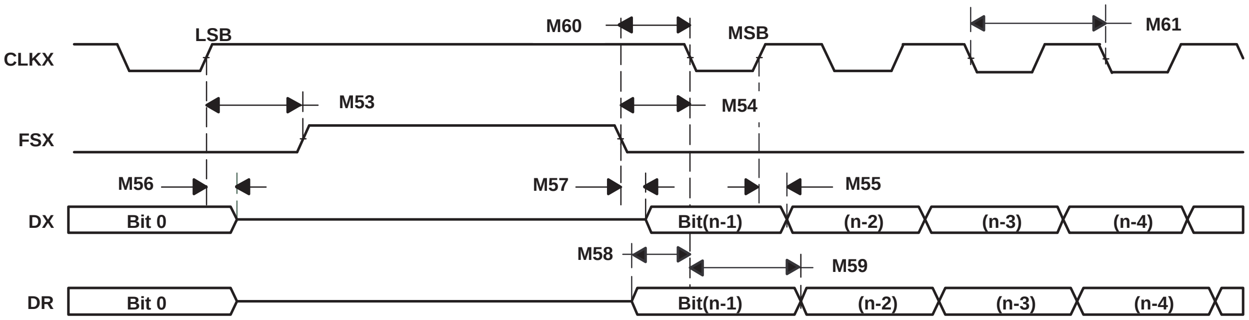


Figure 6-40. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1

ADVANCE INFORMATION

7 Thermal/Mechanical Data

Table 7-1, Table 7-2, and Table 7-3 show the thermal data.

The mechanical package diagram(s) that follow the tables reflect the most current released mechanical data available for the designated device(s).

Table 7-1. F2833x Thermal Model 176-pin PGF Results

AIR FLOW	
PARAMETER	**0 lfm**
θ_{JA}[°C/W] High k PCB	44
Ψ_{JT}[°C/W]	0.1
θ_{JC}	8.2
θ_{JB}	28.1

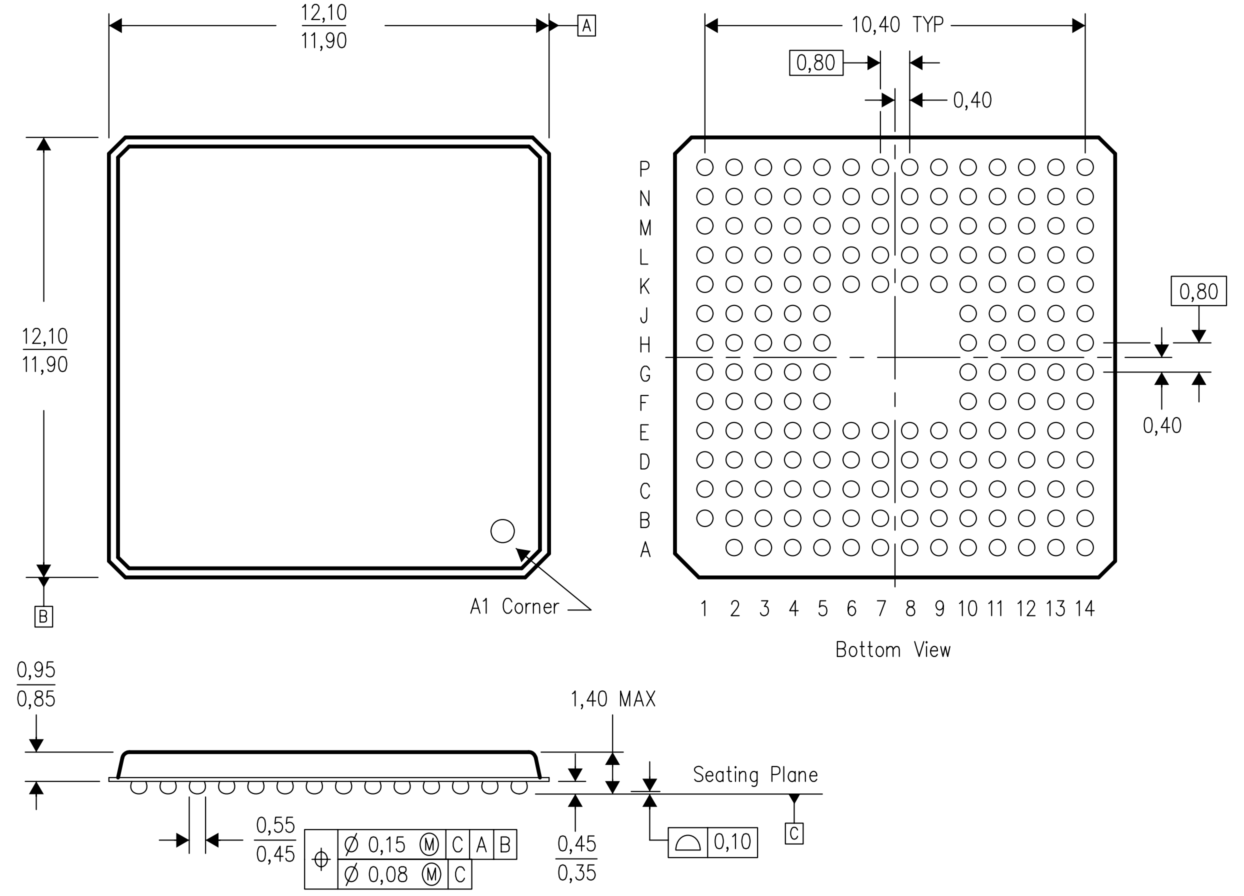
Table 7-2. F2833x Thermal Model 179-pin ZHH Results

AIR FLOW	
PARAMETER	**0 lfm**
θ_{JA}[°C/W] High k PCB	32.8
Ψ_{JT}[°C/W]	0.1
θ_{JC}	8.8
θ_{JB}	12.5

Table 7-3. F2833x Thermal Model 176-pin ZJZ Results

AIR FLOW	
PARAMETER	**0 lfm**
θ_{JA}[°C/W] High k PCB	30.1
Ψ_{JT}[°C/W]	0.115
θ_{JC}	7.29
θ_{JB}	9.99

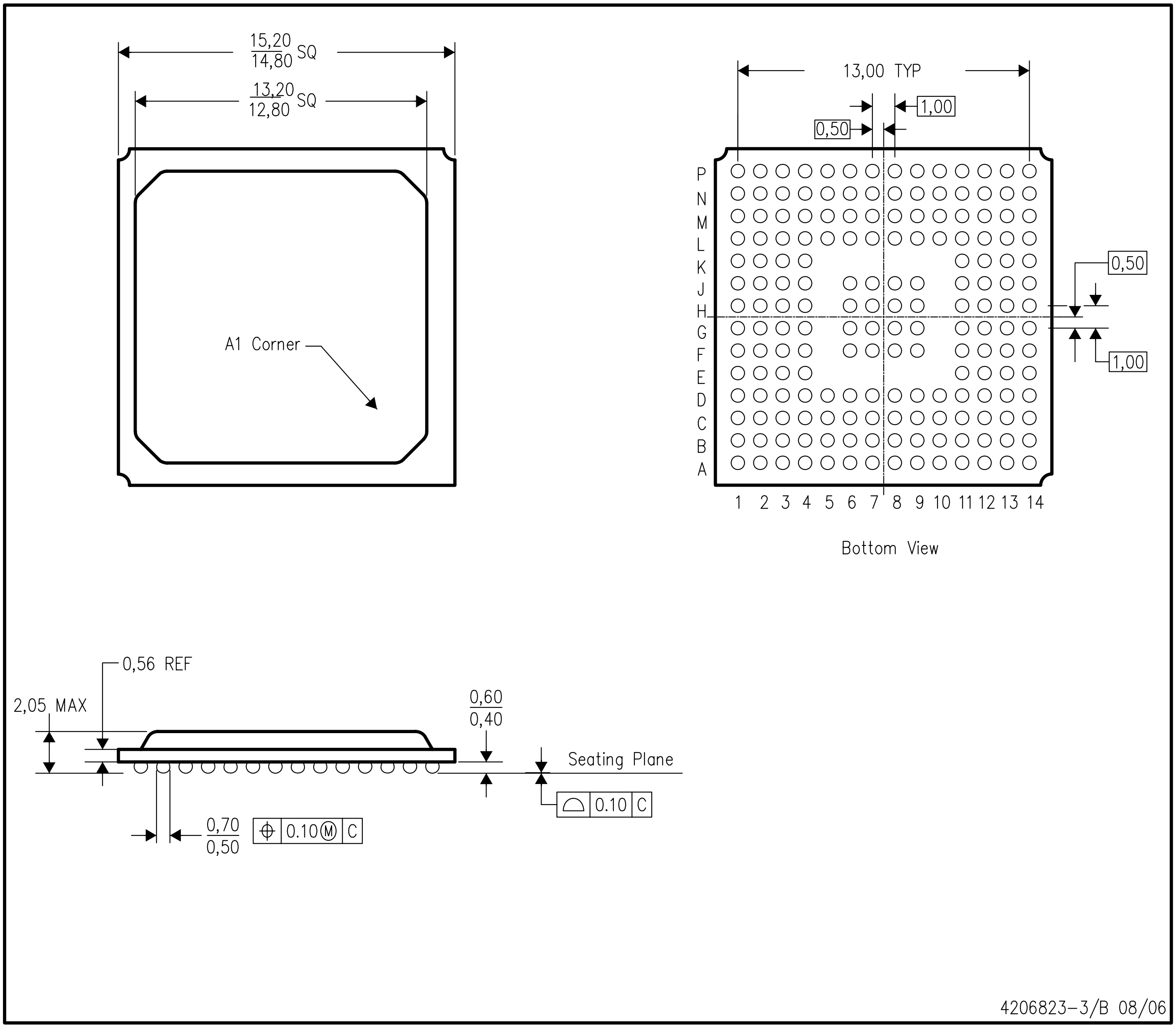
ZHH (S–PBGA–N179) PLASTIC BALL GRID ARRAY



4204739–5/D 07/07

NOTES:
- A. All linear dimensions are in millimeters.
- B. This drawing is subject to change without notice.
- C. Micro Star BGA configuration.
- D. This is a lead–free solder ball design.

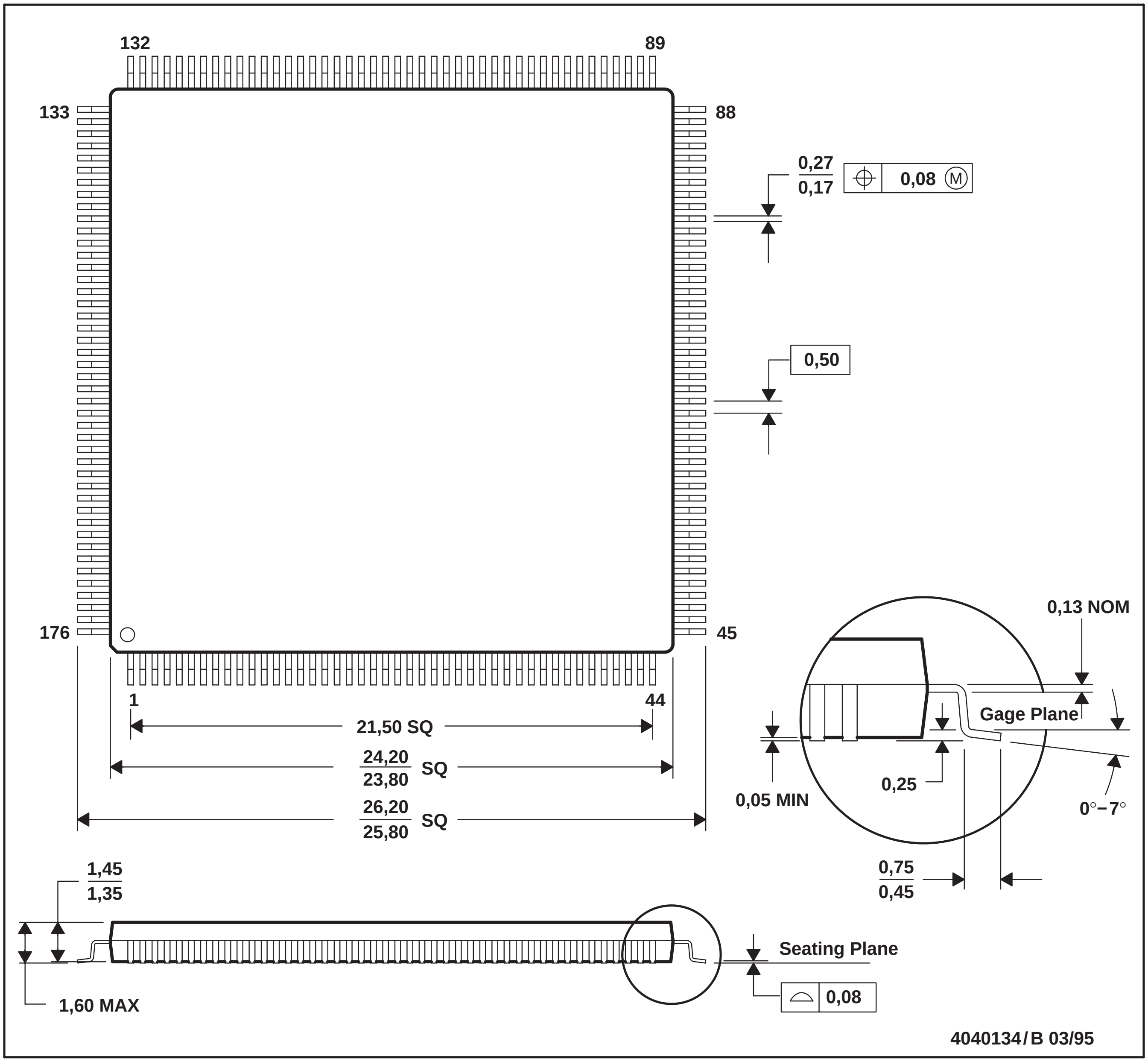
ZJZ (S–PBGA–N176) PLASTIC BALL GRID ARRAY



NOTES:
- A. All linear dimensions are in millimeters.
- B. This drawing is subject to change without notice.
- C. This is a lead–free solder ball design.

OCTOBER 1994

PGF (S-PQFP-G176) **PLASTIC QUAD FLATPACK**



NOTES:
A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MO-136

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI’s standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
Low Power Wireless	www.ti.com/lpw	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2007, Texas Instruments Incorporated