- **High-Performance Static CMOS Technology**
  - **150 MHz (6.67-ns Cycle Time)**
  - **Low-Power (1.8-V Core, 3.3-V I/O) Design**
  - **3.3-V Flash Programming Voltage**
- **High-Performance CPU (C28x)**
  - **16 x 16 and 32 x 32 MAC Operations**
  - **16 x 16 Dual MAC**
  - **Harvard Bus Architecture**
  - **Atomic Operations**
  - **Fast Interrupt Response and Processing**
  - **Unified Memory Programming Model**
  - **4M Linear Program Address Reach**
  - **4G Linear Data Address Reach**
  - **Code-Efficient (in C/C++ and Assembly)**
  - **TMS320F24x/LF240x Processor Source Code Compatible**
- **On-Chip Memory**
  - **Up to 128K x 16 Flash (8 x 4K and 6 x 16K Sectors)**
  - **2K x 16 OTP ROM**
  - **L0 and L1: 2 Blocks of 4K x 16 Single-Access RAM (SARAM)**
  - **H0: 1 Block of 8K x 16 SARAM**
  - **M0 and M1: 2 Blocks of 1K x 16 SARAM**
- **Boot ROM (4K x 16)**
  - **With Software Boot Modes**
  - **Standard Math Tables**
- **External Interface (F2812)**
  - **Up to 1.5M Total Memory**
  - **Programmable Wait States**
  - **Programmable Read/Write Strobe Timing**
  - **Four Individual Chip Selects**
- **Clock and System Control**
  - **Dynamic PLL Ratio Changes Supported**
  - **On-Chip Oscillator**
  - **Watchdog Timer Module**
- **Three External Interrupts**
- **Peripheral Interrupt Expansion (PIE) Block That Supports 45 Peripheral Interrupts**
- **128-Bit Security Key/Lock**
  - **Protects Flash/OTP and L0/L1 SARAM**
  - **Prevents Firmware Reverse Engineering**
- **Motor Control Peripherals**
  - **Two Event Managers (EVA, EVB)**
  - **Compatible to 240x Devices**
- **Three 32-Bit CPU-Timers**
- **Serial Port Peripherals**
  - **Serial Peripheral Interface (SPI)**
  - **Two Serial Communications Interfaces (SCIs), Standard UART**
  - **Enhanced Controller Area Network (eCAN)**
  - **Multichannel Buffered Serial Port (McBSP) With SPI Mode**
- **12-Bit ADC, 16 Channels**
  - **2 x 8 Channel Input Multiplexer**
  - **Two Sample-and-Hold**
  - **Single Conversion Time: 200 ns**
  - **Pipeline Conversion Time: 60 ns**
- **Up to 56 Individually Programmable, Multiplexed General-Purpose Input/Output (GPIO) Pins**
- **Advanced Emulation Features**
  - **Analysis and Breakpoint Functions**
  - **Real-Time Debug via Hardware**
- **Development Tools Include**
  - **ANSI C/C++ Compiler/Assembler/Linker**
  - **Supports TMS320C24x™/240x Instructions**
  - **Code Composer Studio™ IDE**
  - **DSP BIOS**
  - **JTAG Scan Controllers† (TI or Third-Party)**
  - **Evaluation Modules**
  - **Broad Third-Party Digital Motor Control Support**
- **Low-Power Modes and Power Savings**
  - **IDLE, STANDBY, HALT Modes Supported**
  - **Disable Individual Peripheral Clocks**
- **Package Options**
  - **179-Pin MicroStar BGA™ With External Interface (GHH) (F2812)**
  - **176-Pin Low-Profile Quad Flatpack (LQFP) With External Interface (PGF) (F2812)**
  - **128-Pin LQFP Without External Interface (PBK) (F2810)**
- **Temperature Options:**
  - **A: –40°C to 85°C**
  - **S: –40°C to 125°C**

PRODUCT PREVIEW

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

TMS320C24x, Code Composer Studio, and MicroStar BGA are trademarks of Texas Instruments.
All trademarks are the property of their respective owners.
† IEEE Standard 1149.1–1990, IEEE Standard Test-Access Port

PRODUCT PREVIEW

Table of Contents

device summary

Note that throughout this data sheet, F2810 is used to denote TMS320F2810; F2812 is used to denote TMS320F2812; and F28x is used to denote F2810 and F2812.

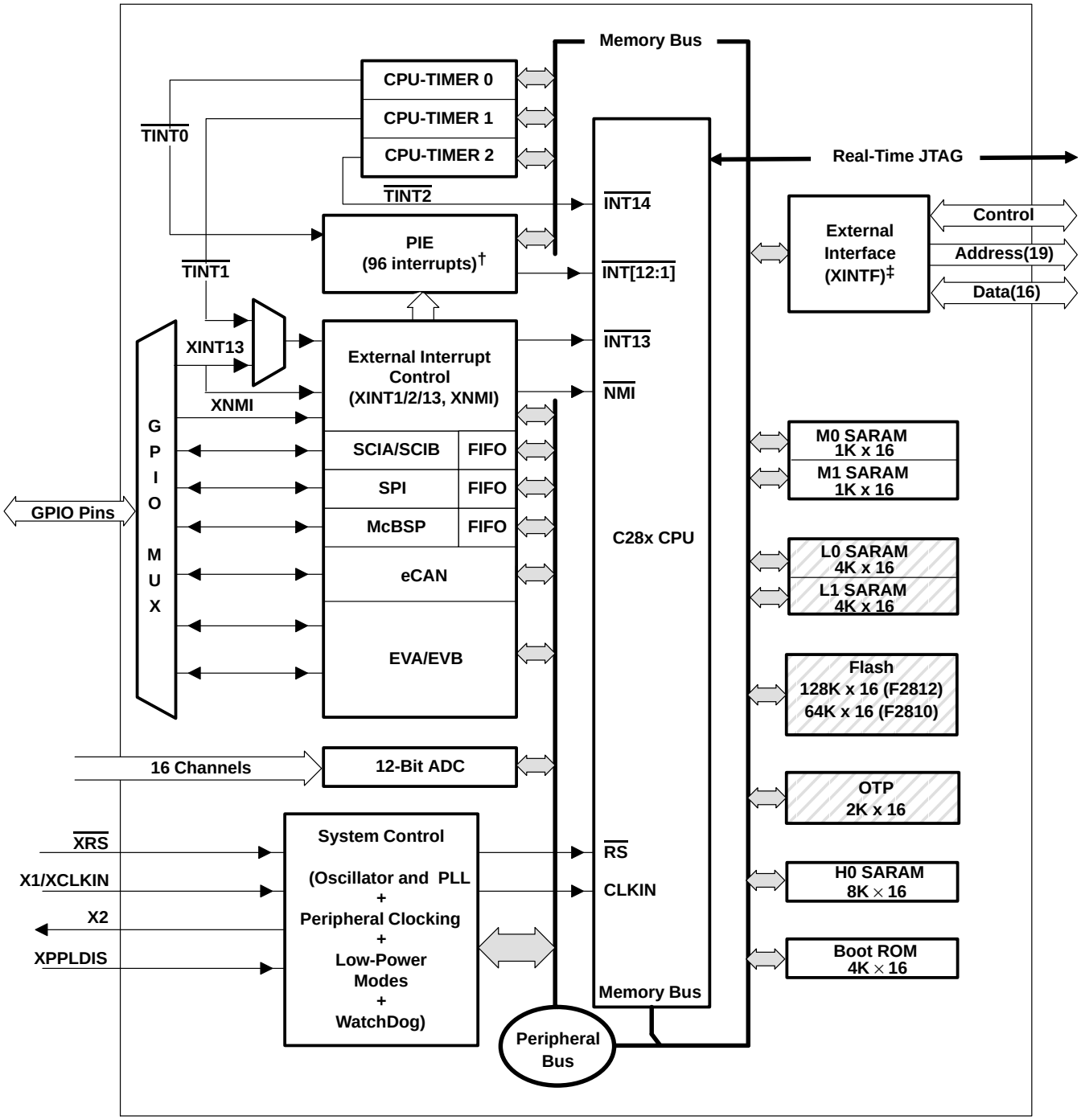
Table 1. Hardware Features of the F2810 and F2812 Devices

FEATURE	F2810	F2812
Instruction Cycle (at 150 MHz)	6.67 ns	6.67 ns
Single-Access RAM (SARAM) (16-bit word)	18K	18K
3.3-V On-Chip Flash (16-bit word)	64K	128K
Code Security for On-Chip Flash/SARAM	Yes	Yes
Boot ROM	Yes	Yes
OTP ROM	Yes	Yes
External Memory Interface	—	Yes
Event Managers A and B (EVA and EVB)	EVA, EVB	EVA, EVB
• General-Purpose (GP) Timers	4	4
• Compare (CMP)/PWM	16	16
• Capture (CAP)/QEP Channels	6/2	6/2
Watchdog Timer	Yes	Yes
12-Bit ADC	Yes	Yes
• Channels	16	16
32-bit CPU Timers	3	3
SPI	Yes	Yes
SCIA, SCIB	SCIA, SCIB	SCIA, SCIB
CAN	Yes	Yes
McBSP	Yes	Yes
Digital I/O Pins (Shared)	56	56
External Interrupts	3	3
Supply Voltage	1.8-V Core, 3.3-V I/O	1.8-V Core, 3.3-V I/O
Packaging	128-pin PBK	179-pin GHH 176-pin PGF
Product Status: Product Preview (PP) Advance Information (AI) Production Data (PD)	PP	PP

PRODUCT PREVIEW

PRODUCT PREVIEW

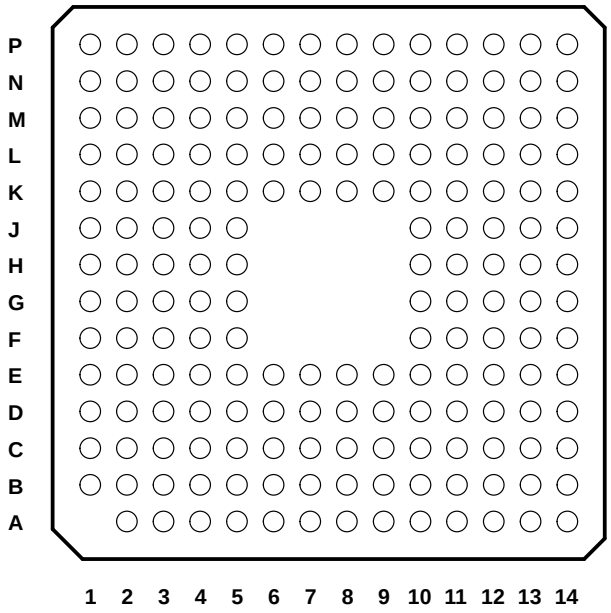
functional block diagram



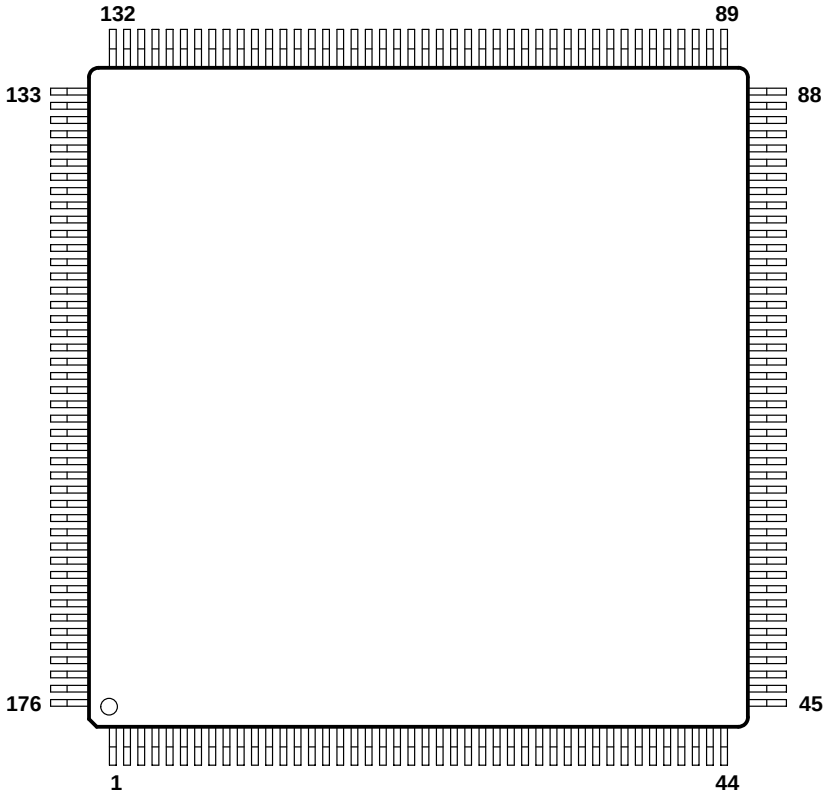
Protected by the Code Security Module.

† 45 of the possible 96 interrupts are used on F2810/F2812.
‡ XINTF is not available on the F2810.

179-Pin GHH
(Ball Grid Array)
(TOP VIEW)

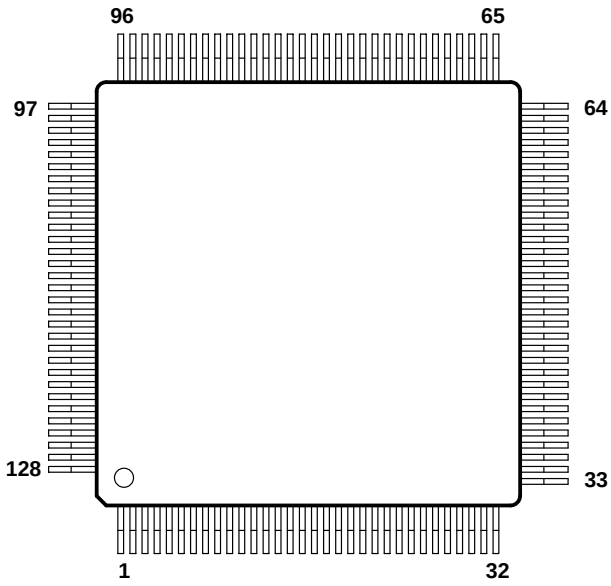


176-Pin PGF
(Low-Profile Quad Flatpack)
(TOP VIEW)



PRODUCT PREVIEW

128-Pin PBK
(Low-Profile Quad Flatpack)
(TOP VIEW)



PRODUCT PREVIEW

pin functions

Table 2 specifies the signals on the F2810 and F2812 devices. All digital inputs are TTL-compatible. All outputs are 3.3 V with CMOS levels. Inputs are not 5-V tolerant. A 20-µA resistor is used for pullup/down.

Table 2. Signal Descriptions

NAME	PIN NO.	I/O/Z	DRIVE	PU/PD†	DESCRIPTION
XINTF SIGNALS (2812 ONLY)					
XA[18:0]		O/Z			19-bit Address Bus
XD[15:0]		I/O/Z			16-bit Data Bus
XMP/$\overline{\text{MC}}$		I		PU	Microprocessor/Microcomputer Mode Select
$\overline{\text{XHOLD}}$		I		PU	External DMA Hold Request
$\overline{\text{XHOLDA}}$		O/Z			External DMA Hold Acknowledge
$\overline{\text{XZCS0}}$		O/Z			Zone 0 Chip Select Strobe
$\overline{\text{XZCS1}}$		O/Z			Zone 1 Chip Select Strobe
$\overline{\text{XZCS2}}$		O/Z			Zone 2 Chip Select Strobe
$\overline{\text{XZCS6AND7}}$		O/Z			Zone 6 and 7 Chip Select Strobe
$\overline{\text{XWE}}$		O/Z			Write Enable
$\overline{\text{XRD}}$		O/Z			Read Enable
XRNW		O/Z			Read Not Write Select
XREADY		I		PU	Input Ready Signal
JTAG AND MISCELLANEOUS SIGNALS					
X1/XCLKIN		I		–	Oscillator Input Or Clock Generator Input
X2		I		–	Oscillator Output
XPPLDIS		I		PU	Disable PLL When High
TESTSEL		I		PU	Test Mode Select Signal
$\overline{\text{XRS}}$		I/O		PU	Device Reset (in) and Watchdog Reset (out)
TEST1		I/O		–	Flash Test Signal 1
TEST2		I/O		–	Flash Test Signal 2
JTAG					
$\overline{\text{TRST}}$		I		PD	JTAG Test-Logic Reset
TCK		I			JTAG Test clock
TMS		I			JTAG Test Mode Select
TDI		I			JTAG Test Data Input
TDO		O/Z			JTAG Test Data Output
EMU0		I/O/Z		PU	Emulation/Test trigger channel 0
EMU1		I/O/Z		PU	Emulation/Test trigger channel 1

† PU = pin has internal pullup; PD = pin has internal pulldown

pin functions (continued)

Table 2. Signal Descriptions (Continued)

NAME	PIN NO.	I/O/Z	DRIVE	PU/PD†	DESCRIPTION
ADC ANALOG INPUT SIGNALS					
ADCIN0[7:0]		I			8 Channel Analog Inputs
ADCIN1[7:0]		I			8 Channel Analog Inputs
ADCREFP		O			ADC Reference Output
ADCREFM		O			ADC Reference Output
ADCRESEXT		O			ADC External Current Bias Resistor
AVSSREFBG		I			Analog GND
AVDDREFBG		I			Analog Power
ADCLO		I			Common Low Side Analog Input
AGND (2 pins)		I			Analog GND
AVDD (2 pins)		I			Analog 3.3-V Supply
POWER SIGNALS					
V_{DDO}					3.3-V I/O Power Pins
V_{SS}					I/O Ground Pins
CV_{DD}					1.8-V CPU/Core Power Pins
CV_{SS}					CPU/Core Ground Pins

† PU = pin has internal pullup; PD = pin has internal pulldown

PRODUCT PREVIEW

pin functions (continued)

Table 2. Signal Descriptions (Continued)

GPIO	PERIPHERAL SIGNAL	PIN NO.	I/O/Z	DRIVE	PU/PD†	DESCRIPTION
GPIO OR PERIPHERAL SIGNALS						
GPIOA OR EVA SIGNALS						
GPIOA0	PWM1 (O)		I/O/Z		PU	GPIO or PWM Output Pin #1
GPIOA1	PWM2 (O)		I/O/Z		PU	GPIO or PWM Output Pin #2
GPIOA2	PWM3 (O)		I/O/Z		PU	GPIO or PWM Output Pin #3
GPIOA3	PWM4 (O)		I/O/Z		PU	GPIO or PWM Output Pin #4
GPIOA4	PWM5 (O)		I/O/Z		PU	GPIO or PWM Output Pin #5
GPIOA5	PWM6 (O)		I/O/Z		PU	GPIO or PWM Output Pin #6
GPIOA6	T1PWM_T1CMP (I)		I/O/Z		PU	GPIO or Timer 1 Output
GPIOA7	T2PWM_T2CMP (I)		I/O/Z		PU	GPIO or Timer 2 Output
GPIOA8	CAP1_QEP1 (I)		I/O/Z		PU	GPIO or Capture Input #1
GPIOA9	CAP2_QEP2 (I)		I/O/Z		PU	GPIO or Capture Input #2
GPIOA10	CAP3_QEPI1 (I)		I/O/Z		PU	GPIO or Capture Input #3
GPIOA11	TDIRA (I)		I/O/Z		PU	GPIO or Timer Direction
GPIOA12	TCLKINA (I)		I/O/Z		PU	GPIO or Timer Clock Input
GPIOA13	$\overline{\text{C1TRIP}}$ (I)		I/O/Z		PU	GPIO or Compare 1 Output Trip
GPIOA14	$\overline{\text{C2TRIP}}$ (I)		I/O/Z		PU	GPIO or Compare 2 Output Trip
GPIOA15	$\overline{\text{C3TRIP}}$ (I)		I/O/Z		PU	GPIO or Compare 3 Output Trip
GPIOB OR EVB SIGNALS						
GPIOB0	PWM7 (O)		I/O/Z		PU	GPIO or PWM Output Pin #7
GPIOB1	PWM8 (O)		I/O/Z		PU	GPIO or PWM Output Pin #8
GPIOB2	PWM9 (O)		I/O/Z		PU	GPIO or PWM Output Pin #9
GPIOB3	PWM10 (O)		I/O/Z		PU	GPIO or PWM Output Pin #10
GPIOB4	PWM11 (O)		I/O/Z		PU	GPIO or PWM Output Pin #11
GPIOB5	PWM12 (O)		I/O/Z		PU	GPIO or PWM Output Pin #12
GPIOB6	T3PWM_T3CMP (I)		I/O/Z		PU	GPIO or Timer 3 Output
GPIOB7	T4PWM_T4CMP (I)		I/O/Z		PU	GPIO or Timer 4 Output
GPIOB8	CAP4_QEP3 (I)		I/O/Z		PU	GPIO or Capture Input #4
GPIOB9	CAP5_QEP4 (I)		I/O/Z		PU	GPIO or Capture Input #5
GPIOB10	CAP6_QEPI2 (I)		I/O/Z		PU	GPIO or Capture Input #6
GPIOB11	TDIRB (I)		I/O/Z		PU	GPIO or Timer Direction
GPIOB12	TCLKINB (I)		I/O/Z		PU	GPIO or Timer Clock Input
GPIOB13	$\overline{\text{C4TRIP}}$ (I)		I/O/Z		PU	GPIO or Compare 4 Output Trip
GPIOB14	$\overline{\text{C5TRIP}}$ (I)		I/O/Z		PU	GPIO or Compare 5 Output Trip
GPIOB15	$\overline{\text{C6TRIP}}$ (I)		I/O/Z		PU	GPIO or Compare 6 Output Trip
GPIOD OR EVA SIGNALS						
GPIOD0	$\overline{\text{T1CTRIP\_PDPINTA}}$ (I)		I/O/Z		PU	Timer 1 Compare Output Trip
GPIOD1	$\overline{\text{T2CTRIP}}$/EVASOC (I)		I/O/Z		PU	Timer 2 Compare Output Trip or External ADC Start-of-Conversion EV-A

† PU = pin has internal pullup; PD = pin has internal pulldown

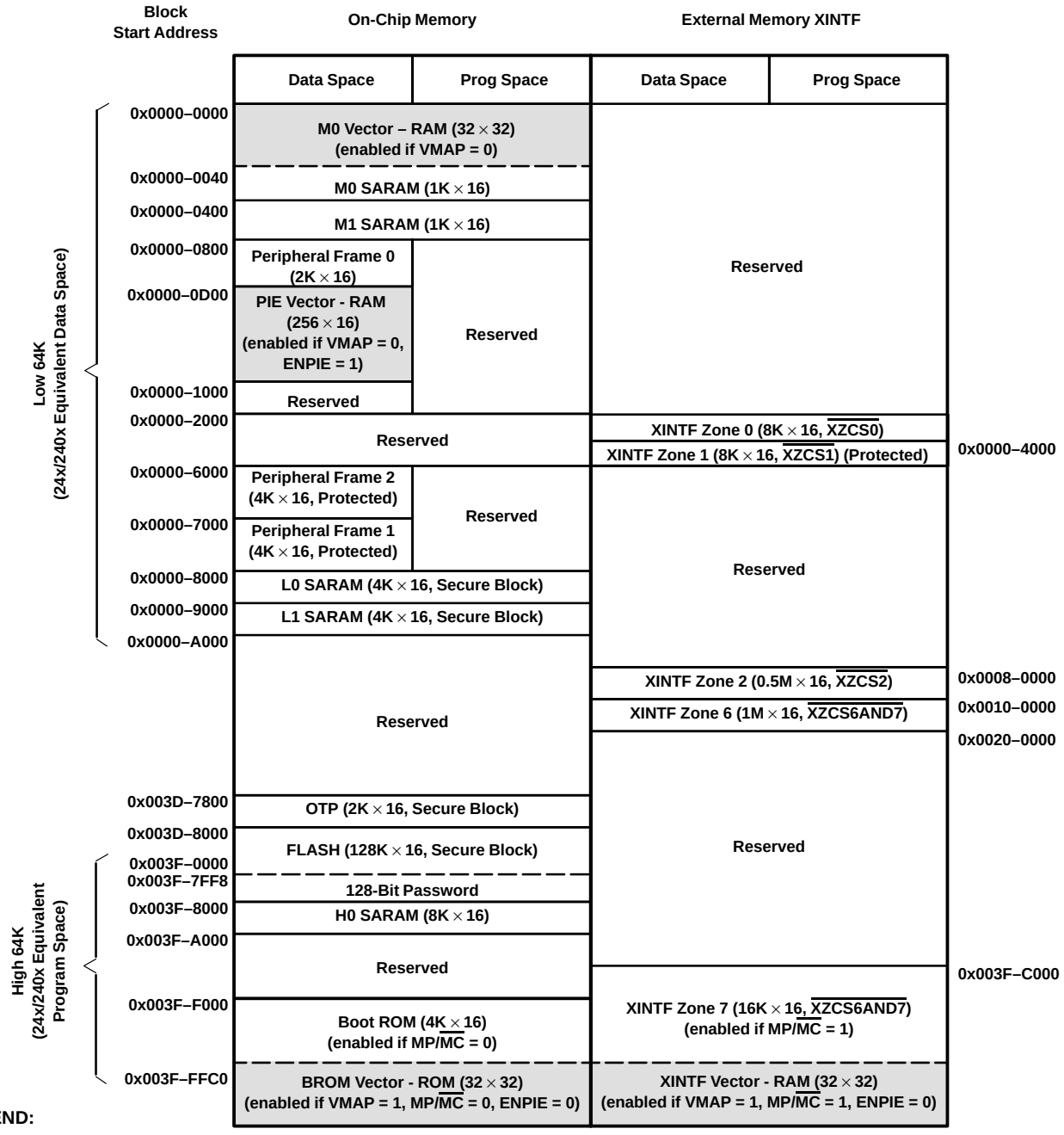
pin functions (continued)

Table 2. Signal Descriptions (Continued)

GPIO	PERIPHERAL SIGNAL	PIN NO.	I/O/Z	DRIVE	PU/PD†	DESCRIPTION
GPIOD OR EVB SIGNALS						
GPIOD5	$\overline{\text{T3CTRIP\_PDPINTB}}$ (I)		I/O/Z		PU	Timer 3 Compare Output Trip
GPIOD6	$\overline{\text{T4CTRIP}}$/EVBSOC (I)		I/O/Z		PU	Timer 4 Compare Output Trip or External ADC Start-of-Conversion EV-B
GPIOE OR INTERRUPT SIGNALS						
GPIOE0	XINT1_$\overline{\text{XBIO}}$ (I)		I/O/Z		PU	GPIO or XINT1 or $\overline{\text{XBIO}}$ core input
GPIOE1	XINT2_ADCSOC (I)		I/O/Z		PU	GPIO or XINT2 or ADC start of conversion
GPIOE2	XNMI_XINT13 (I)		I/O/Z		PU	GPIO or XNMI or XINT13
GPIOF OR SPI SIGNALS						
GPIOF0	SPISIMO (O)		I/O/Z		PU	GPIO or SPI slave in, master out
GPIOF1	SPISOMI (I)		I/O/Z		PU	GPIO or SPI slave out, master in
GPIOF2	SPICLK (I/O)		I/O/Z		PU	GPIO or SPI clock
GPIOF3	SPISTE (I/O)		I/O/Z		PU	GPIO or SPI slave transmit enable
GPIOF OR SCI-A SIGNALS						
GPIOF4	SCITXDA (O)		I/O/Z		PU	GPIO or SCI asynchronous serial port TX data
GPIOF5	SCIRXDA (I)		I/O/Z		PU	GPIO or SCI asynchronous serial port RX data
GPIOF OR CAN SIGNALS						
GPIOF6	CANTX (O)		I/O/Z		PU	GPIO or eCAN transmit data
GPIOF7	CANRX (I)		I/O/Z		PU	GPIO or eCAN receive data
GPIOF OR MCBSP SIGNALS						
GPIOF8	MCLKX (I/O)		I/O/Z		PU	GPIO or transmit clock
GPIOF9	MCLKR (I/O)		I/O/Z		PU	GPIO or receive clock
GPIOF10	MFSX (I/O)		I/O/Z		PU	GPIO or transmit frame synch
GPIOF11	MFSR (I/O)		I/O/Z		PU	GPIO or receive frame synch
GPIOF12	MDX (O)		I/O/Z		PU	GPIO or transmitted serial data
GPIOF13	MDR (I)		I/O/Z		PU	GPIO or received serial data
GPIOG OR XF CPU OUTPUT SIGNAL						
GPIOF14	XF(0)		I/O/Z		PU	GPIO or input clock
GPIOG OR SCI-B SIGNALS						
GPIOG4	SCITXDB (O)		I/O/Z		PU	GPIO or SCI asynchronous serial port transmit data
GPIOG5	SCIRXDB (I)		I/O/Z		PU	GPIO or SCI asynchronous serial port receive data

† PU = pin has internal pullup; PD = pin has internal pulldown

memory map



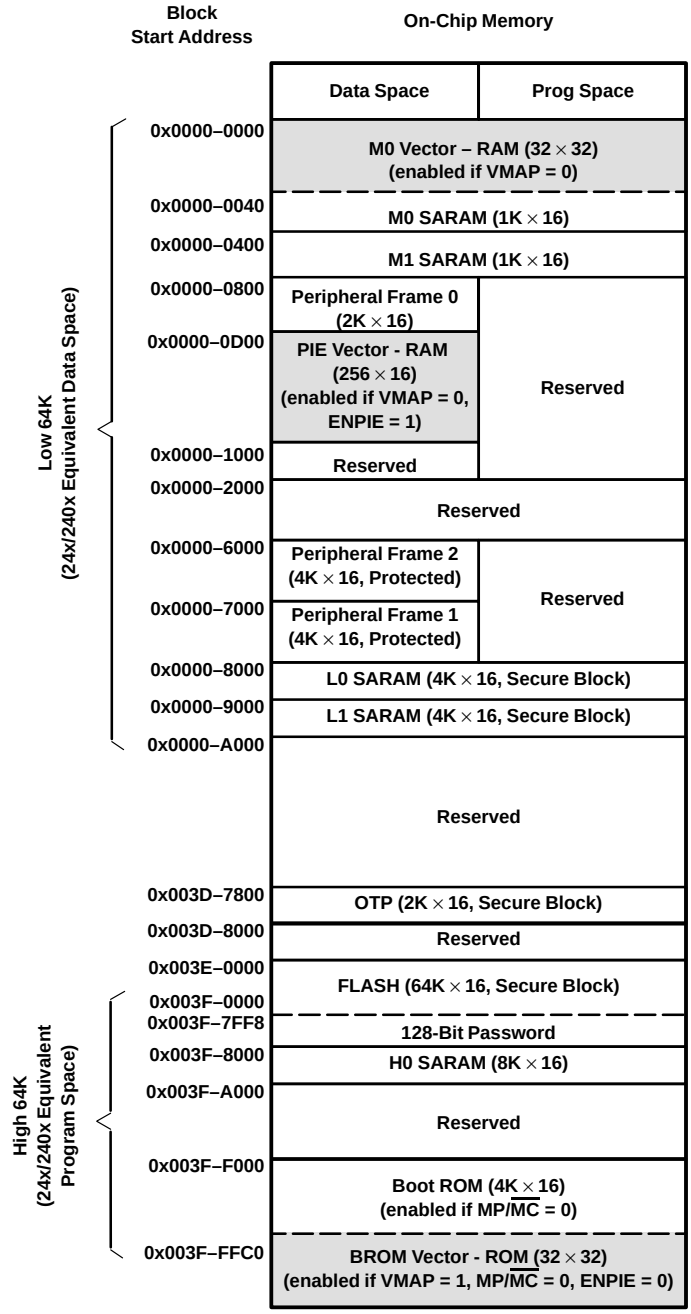
LEGEND:

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

NOTES: A. Memory blocks are not to scale.
B. Reserved locations are reserved for future expansion. Application should not access these areas.
C. Boot ROM and Zone 7 memory maps are active either in on-chip or XINTF zone depending on MP/$\overline{\text{MC}}$, not in both.
D. Peripheral Frame 0, Peripheral Frame 1, and Peripheral Frame 2 memory maps are restricted to data memory only. User program cannot access these memory maps in program space.
E. “Protected” means the order of Write followed by Read operations is preserved rather than the pipeline order.
F. Certain memory ranges are EALLOW protected for spurious writes after configuration.
G. Zone 6 and Zone 7 share the same chip select; hence, these memory blocks have mirrored locations.

Figure 1. F2812 Memory Map

memory map (continued)



LEGEND:

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

NOTES: A. Memory blocks are not to scale. Flash location subject to change.
B. Reserved locations are reserved for future expansion. Application should not access these areas.
C. Boot ROM and Zone 7 memory maps are active either in on-chip or XINTF zone depending on MP/$\overline{MC}$, not in both.
D. Peripheral Frame 0, Peripheral Frame 1, and Peripheral Frame 2 memory maps are restricted to data memory only. User program cannot access these memory maps in program space.
E. "Protected" means the order of Write followed by Read operations is preserved rather than the pipeline order.
F. Certain memory ranges are EALLOW protected for spurious writes after configuration.
G. Zone 6 and Zone 7 share the same chip select; hence, these memory blocks have mirrored locations.

Figure 2. F2810 Memory Map

PRODUCT PREVIEW

memory map (continued)

The “Low 64K” of the memory address range maps into the data space of the 240x. The “High 64K” of the memory address range maps into the program space of the 24x/240x. 24x/240x-compatible code will only execute from the “High 64K” memory area. Hence, the top 32K of Flash and H0 SARAM block can be used to run 24x/240x-compatible code (if MP/$\overline{MC}$ mode is low) or, on F2812, code can be executed from XINTF Zone 7 (if MP/$\overline{MC}$ mode is high).

The XINTF consists of five independent zones. Three zones have their own chip selects and two zones share a single chip select. Each zone can be programmed with its own timing (wait states) and to either sample or ignore external ready signal. This makes interfacing to external peripherals easy and glueless.

Note: The chip selects of XINTF Zone 6 and Zone 7 are merged together into a single chip select ($\overline{ZCS6AND7}$). Refer to the ”External Interface – XINTF (F2812 only)” section of this data sheet for details.

Peripheral Frame 1, Peripheral Frame 2, and XINTF Zone 1 are grouped together so as 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.

On the F2812, at reset, XINTF Zone 7 is enabled if the XMP/$\overline{MC}$ signal is pulled high. This signal selects microprocessor or microcomputer mode of operation. In microprocessor mode, Zone 7 is mapped to high memory such that the vector table is fetched externally. The Boot ROM is disabled in this mode. In microcomputer mode, Zone 7 is disabled such that the vectors are fetched from Boot ROM. This allows the user to either boot from on-chip memory or from off-chip memory. The state of the XMP/$\overline{MC}$ signal on reset is stored in an MP/$\overline{MC}$ mode bit in the XINTCNF2 register. The user can change this mode in software and hence control the mapping of Boot ROM and XINTF Zone 7. No other memory blocks are affected by XMP/$\overline{MC}$.

I/O space is not supported on the F2812 XINTF.

memory map (continued)

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

Table 3. Wait States

AREA	WAIT-STATES	COMMENTS
M0 & M1 SARAMs	0-wait	
Peripheral Frame 0	0-wait	Includes the Flash registers.
Peripheral Frame 1	0-wait (writes) 2-wait (reads)	Cycles can be extended by peripheral generated ready.
Peripheral Frame 2	0-wait (writes) 2-wait (reads)	Fixed. Cycles cannot be extended by the peripheral.
L0 & L1 SARAMs	0-wait	
OTP	Programmable, 0-wait minimum	Programmed via the Flash registers.
Flash	Programmable, 0-wait minmum	Programmed via the Flash registers.
H0 SARAM	0-wait	
Boot-ROM	1-wait	
XINTF	Programmable, 1-wait minimum	Programmed via the XINTF registers. Cycles can be extended by external memory or peripheral. 0-wait operation is not possible.

PRODUCT PREVIEW

description

The TMS320F2810 and TMS320F2812 devices, members of the TMS320C28x™ DSP generation, are highly integrated, high-performance solutions for demanding control applications. The functional blocks and the memory maps are described in subsequent paragraphs.

C28x CPU

The C28x™ DSP generation is the newest member of the TMS320C2000™ DSP platform. The C28x is source code compatible to the 24x/240x DSP devices, hence existing 240x users can leverage their significant software investment. Additionally, the C28x 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 C28x 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 C28x and its 64-bit processing capabilities, enable the C28x to efficiently handle higher numerical resolution problems that would otherwise demand a more expensive floating-point processor solution. 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 C28x has an 8-level-deep protected pipeline with pipelined memory accesses. This pipelining enables the C28x 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.

memory bus (Harvard bus architecture)

As with many DSP 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†
	Program Writes†
	Data Reads
	Program Reads‡
Lowest:	Fetches‡

peripheral bus

To enable migration of peripherals between various Texas Instruments (TI) DSP family of devices, the F2810 and F2812 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. There are two versions of the peripheral bus supported on the F2810 and F2812. One version only supports 16-bit accesses (called peripheral frame 2) and this retains compatibility with C240x compatible peripherals. The other version supports both 16- and 32-bit accesses (called peripheral frame 1) and is used to connect peripherals requiring higher throughput.

TMS320C28x, C28x, and TMS320C2000 are trademarks of Texas Instruments.
† Simultaneous Data and Program writes cannot occur on the Memory Bus.
‡ Simultaneous Program Reads and Fetches cannot occur on the Memory Bus.

real-time JTAG and analysis

The C28x implements the standard IEEE 1149.1 JTAG interface. Additionally, the C28x 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 C28x implements the real-time mode in hardware within the CPU. This is a unique feature to the C28x, 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.

external interface (XINTF) (F2812 only)

This asynchronous interface consists of 19 address lines, 16 data lines, and four chip-select lines. The chip-select lines are mapped to five external zones, Zone 0, 1, 2, 6, and 7. Zones 6 and 7 share a single chip-select. Each of the five zones can be programmed with 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.

flash

The F2812 contains 128K x16 of embedded Flash memory and 2K x16 of OTP memory. The Flash memory is segregated into eight 4K x16 sized sectors, and six 16K x16 sized sectors. The user can individually erase, program and validate a sector while leaving other sectors untouched. 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 hence can be used to execute code or store data information.

The F2810 has 64K x 16 of embedded Flash and 2K x 16 of OTP memory.

M0, M1 SARAMs

All C28x devices will contain these two blocks of single access memory, each 1Kx16 in size. The stack pointer points to the beginning of block M1 on reset. The M0 block overlaps the 240x device B0, B1, B2 RAM blocks and hence the mapping of data variables on the 240x devices can remain at the same physical address on C28x devices. 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.

L0, L1, H0 SARAMs

The F2810 and the F2812 will contain an additional 16K x 16 of single-access RAM, divided into 3 blocks (4K + 4K + 8K). Each block can be independently accessed hence minimizing pipeline stalls. Each block is mapped to both program and data space.

boot ROM

The Boot ROM is factory programmed with boot loading software. Boot-mode signals are provided to tell the boot loader software, programmed into the Boot ROM, 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. The Boot ROM will also contain standard tables, such as SIN/COS waveforms, for use in math related algorithms.

security

The F2810 and F2812 support high levels of security to protect the user firmware from being reversed engineered. The security features a 128-bit password, which the user programs into the Flash. One code security module (CSM) is used to protect the Flash/OTP and the L0/L1 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.

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.

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 F2810/F2812, 45 of the possible 96 interrupts are used by peripherals. The 96 interrupts are grouped into blocks of 8 and each group is fed into one of 12 CPU interrupt lines ($\overline{\text{INT1}}$ to $\overline{\text{INT12}}$). 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 9 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.

external interrupts (XINT1, 2, 13, XNMI)

The F2810 and F2812 support three masked external interrupts (XINT1, 2, 13). XINT13 is combined with one non-masked external interrupt (XNMI). The combined signal name is XNMI_XINT13. Each of the interrupts can be selected for negative or positive edge triggering and can also be enabled/disabled (including the XNMI). The masked interrupts 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.

oscillator and PLL

The F2810 and F2812 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. The PLL block can be set in bypass mode.

watchdog

The F2810 and F2812 support 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.

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 and the event managers, CAP and QEP blocks can be scaled relative to the CPU clock. This enables the timing of peripherals to be decoupled from increasing CPU clock speeds.

low-power modes

The F2810 and F2812 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 will wake the processor from IDLE mode.
STANDBY:	Turn 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:	Turn off oscillator. This mode basically shuts down the device and places it in the lowest possible power consumption mode. Only a reset or XNMI will wake the device from this mode.

PRODUCT PREVIEW

peripheral frames 0, 1, 2 (PFn)

The F2810 and F2812 segregate peripherals into three sections. The mapping of peripherals is as follows:

PF0:	XINTF:	External Interface Configuration Registers
	PIE:	PIE Interrupt Enable and Control Registers Plus PIE Vector Table
	Flash:	Flash Control, Programming, Erase, Verify Registers
	Timers:	CPU-Timers 0, 1, 2 Registers
	CSM:	Code Security Module KEY Registers
PF1:	eCAN:	eCAN Mailbox and Control Registers
PF2:	SYS:	System Control Registers
	GPIO:	GPIO Mux Configuration and Control Registers
	EV:	Event Manager (EVA/EVB) Control Registers
	McBSP:	McBSP Control and TX/RX Registers
	SCI:	Serial Communications Interface (SCI) Control and RX/TX Registers
	SPI:	Serial Peripheral Interface (SPI) Control and RX/TX Registers
	ADC:	12-Bit ADC Registers

general-purpose input/output (GPIO) multiplexer

Most of the peripheral signals are multiplexed with general-purpose I/O (GPIO) signals. This enables the user to use a pin as GPIO if the peripheral signal or function is not used. On reset, all GPIO pins are configured as inputs. The user can then 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.

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-Timers 1 and 2 are reserved for Real-Time OS (RTOS) applications. CPU-Timer 2 is connected to INT14 of the CPU. CPU-Timer 1 can be connected to INT13 of the CPU. CPU-Timer 0 is for general use and is connected to the PIE block.

motor control peripherals

The F2810 and F2812 support the following peripherals which, are used for controlling motors:

EV: The event manager module includes general-purpose timers, full-compare/PWM units, capture inputs (CAP) and quadrature-encoder pulse (QEP) circuits. Two such event managers are provided which enable two three-phase motors to be driven or four two-phase motors. The event managers on the F2810 and F2812 are compatible to the event managers on the 240x devices (with some minor enhancements).

ADC: The ADC block is a 12-bit converter, single ended, 16-channels. It will contain two sample-and-hold units for simultaneous sampling.

serial port peripherals

The F2810 and F2812 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: This is the multichannel buffered serial port that is used to connect to E1/T1 lines, phone-quality codecs for modem applications or high-quality stereo-quality Audio DAC devices. The McBSP receive and transmit registers are supported by a 16-level FIFO. This significantly reduces the overhead for servicing this peripheral.

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 DSP controller and external peripherals or another processor. Typical applications include external I/O or peripheral expansion through devices such as shift registers, display drivers, and ADCs. Multi-device communications are supported by the master/slave operation of the SPI. On the F2810 and the F2812, the port supports a 16-level, receive and transmit FIFO for reducing servicing overhead.

SCI: The serial communications interface is a two-wire asynchronous serial port, commonly known as UART. On the F2810 and the F2812, the port supports a 16-level, receive and transmit FIFO for reducing servicing overhead.

register map

The F2810 device contains three 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 4.
- Peripheral Frame 1: These are peripherals that are mapped to the 32-bit peripheral bus. See Table 5.
- Peripheral Frame 2: These are peripherals that are mapped to the 16-bit peripheral bus. See Table 6.

register map (continued)

Table 4. Peripheral Frame 0 Registers

NAME	ADDRESS RANGE	SIZE (x16)	ACCESS TYPE†
CPU Emulation Register Space	0x0000 0800 0x0000 087F	128	EALLOW protected
Device Emulation Registers	0x0000 0880 0x0000 09FF	384	EALLOW protected
reserved	0x0000 0A00 0x0000 0B00	128	
FLASH Registers‡	0x0000 0A80 0x0000 0ADF	96	EALLOW protected CSM Protected
Code Security Module Registers	0x0000 0AE0 0x0000 0AEF	16	EALLOW protected
reserved	0x0000 0AF0 0x0000 0B1F	48	
XINTF Registers	0x0000 0B20 0x0000 0B3F	32	Not EALLOW protected
reserved	0x0000 0B40 0x0000 0BFF	192	
CPU-TIMER0/1/2 Registers	0x0000 0C00 0x0000 0C3F	64	Not EALLOW protected
reserved	0x0000 0C40 0x0000 0CDF	160	
PIE Registers	0x0000 0CE0 0x0000 0CFF	32	Not EALLOW protected
PIE Vector Table	0x0000 0D00 0x0000 0DFF	256	EALLOW protected
reserved	0x0000 0E00 0x0000 0FFF	512	

† If registers are EALLOW protected, then writes cannot be performed until the user executes the EALLOW instruction. The EDIS instruction disables writes. This prevents stray code or pointers from corrupting register contents.
‡ The Flash Registers are also protected by the Code Security Module (CSM).

Table 5. Peripheral Frame 1 Registers§

NAME	ADDRESS RANGE	SIZE (x16)	ACCESS TYPE
eCAN Registers	0x0000 6000 0x0000 61FF	512	User Accessible
reserved	0x0000 6200 0x0000 6FFF	3584	

§ Peripheral Frame 1 allows 16-bit and 32-bit accesses. All 32-bit accesses are aligned to even address boundaries.

register map (continued)

Table 6. Peripheral Frame 2 Registers†

NAME	ADDRESS RANGE	SIZE (x16)	ACCESS TYPE
reserved	0x0000 7000 0x0000 700F	16	
System Control Registers	0x0000 7010 0x0000 702F	32	EALLOW Protected
reserved	0x0000 7030 0x0000 703F	16	
SPI-A Registers	0x0000 7040 0x0000 704F	16	Not EALLOW Protected
SCI-A Registers	0x0000 7050 0x0000 705F	16	Not EALLOW Protected
reserved	0x0000 7060 0x0000 706F	16	
External Interrupt Registers	0x0000 7070 0x0000 707F	16	Not EALLOW Protected
reserved	0x0000 7080 0x0000 70BF	64	
GPIO Mux Registers	0x0000 70C0 0x0000 70DF	32	EALLOW Protected
GPIO Data Registers	0x0000 70E0 0x0000 70FF	32	Not EALLOW Protected
ADC Registers	0x0000 7100 0x0000 711F	32	Not EALLOW Protected
reserved	0x0000 7120 0x0000 73FF	736	
EV-A Registers	0x0000 7400 0x0000 743F	64	Not EALLOW Protected
reserved	0x0000 7440 0x0000 74FF	192	
EV-B Registers	0x0000 7500 0x0000 753F	64	Not EALLOW Protected
reserved	0x0000 7540 0x0000 774F	528	
SCI Registers	0x0000 7750 0x0000 775F	16	Not EALLOW Protected
reserved	0x0000 7760 0x0000 77FF	160	
McBSP Registers	0x0000 7800 0x0000 783F	64	Not EALLOW Protected
reserved	0x0000 7840 0x0000 7FFF	1984	

† Peripheral Frame 2 only allows 16-bit accesses. All 32-bit accesses are ignored (invalid data may be returned or written).

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 7.

Table 7. Device Emulation Registers

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
DEVICECNF	0x0000 0880 0x0000 0881	2	Device Configuration Register
DEVICEID	0x0000 0882 0x0000 0883	2	Device ID Register
PROTSTART	0x0000 0884	1	Block Protection Start Address Register
PROTRANGE	0x0000 0885	1	Block Protection Range Address Register
reserved	0x0000 0886 0x0000 09FF	378	

Table 8. DEVICECNF Register Bit Definitions

BITS	NAME	TYPE	RESET	DECSRIPTION
1:0	reserved	R/W	1,1	For Test Only
2	reserved	R =0	0	
3	VMAPS	R	0/1	VMAP Configure Status. This indicates the status of VMAP.
4	reserved	R = 0	0	
5	$\overline{\text{XRS}}$	R	0/1	Reset Input Signal Status. This is connected directly to the $\overline{\text{XRS}}$ input pin.
6	reserved	R = 1	1	
7	reserved	R/W	0	
14:8	reserved	R = 0	0:0	
15	reserved	R/W	0	For Test Only
16	reserved	R = 1	1	
17	reserved	R = 1	1	
18	reserved	R = 1	1	
19	ENPROT	R/W	1	Enable Write-Read Protection Mode Bit. This bit, when set to 1, will enable write-read protection as specified by the PROTSTART and PROTRANGE registers. This bit, when set to 0, disables this protection mode.
31:20	spares	R = 0	0	

Table 9. DEVICEID Register Bit Definitions

BITS	NAME	TYPE	RESET	DECSRIPTION
15:0	PARTID	R	Dependent on device	These 16 bits specify the part number of the device as follows: 0x0001: F2810 device 0x0002: F2812 device
31:16	REVID	R	0x0001 (for first silicon)	These 16 bits specify the silicon revision number for the particular part. This number always starts with 0x0001 on the first revision of the silicon and is incremented on any subsequent revisions.

device emulation registers (continued)

The PROTSTART and PROTRANGE registers set the memory address range for which CPU “write” followed by “read” operations are protected (operations occur in sequence rather then in their natural pipeline order). This is necessary protection for certain peripheral operations.

Example: The following lines of code perform a write to register 1 (REG1) location and then the next instruction performs a read from Register 2 (REG2) location. On the processor memory bus, with block protection disabled, the read operation will be issued before the write as shown:

```
MOV   @REG1,AL          -----+
TBIT  @REG2,#BIT_X      -----|----> Read
                             +----> Write
```

If block protection is enabled, then the read is stalled until the write occurs as shown:

```
MOV   @REG1,AL          -----+
TBIT  @REG2,#BIT_X      ---+ |
                           | +----> Write
                           +------> Read
```

NOTE: The C28x CPU automatically protects writes followed by reads to the same memory address. The protection mechanism described above is for cases where the address is not the same, but within a given region in memory (as defined by the PROTSTART and POROTRANGE registers).

Table 10. PROTSTART and PROTRANGE Registers

NAME	ADDRESS	SIZE	TYPE	RESET	DECSRIPTION
PROTSTART	0x0000 0884	16	R/W	0x0100†	The PROTSTART register sets the starting address relative to the 16 most significant bits of the processors lower 22-bit address reach. Hence, the smallest resolution is 64 words.
PROTRANGE	0x0000 0885	16	R/W	0x00FF†	The PROTRANGE register sets the block size (from the starting address), starting with 64 words and incrementing by binary multiples (64, 128, 256, 512, 1K, 2K, 4K, 8K, 16K, ..., 2M).

† The default values of these registers on reset are selected to cover the Peripheral Frame 1, Peripheral Frame 2, and XINTF Zone 1 areas of the memory map (address range 0x0000 4000 to 0x0000 8000).

PRODUCT PREVIEW

device emulation registers (continued)

Table 11. PROTSTART Valid Values†

		REGISTER BITS															
START ADDRESS	REGISTER VALUE	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x0000 0000	0x0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0x0000 0040	0x0001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0x0000 0080	0x0002	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0x0000 00C0	0x0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .	. . . . .
0x003F FF00	0xFFFC	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
0x003F FF40	0xFFFD	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0x003F FF80	0xFFFE	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0x003F FFC0	0xFFFF	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

† The quickest way to calculate register value is to divide the desired block starting address by 64.

Table 12. PROTRANGE Valid Values‡

		REGISTER BITS															
BLOCK SIZE	REGISTER VALUE	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
64	0x0000	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
128	0x0001	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
256	0x0003	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
512	0x0007	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
1K	0x000F	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
2K	0x001F	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
4K	0x003F	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
8K	0x007F	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
16K	0x00FF	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
32K	0x01FF	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1
64K	0x03FF	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
128K	0x07FF	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
256K	0x0FFF	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
512K	0x1FFF	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
1M	0x3FFF	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2M	0x7FFF	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4M	0xFFFF	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

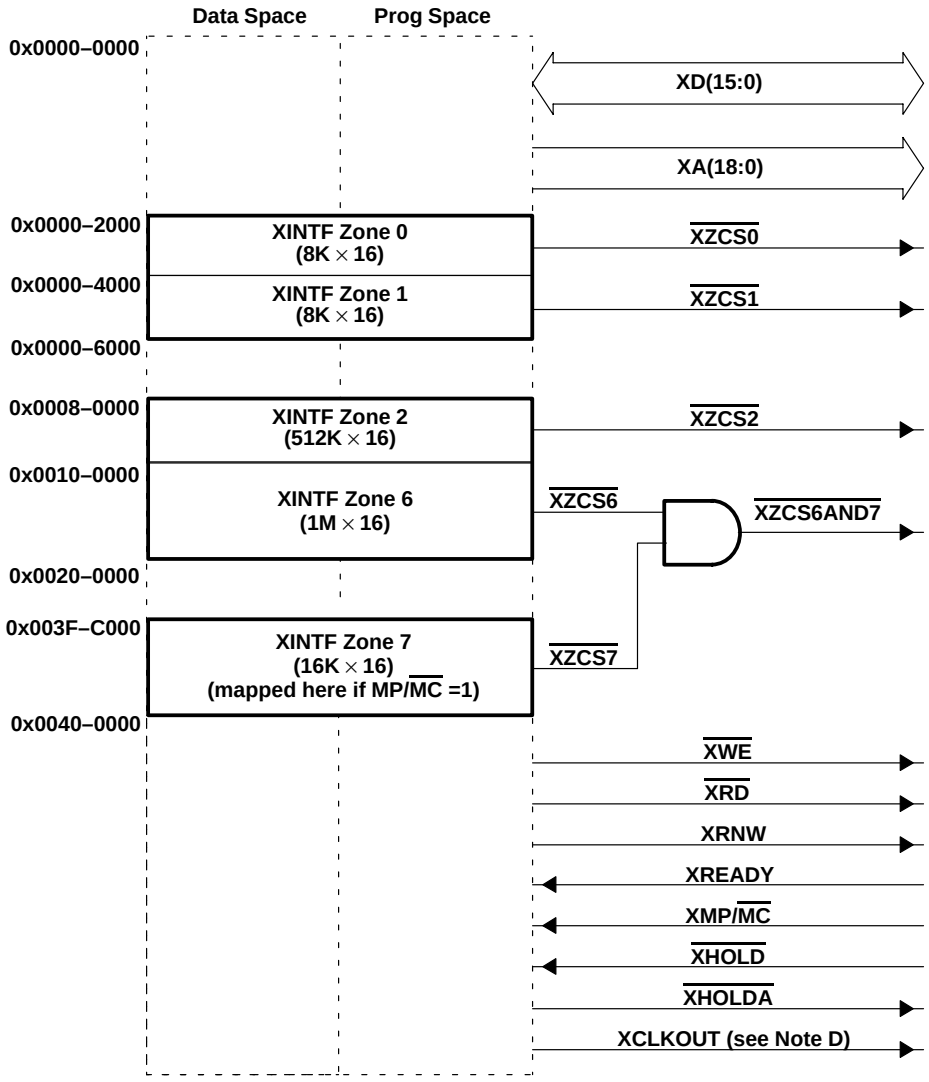
‡ Not all register values are valid. The PROTSTART address value must be a multiple of the range value. For example: if the block size is set to 4K, then the start address can only be at any 4K boundary.

external interface, XINTF (F2812 only)

This section gives a top-level view of the external interface (XINTF) that is implemented on the F2812 device.

The external interface is a non-multiplexed asynchronous bus, similar to the C240x external interface. The external interface on the F2812 is mapped into five fixed zones shown in Figure 3.

Figure 3 shows the F2812 XINTF signals.



NOTES: A. The mapping of XINTF Zone 7 is dependent on the XMP/$\overline{MC}$ device input signal and the MP/$\overline{MC}$ mode bit (bit 8 of XINTCNF2 register). Zones 0, 1, 2, and 6 are always enabled.

B. Each zone can be programmed with different wait states, setup and hold timings, and is supported by zone chip selects ($\overline{XZCS0}$, $\overline{XZCS1}$, $\overline{XZCS2}$, $\overline{XZCS6AND7}$), which toggle when an access to a particular zone is performed. These features enable glueless connection to many external memories and peripherals.

C. The chip selects for Zone 6 and 7 are ANDed internally together to form one chip select ($\overline{XZCS6AND7}$). Any external memory that is connected to $\overline{XZCS6AND7}$ is dually mapped to both Zones 6 and Zone 7. This means that if Zone 7 is disabled (via the MP/$\overline{MC}$ mode) then any external memory is still accessible via Zone 6 address space.

D. XCLKOUT is also pinned out on the F2810.

Figure 3. External Interface Block Diagram

external interface, XINTF (F2812 only) (continued)

The operation and timing of the external interface, can be controlled by the registers listed in Table 13.

Table 13. XINTF Configuration and Control Register Mappings

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
XTIMING0	0x0000–0B20	2	XINTF Timing Register, Zone 0
XTIMING1	0x0000–0B22	2	XINTF Timing Register, Zone 1
XTIMING2	0x0000–0B24	2	XINTF Timing Register, Zone 2
XTIMING6	0x0000–0B2C	2	XINTF Timing Register, Zone 6
XTIMING7	0x0000–0B2E	2	XINTF Timing Register, Zone 7
XINTCNF2	0x0000–0B34	2	XINTF Configuration Register
XBANK	0x0000–0B38	1	XINTF Bank Control Register
XREVISION	0x0000–0B3A	1	XINTF Revision Register

timing registers

XINTF signal timing can be tuned to match specific external device requirements such as setup and hold times to strobe signals for contention avoidance and maximizing bus efficiency. The timing parameters can be configured individually for each zone. This allows the programmer to maximize the efficiency of the bus, based on the type of memory or peripheral that the user needs to access. All XINTF timing values are with respect to XTIMCLK, which is equal to or one-half of the SYSCLKOUT rate, as shown in Figure 4.

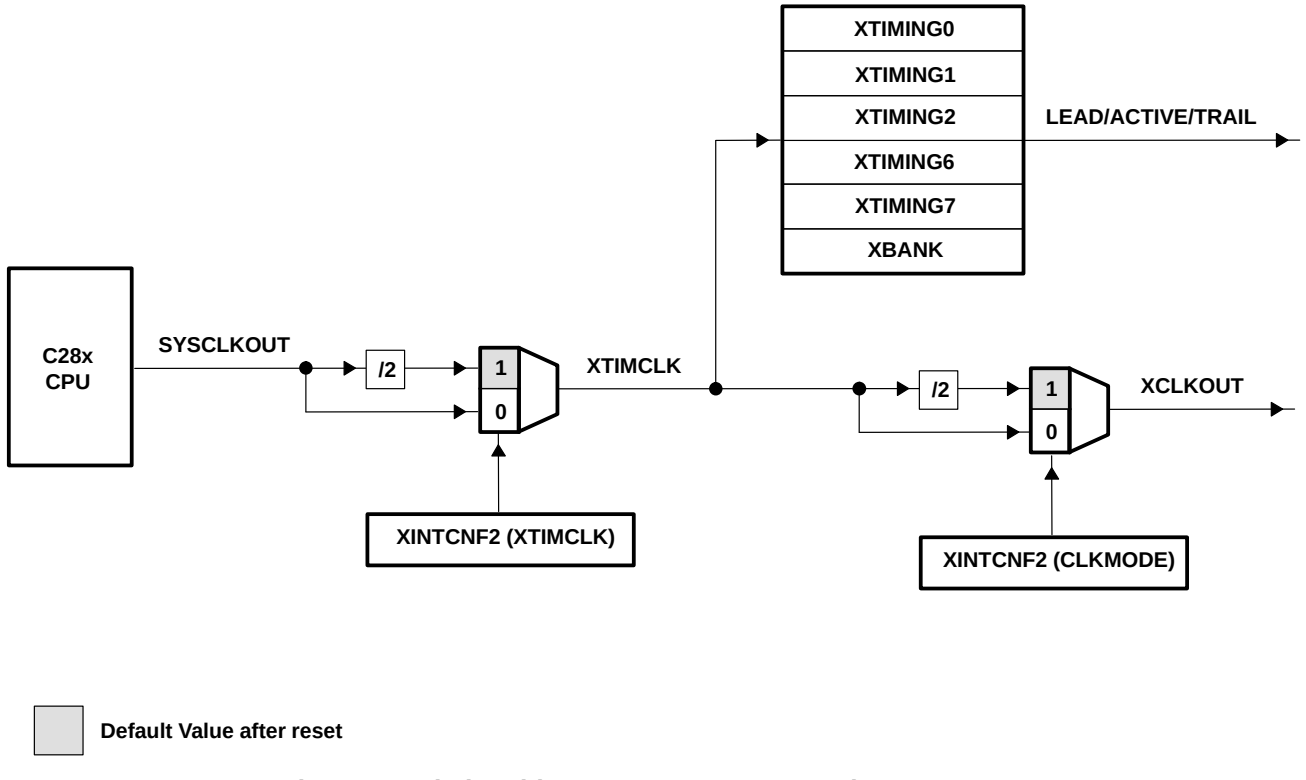


Figure 4. Relationship Between XTIMCLK and SYSCLKOUT

timing registers (continued)

The individual timing parameters can be programmed into the XTIMING registers as described in Table 14.

Table 14. XTIMING0/1/2/6/7 Register Bit Definitions

BIT	NAME	ACCESS	RESET	DESCRIPTION
1:0	XWRTRAIL	R/W	1,1	Two-bit field that defines the write cycle trail period, in XTIMCLK cycles, from 0,1,2,3 (if X2TIMING bit is 0) or 0,2,4,6 (if X2TIMING bit is 1).
4:2	XWRACTIVE	R/W	1,1,1	Three-bit field that defines the write cycle active wait-state period, in XTIMCLK cycles, from 0,1,2,3,4,5,6,7 (if X2TIMING bit is 0) or 0,2,4,6,8,10,12,14 (if X2TIMING bit is 1). Notes: 1. If the USEREADY bit is set to 1 (using XREADY), then XWRACTIVE must be ≥ 1. 2. The active period is by default 1 cycle. Hence the total active period is 1 + XWRACTIVE value.
6:5	XWRLEAD	R/W	1,1	Two-bit field that defines the write cycle lead period, in XTIMCLK cycles, from 1,2,3 (if X2TIMING bit is 0) or 2,4,6 (if X2TIMING bit is 1). Note: XWRLEAD must be ≥ 1.
8:7	XRDTRAIL	R/W	1,1	Two-bit field that defines the read cycle trail period, in XTIMCLK cycles, from 0,1,2,3 (if X2TIMING bit is 0) or 0,2,4,6 (if X2TIMING bit is 1).
11:9	XRDACTIVE	R/W	1,1,1	Three-bit field that defines the read cycle active wait-state period, in XTIMCLK cycles, from 0,1,2,3,4,5,6,7 (if X2TIMING bit is 0) or 0,2,4,6,8,10,12,14 (if X2TIMING bit is 1). Notes: 1. If the USEREADY bit is set to 1 (using XREADY), then XRDACTIVE must be ≥ 1. 2. The active period is by default 1 cycle. Hence the total active period is 1 + XRDACTIVE value.
13:12	XRDLEAD	R/W	1,1	Two-bit field that defines the read cycle lead period, in XTIMCLK cycles, from 1,2,3 (if X2TIMING bit is 0) or 2,4,6 (if X2TIMING bit is 1). Note: XRDLEAD must be ≥ 1.
14	USEREADY	R/W	1	When set, the XREADY signal can be used to further extend the active portion of the cycle past the minimum defined by the XRDACTIVE and XWRACTIVE fields. When cleared XREADY is ignored.
15	READYMODE	R/W	1	When set, the XREADY input is asynchronous. When cleared, the XREADY input is synchronous.
17:16	Reserved	R/W	1,1	Reserved. These two bits must always be written to as 1,1. Any other combination is reserved and will result in incorrect XINTF behavior.
21:18	Reserved	R	0	Reserved
22	X2TIMING	R/W	1	This bit specifies the scaling factor of the LEAD, ACTIVE, TRAIL values in the individual timing registers. If this bit is 0, the values are scaled 1:1. If this bit is 1, the values are scaled 2:1 (doubled). The default mode of operation on power up and reset is 2:1 scaling (doubled) mode.
31:23	Reserved	R	0	

timing registers (continued)

The minimum timing settings for an XINTF access is as follows:

- When the XREADY option is NOT used:

  The minimum strobe setting is Lead = 1, Active = 0, Trail = 0
  Hence: L = 0, A = 0,T = 0 settings are not allowed (L = 1, A = 0,T = 0 or L = 1, A = 1,T = 0 or L = 1, A = 0, T = 1 or greater are allowed)

- When the XREADY option is used:

  The minimum strobe setting is Lead = 1, Active = 1, Trail = 0
  Hence: L = 0, A = 0, T = 0 settings are not allowed (L = 1, A = 1, T = 0 or L = 1, A = 1, T = 1 or greater are allowed).

No logic is included to detect illegal settings.

XINTCNF2 register

Table 15. XINTCNF2 Register Bit Definitions

BITS	TYPE	NAME	RESET	DESCRIPTION
1,0	R/W	Write Buffer Depth	0,0	The write buffer allows the processor to continue execution without waiting for XINTF write accesses to complete. The write buffer depth is selectable as follows: **Depth** **Action** 00 No write buffering. The CPU will be stalled until the write completes on the XINTF. **Note:** Default mode on reset ($\overline{XRS}$). 01 The XINTF will buffer one word. The CPU is stalled until the write cycle begins on the XINTF (there could be a read cycle currently active on the XINTF). 10 One write will be buffered without stalling the CPU. The CPU is stalled if a second write follows. The CPU will be stalled until the first write begins its cycle on the XINTF. 11 Two writes will be buffered without stalling the CPU. The CPU is stalled if a third write follows. The CPU will be stalled until the first write begins its cycle on the XINTF. The buffered access can be 8, 16, or 32 bits in length. Order of execution is preserved, e.g., writes are performed in the order they were accepted. The processor is stalled on XINTF reads until all pending writes are done and the read access completes. If the buffer is full, any pending reads or writes to the buffer will stall the processor. The “Write Buffer Depth” can be changed; however, it is recommended that the write buffer depth be changed only when the buffer is empty (this can be checked by reading the “Write Buffer Level” bits). Writing to these bits when the level is not zero may have unpredictable results.
2	R/W	CLKMODE Mode	1	XCLKOUT divide by 2 mode. If this bit is set to 1, XCLKOUT is a divide by 2 of XTIMCLK. If this bit is set to 0, XCLKOUT is equal to XTIMCLK. All bus timings, irrespective of which mode is enabled, will start from the rising edge of XCLKOUT. The default mode of operation on power up and reset is /2 mode.
3	R/W	CLKOFF	0	Turn XCLKOUT off mode. When this bit is set to 1, the XCLKOUT signal is turned off. This is done for power savings and noise reduction. This bit is set to 0 on a reset.
4	R	Reserved	1	Reserved
5	R	Reserved	0	Reserved

XINTCNF2 register (continued)

Table 15. XINTCNF2 Register Bit Definitions (Continued)

BITS	TYPE	NAME	RESET	DESCRIPTION
7,6	R	WLEVEL	0,0	The current number of writes buffered are detectable as follows: **Level** **Action** 00 empty 01 1 value currently in the write buffer 10 2 values currently in the write buffer 11 3 values currently in the write buffer The value in the write buffer may be 8-, 16-, or 32-bit data. **Note**: There may be a few cycle delay from when a value enters the write buffer to the buffer level depth being updated.
8	R/W	MP/$\overline{\text{MC}}$ Mode		On reset, this bit reflects the state of the XMP/$\overline{\text{MC}}$ input signal sampled at $\overline{\text{XRS}}$. The user can modify the state of this bit by writing a 1 or a 0 to this location. This will be reflected on the XMP/$\overline{\text{MC}}$ output signal. This mode also affects ZONE 7 and Boot ROM mapping as follows: MP/$\overline{\text{MC}}$ = 1, microprocessor state (XINTF ZONE 7 enabled, Boot ROM disabled). MP/$\overline{\text{MC}}$ = 0, microcomputer state (XINTF ZONE 7 disabled, Boot ROM enabled). **Note**: The XMP/$\overline{\text{MC}}$ input signal state is ignored after reset.
9	R/W	HOLD	0	This bit, when low, will automatically grant a request to an external device driving the $\overline{\text{XHOLD}}$ input signal low ($\overline{\text{XHOLDA}}$ output signal is driven low when request granted). This bit, when set high, will not automatically grant a request to an external device driving the $\overline{\text{XHOLD}}$ input signal low ($\overline{\text{XHOLDA}}$ output signal stays high). If this bit is set, while $\overline{\text{XHOLD}}$ and $\overline{\text{XHOLDA}}$ are both low (external bus accesses granted) then the $\overline{\text{XHOLDA}}$ signal is forced high (at the end of the current cycle) and the exteranl interface is taken out of high-impedance mode. On a reset $\overline{\text{XRS}}$, this bit is set to zero. If on a reset the $\overline{\text{XHOLD}}$ signal is active-low, then the bus and all signal strobes must be in high-impedance state and the $\overline{\text{XHOLDA}}$ signal also driven active-low. When HOLD mode is enabled and $\overline{\text{XHOLDA}}$ is active-low (external bus grant active) then the core can still execute code from internal memory. If an access is made to the external interface, then a not ready signal is generated and the core is stalled until the $\overline{\text{XHOLD}}$ signal is removed.
10	R	HOLDS	$\overline{\text{XHOLD}}$ input signal	This bit reflects the current state of the $\overline{\text{XHOLD}}$ input signal. It can be read by the user to determine if an external device is requesting access to the external bus.
11	R	HOLDAS	$\overline{\text{XHOLDA}}$ input signal	This bit reflects the current state of the $\overline{\text{XHOLDA}}$ output signal. It can be read by the user to determine if the external interface is currently granting access to an external device.

XINTCNF2 register (continued)

Table 15. XINTCNF2 Register Bit Definitions (Continued)

BITS	TYPE	NAME	RESET	DESCRIPTION
15:12	X	Reserved	0	Reserved
18:16	R/W	XTIMCLK	0,0,1	These bits select the fundamental clock for the timing of lead, active and trail switching operations as defined by the XTIMING and XBANK registers: **Mode** **Action** 0,0,0 XTIMCLK = SYSCLKOUT/1 0,0,1 XTIMCLK = SYSCLKOUT/2 0,1,0 reserved 0,1,1 reserved 1,0,0 reserved 1,0,1 reserved 1,1,0 reserved 1,1,1 reserved

XBANK register

Table 16. XBANK Register Bit Defintions

BITS	TYPE	NAME	RESET	DESCRIPTION
2:0	R/W	BANK	1,1,1	These bits specify the XINTF zone for which bank switching is enabled, ZONE 0 to ZONE 7. At reset, XINTF Zone 7 is selected.
5:3	R/W	BCYC	1,1,1	These bits specify the number of XTIMCLK cycles to add between any consecutive access that crosses into or out of the specified zone, be it a read or write, program or data space. The number of XTIMCLK cycles can be 0 to 14. On a reset ($\overline{\text{XRS}}$) the value defaults to 14 cycles.
14:6	X	Reserved		
15	R/W	Reserved	1	

XREVISION register

The XREVISION register contains a unique number to identify the particular version of XINTF used in the product. For the F2812, this register will be configured as described in Table 17.

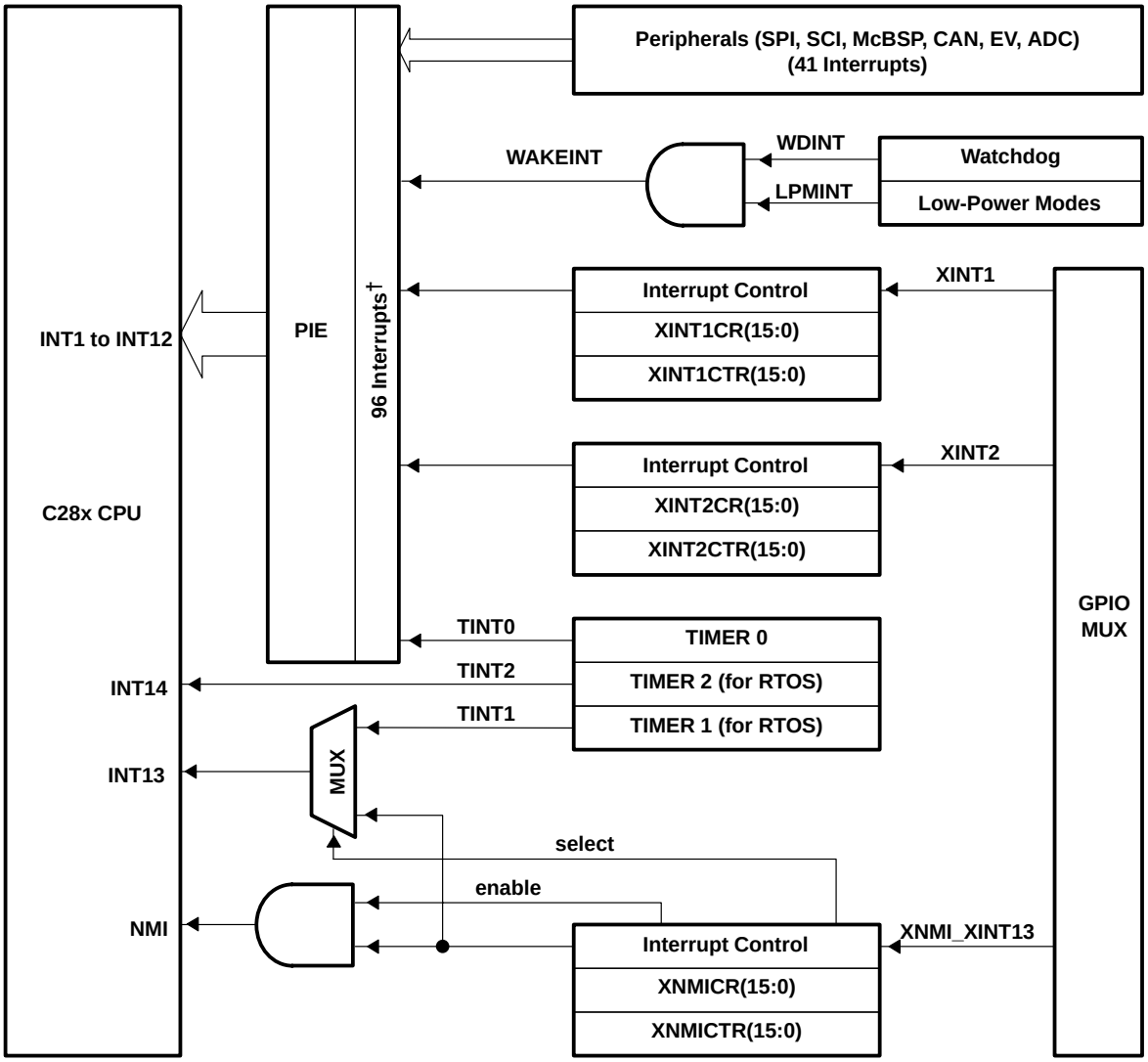
Table 17. XREVISION Register Bit Defintions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
15–0	REVISION	R	0x0004	Current XINTF Revision. For internal use/reference. Test purposes only. Subject to change.

PRODUCT PREVIEW

interrupts

Figure 5 shows how the various interrupt sources are multiplexed within the F2810 and F2812 devices.



† Out of a possible 96 interrupts, 45 are currently used by peripherals.

Figure 5. Interrupt Sources

interrupts (continued)

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 F2810/F2812, 45 of these are used by peripherals as shown in Table 18.

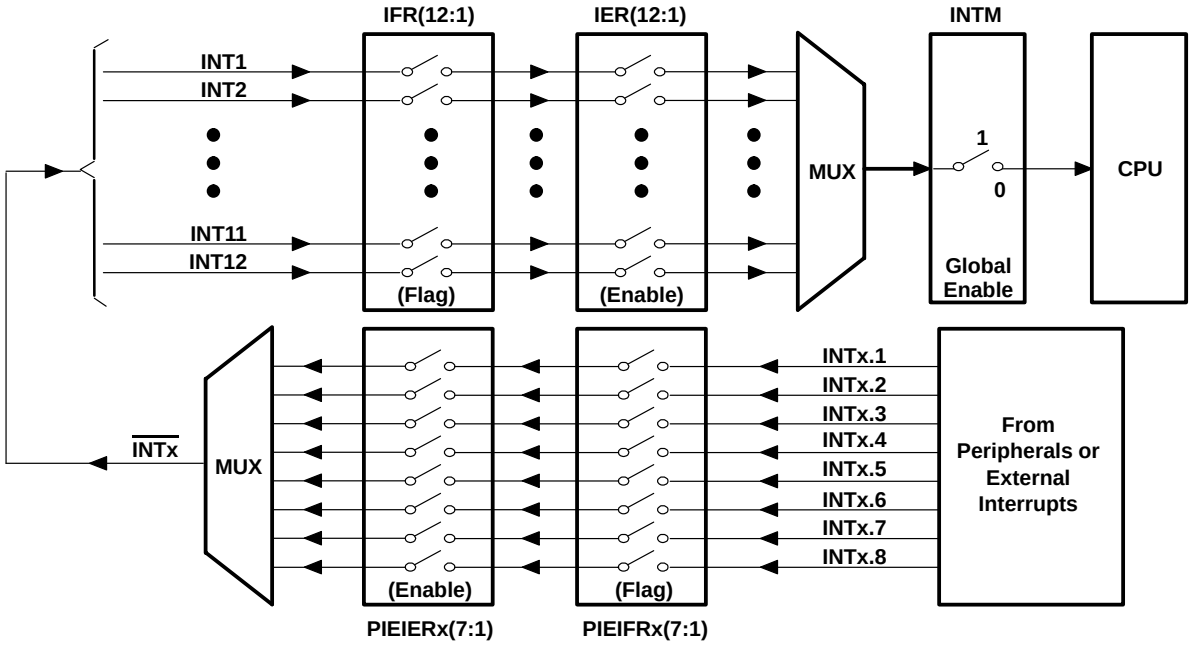


Figure 6. Multiplexing of Interrupts Using the PIE Block

Table 18. PIE Peripheral Interrupts†

CPU INTERRUPTS	PIE INTERRUPTS							
	INTx.1	INTx.2	INTx.3	INTx.4	INTx.5	INTx.6	INTx.7	INTx.8
INT1	PDPINTA (EV-A)	PDPINTB (EV-B)	reserved	XINT1	XINT2	ADCINT (ADC)	TINT0 (TIMER 0)	$\overline{\text{WAKEINT}}$ (LPM/WD)
INT2	CMP1INT (EV-A)	CMP2INT (EV-A)	CMP3INT (EV-A)	T1PINT (EV-A)	T1CINT (EV-A)	T1UFINT (EV-A)	T1OFINT (EV-A)	reserved
INT3	T2PINT (EV-A)	T2CINT (EV-A)	T2UFINT (EV-A)	T2OFINT (EV-A)	CAPINT1 (EV-A)	CAPINT2 (EV-A)	CAPINT3 (EV-A)	reserved
INT4	CMP4INT (EV-B)	CMP5INT (EV-B)	CMP6INT (EV-B)	T3PINT (EV-B)	T3CINT (EV-B)	T3UFINT (EV-B)	T3OFINT (EV-B)	reserved
INT5	T4PINT (EV-B)	T4CINT (EV-B)	T4UFINT (EV-B)	T4OFINT (EV-B)	CAPINT4 (EV-B)	CAPINT5 (EV-B)	CAPINT6 (EV-B)	reserved
INT6	SPIAINT (SPI)	SPIATX (SPI)	reserved	reserved	MRINT (McBSP)	MXINT (McBSP)	***reserved***	***reserved***
INT7	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved
INT8	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved
INT9	RXAINT (SCI-A)	TXAINT (SCI-A)	RXBINT (SCI-B)	TXBINT (SCI-B)	HECC0INT (CAN)	HECC1INT (CAN)	reserved	reserved
INT10	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved
INT11	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved
INT12	reserved	reserved	reserved	reserved	reserved	reserved	reserved	reserved

† Out of the 96 possible interrupts, 45 interrupts are currently used. the remaining interrupts are reserved for future devices. However, these interrupts can be used as software interrupts if they are enabled at the PIEIFRx level.

PRODUCT PREVIEW

vector table mapping

The interrupt vector table can be mapped into the five distinct areas listed in Table 19.

Table 19. Interrupt Vector Table Mapping†

VECTOR MAPS	VECTORS FETCHED FROM	ADDRESS RANGE	VMAP	M0M1MAP	MP/$\overline{MC}$	ENPIE
M1 Vector‡	M1 SARAM Block	0x000000–0x00003F	0	0	X	X
M0 Vector	M0 SARAM Block	0x000000–0x00003F	0	1	X	X
BROM Vector	ROM Block	0x3FFFC0–0x3FFFFF	1	X	0	0
XINTF Vector§	XINTF Zone 7 Block	0x3FFFC0–0x3FFFFF	1	X	1	0
PIE Vector	PIE Block	0x000D00–0x000DFF	1	X	X	1

† On the F2810 and F2812 devices, the VMAP and M0M1MAP modes are set to “1” on reset. The ENPIE mode is forced to “0” on reset.
‡ Vector map M1 Vector is a reserved mode only.
§ Valid on F2812 only

After reset operation, the vector table will be located in the areas listed in Table 20.

Table 20. Vector Table Mapping After Reset Operation†

VECTOR MAPS	RESET FETCHED FROM	ADDRESS RANGE	VMAP	M0M1MAP	MP/$\overline{MC}$	ENPIE
BROM Vector	ROM Block	0x3FFFC0–0x3FFFFF	1	1	0	0
XINTF Vector§	XINTF Zone 7 Block	0x3FFFC0–0x3FFFFF	1	1	1	0

† On the F2810 and F2812 devices, the VMAP and M0M1MAP modes are set to “1” on reset. The ENPIE mode is forced to “0” on reset.
§ Valid on F2812 only

The vector mapping is controlled by the following mode bits/signals:

VMAP: This bit is found in Status Register 1 (bit 3). A device reset sets this bit to 1. The state of this bit can be modified by writing to ST1 or by “SETC/CLRC VMAP” instructions.

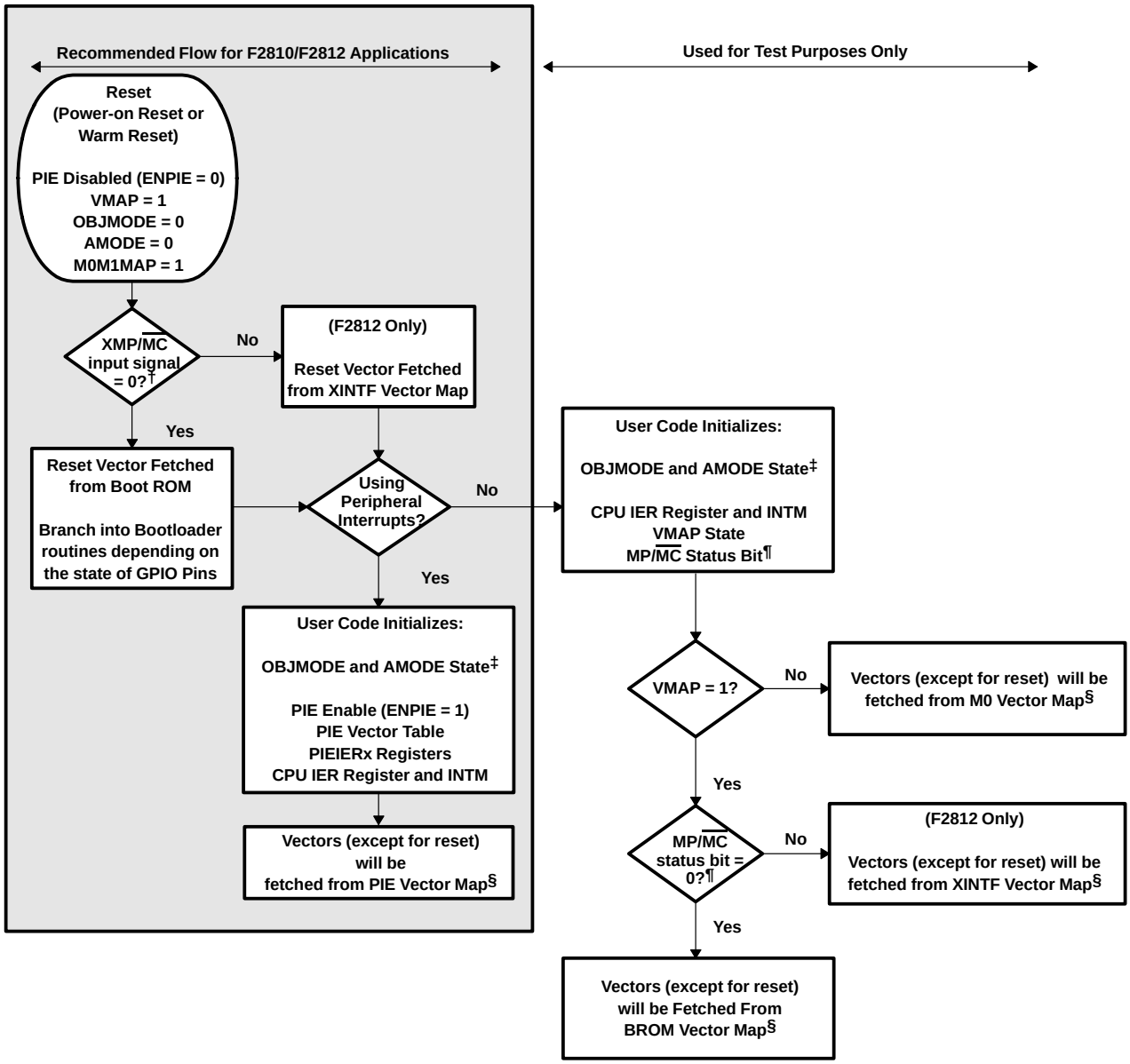
M0M1MAP: This bit is found in Status Register 1 (bit 11). A device reset sets this bit to 1. The state of this bit can be modified by writing to ST1 or by “SETC/CLRC M0M1MAP” instructions. This bit should remain set. M0M1MAP = 0 is reserved for TI testing.

MP/$\overline{MC}$: This bit is found in XINTCNF2 Register (bit 8). On the F2812, the default value of this bit, on reset, is set by the XMP/$\overline{MC}$ input device signal. On the F2810, XMP/$\overline{MC}$ is tied low internally. The state of this bit can be modified by writing to the XINTCNF2 register (address 0x0000 0B34).

ENPIE: This bit is found in PIECTRL Register (bit 0). The default value of this bit, on reset, is set to “0” (PIE disabled). The state of this bit can be modified by writing to the PIECTRL register (address 0x0000 0CE0).

The external interrupts are configured using the registers listed in Table 27.

vector table mapping (continued)



† The XMP/MC input signal is tied low internally on the F2810.
‡ The compatibility operating mode of the F2810 and F2812 is determined by a combination of the OBJMODE and AMODE bits in Status Register 1 (ST1):

Operating Mode	OBJMODE	AMODE	
C28x Mode	1	0	
C2xLP Source-Compatible	1	1	
C27x Object-Compatible	0	0	(Default at reset)

§ The reset vector is always fetched from either the BROM or XINTF vector map depending on the XMP/MC input signal.
¶ The state of the XMP/MC signal is latched into the MP/MC bit at reset, it can then be modified by software.

Figure 7. Reset Flow Diagram

PIE vector map

The PIE Vector Table (Table 21) consists of a 256 x 16 SARAM that can also be used as RAM if the PIE block is not in use. The PIE vector table contents are undefined on reset. Interrupt priority for INT1 to INT12 is fixed by the CPU. Priority for each group of 8 interrupts is, controlled by the PIE. For example: if INT1.1 should occur simultaneously with INT8.1, both interrupts will be presented to the CPU simultaneously by the PIE block, and the CPU will service INT1.1 first. If INT1.1 should occur simultaneously with INT1.8, then INT1.1 will be sent to the CPU first and then INT1.8 will follow. Interrupt prioritization is performed during the vector fetch portion of the interrupt processing. A “TRAP 1” to “TRAP 12” instruction or an “INTR INT1” to “INTR INT12” instruction will always fetch the vector from the first location of each group (“INTR1.1” to “INT12.1”). Hence, it is recommended that these instructions not be used when PIE is enabled. The “TRAP 0” operation will fetch the vector from location 0x0000 0D00. The vector table is EALLOW protected.

Table 21. PIE Vector Table

NAME	ADDRESS	SIZE (x16)	DESCRIPTION	CORE PRIORITY	PIE GROUP PRIORITY
not used	0x0000 0D00	2	RESET never fetched here	1 (highest)	–
not used	0x0000 0D02	2	INT1 remapped to INT1.1–INT1.8 below	–	–
not used	0x0000 0D04	2	INT2 remapped to INT2.1–INT2.8 below	–	–
not used	0x0000 0D06	2	INT3 remapped below	–	–
not used	0x0000 0D08	2	INT4 remapped below	–	–
not used	0x0000 0D0A	2	INT5 remapped below	–	–
not used	0x0000 0D0C	2	INT6 remapped below	–	–
not used	0x0000 0D0E	2	INT7 remapped below	–	–
not used	0x0000 0D10	2	INT8 remapped below	–	–
not used	0x0000 0D12	2	INT9 remapped below	–	–
not used	0x0000 0D14	2	INT10 remapped below	–	–
not used	0x0000 0D16	2	INT11 remapped below	–	–
not used	0x0000 0D18	2	INT12 remapped below	–	–
INT13	0x0000 0D1A	2	External Interrupt 13 (XINT13) or CPU-Timer 1 (for RTOS use)	17	–
INT14	0x0000 0D1C	2	CPU-Timer 2 (for RTOS use)	18	–
DATALOG	0x0000 0D1E	2	CPU Data Logging Interrupt	19 (lowest)	–
RTOSINT	0x0000 0D20	2	CPU Real-Time OS Interrupt	4	–
EMUINT	0x0000 0D22	2	CPU Emulation Interrupt	2	–
NMI	0x0000 0D24	2	External Non-Maskable Interrupt	3	–
ILLEGAL	0x0000 0D26	2	Illegal Operation	–	–
USER0	0x0000 0D28	2	User Defined Trap	–	–
.	.	.	.	.	.
USER11	0x0000 0D3E	2	User Defined Trap	–	–
INT1.1	0x0000 0D40	2	Group 1 Interrupt Vectors	5	1 (highest)
.	.	.			.
INT1.8	0x0000 0D4E	2			8 (lowest)
. . .	. . .	. . .	Group 2 Interrupt Vectors to Group 11 Interrupt Vectors	6 to 15	
INT12.1	0x0000 0DF0	2	Group 12 Interrupt Vectors	16	1 (highest)
.	.	.			.
INT12.8	0x0000 0DFE	2			8 (lowest)

PIE registers

The registers controlling the functionality of the PIE block are listed in Table 22.

Table 22. PIE Configurations and Control Register Mappings†

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
PIECTRL	0x0000–0CE0	1	PIE, Control Register
PIEACK	0x0000–0CE1	1	PIE, Acknowledge Register
PIEIER1	0x0000–0CE2	1	PIE, INT1 Group Enable Register
PIEIFR1	0x0000–0CE3	1	PIE, INT1 Group Flag Register
PIEIER2	0x0000–0CE4	1	PIE, INT2 Group Enable Register
PIEIFR2	0x0000–0CE5	1	PIE, INT2 Group Flag Register
PIEIER3	0x0000–0CE6	1	PIE, INT3 Group Enable Register
PIEIFR3	0x0000–0CE7	1	PIE, INT3 Group Flag Register
PIEIER4	0x0000–0CE8	1	PIE, INT4 Group Enable Register
PIEIFR4	0x0000–0CE9	1	PIE, INT4 Group Flag Register
PIEIER5	0x0000–0CEA	1	PIE, INT5 Group Enable Register
PIEIFR5	0x0000–0CEB	1	PIE, INT5 Group Flag Register
PIEIER6	0x0000–0CEC	1	PIE, INT6 Group Enable Register
PIEIFR6	0x0000–0CED	1	PIE, INT6 Group Flag Register
PIEIER7	0x0000–0CEE	1	PIE, INT7 Group Enable Register
PIEIFR7	0x0000–0CEF	1	PIE, INT7 Group Flag Register
PIEIER8	0x0000–0CF0	1	PIE, INT8 Group Enable Register
PIEIFR8	0x0000–0CF1	1	PIE, INT8 Group Flag Register
PIEIER9	0x0000–0CF2	1	PIE, INT9 Group Enable Register
PIEIFR9	0x0000–0CF3	1	PIE, INT9 Group Flag Register
PIEIER10	0x0000–0CF4	1	PIE, INT10 Group Enable Register
PIEIFR10	0x0000–0CF5	1	PIE, INT10 Group Flag Register
PIEIER11	0x0000–0CF6	1	PIE, INT11 Group Enable Register
PIEIFR11	0x0000–0CF7	1	PIE, INT11 Group Flag Register
PIEIER12	0x0000–0CF8	1	PIE, INT12 Group Enable Register
PIEIFR12	0x0000–0CF9	1	PIE, INT12 Group Flag Register
reserved	0x0000–0CFA 0x0000–0CFF	6	reserved

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

PRODUCT PREVIEW

PIE registers (continued)

Table 23. PIECTRL Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
0	ENPIE	R/W	0	Enable vector fetching from PIE block. When this bit is set to 1, all vectors are fetched from the PIE vector table. If this bit is set to 0, the PIE block is disabled and vectors are fetched as normal. All PIE block registers (PIEACK, PIEIFR, PIEIER) can be accessed even when the PIE block is disabled.
15:1	PIEVECT	R	0	Vector fetch address. Displays the address of the vector that was fetched. The least significant bit of the address is ignored and only bits 1 to 15 are shown. The vector address can be used to determine which interrupt generated the fetch.

Table 24. PIEACK Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
11:0	PIEACK	R/W=1	0	Writing a 1 to the respective interrupt bit enables the PIE block to drive a pulse into the CPU interrupts input, if an interrupt is pending on any of the group interrupts. Reading this register indicates if an interrupt is pending in the respective group. Bit 0 refers to INT1 up to Bit 11, which refers to INT12. Note: Writes of 0 are ignored.
15:12	spares	R=0	0	

Table 25. PIEIERx Register Bit Definitions†

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
0	INTx.1	R/W	0	These register bits individually enable an interrupt within a group. They behave very much like the CPU interrupt enable register. Setting a bit to 1 will enable the servicing of the respective interrupt. Setting a bit to 0 will disable the servicing of the bit.
1	INTx.2	R/W	0	
2	INTx.3	R/W	0	
3	INTx.4	R/W	0	
4	INTx.5	R/W	0	
5	INTx.6	R/W	0	
6	INTx.7	R/W	0	
7	INTx.8	R/W	0	
15:8	spares	R=0	0	

† x = 1 to 12. INTx means CPU interrupts INT1 to INT12.

Table 26. PIEIFRx Register Bit Definitions†

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
0	INTx.1	R/W	0	These register bits indicate if an interrupt is currently active. They behave very much like the CPU interrupt flag register. When an interrupt is active, the respective register bit is set. The bit is cleared when the interrupt is serviced or by writing a 0 to the register bit. This register can also be read to determine which interrupts are active or pending. Note: The PIEIFR register bit is cleared during the interrupt vector fetch portion of the interrupt processing.
1	INTx.2	R/W	0	
2	INTx.3	R/W	0	
3	INTx.4	R/W	0	
4	INTx.5	R/W	0	
5	INTx.6	R/W	0	
6	INTx.7	R/W	0	
7	INTx.8	R/W	0	
15:8	spares	R=0	0	

† x = 1 to 12. INTx means CPU interrupts INT1 to INT12.

PIE/CPU interrupt response

Figure 8 shows the behavior of the PIE hardware under various PIEIFR and PIEIER register conditions. There is one PIEACK bit for every CPU interrupt group (INT1 to INT12) and is referred to as PIEACK(x). There is a corresponding PIEIFR and PIEIER register for each group and are referred to as the PIEIFRx and PIEIERx registers. Figure 8 describes the operation of one PIE interrupt. This flow is common to all PIE interrupts.

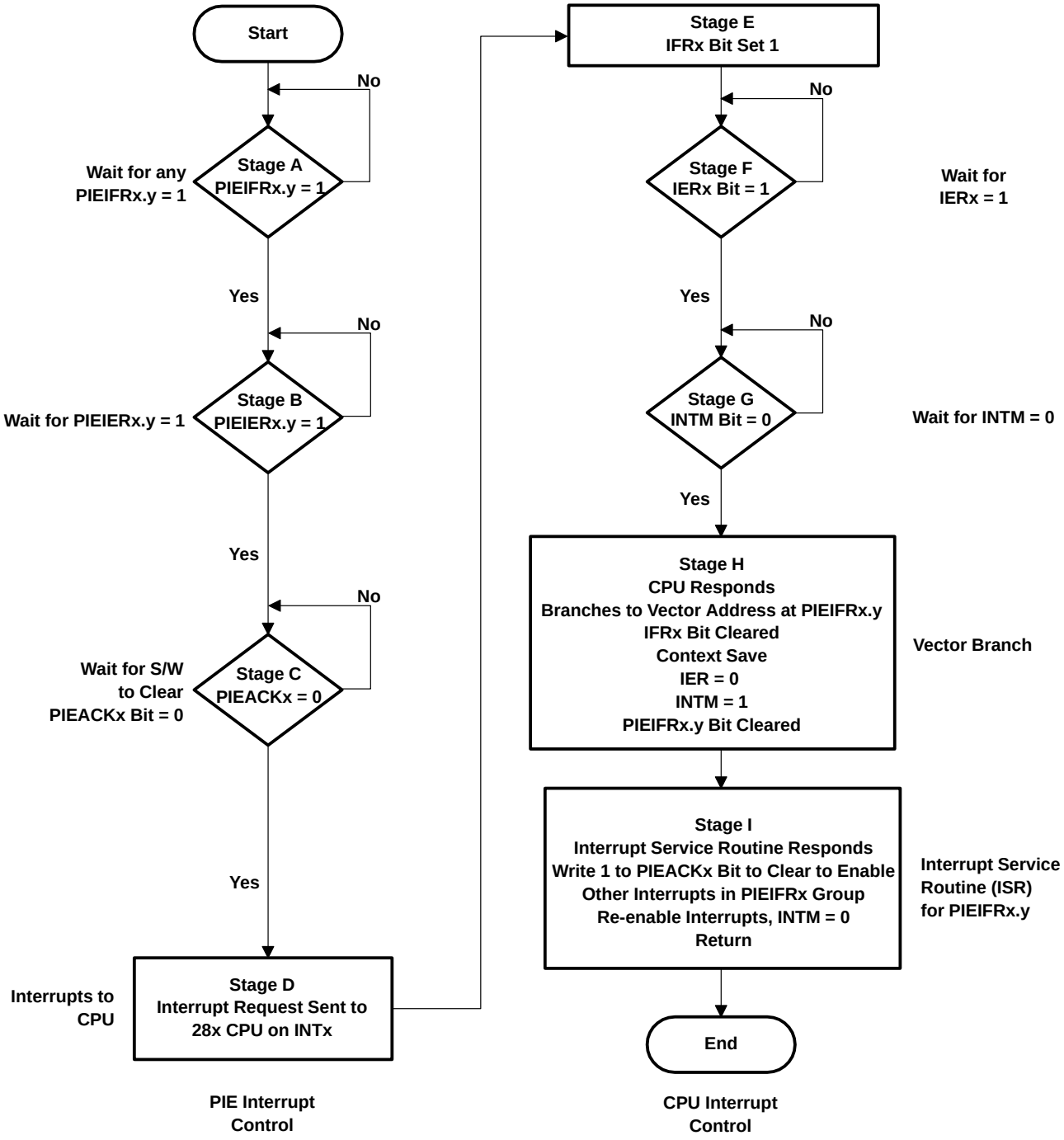


Figure 8. Typical PIE/CPU Interrupt Response–INTx.y

external interrupts

Table 27. External Interrupts Registers

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
XINT1CR	0x0000 7070	1	XINT1 configuration register
XINT2CR	0x0000 7071	1	XINT2 configuration register
reserved	0x0000 7072 0x0000 7076	5	
XNMICR	0x0000 7077	1	XNMI configuration register
XINT1CTR	0x0000 7078	1	XINT1 counter register
XINT2CTR	0x0000 7079	1	XINT2 counter register
reserved	0x0000 707A 0x0000 707E	5	
XNMICTR	0x0000 707F	1	XNMI counter register

Each external interrupt can be enabled/disabled or qualified using positive or negative going edge. The register bits to control this are described in Table 28.

Table 28. XINT1/2CR Register Bit Definitions

BITS	NAME	TYPE	RESET	DESCRIPTION
0	ENABLE	R/W	0	0 Interrupt Disabled 1 Interrupt Enabled
1	reserved	R = 0	0	
2	POLARITY	R/W	0	0 Interrupt is selected as negative edge triggered 1 Interrupt is selected as positive edge triggered
15:3	reserved	R = 0	0:0	

Table 29 shows the bit definitions of the XNMICR register.

Table 29. XNMICR Register Bit Definitions

BITS	NAME	TYPE	RESET	DESCRIPTION
0	ENABLE	R/W	0	0 NMI Interrupt Disabled 1 NMI Interrupt Enabled
1	SELECT	R/W	0	0 Timer 1 Connected To INT13 1 XNMI Connected To INT13
2	POLARITY	R/W	0	0 Interrupt is selected as negative edge triggered 1 Interrupt is selected as positive edge triggered
15:3	reserved	R = 0	0:0	

external interrupts (continued)

The masked interrupts, XINT1/2 and NMI, also contain a 16-bit up-counter register that is reset to 0x0000 whenever an interrupt edge is detected. This counter can be used to accurately time stamp the occurrence of the interrupt. Table 30 shows the bit definitions of the XINT1/2CTR and XNMICTR registers.

Table 30. XINT1/2CTR and XNMICTR Registers Bit Definitions

BITS	NAME	TYPE	RESET	DESCRIPTION
15:0	INTCTR	R	0:0	This is a free running 16-bit up-counter that is clocked at the SYSCLKOUT rate. The counter value is reset to 0x0000 when a valid interrupt edge is detected and then continues counting until the next valid interrupt edge is detected. The counter must only be reset by the selected POLARITY edge as selected in the respective interrupt control register. When the interrupt is disabled, the counter will stop. The counter is a free-running counter and will wrap around to zero when the max value is reached. The counter is a read only register and can only be reset to zero by a valid interrupt edge or by reset.

system control

This section describes the F2810 and F2812 oscillator, PLL and clocking mechanisms, the watchdog function and the low power modes. Figure 9 shows the various clock and reset domains in the F2810 and F2812 devices that will be discussed.

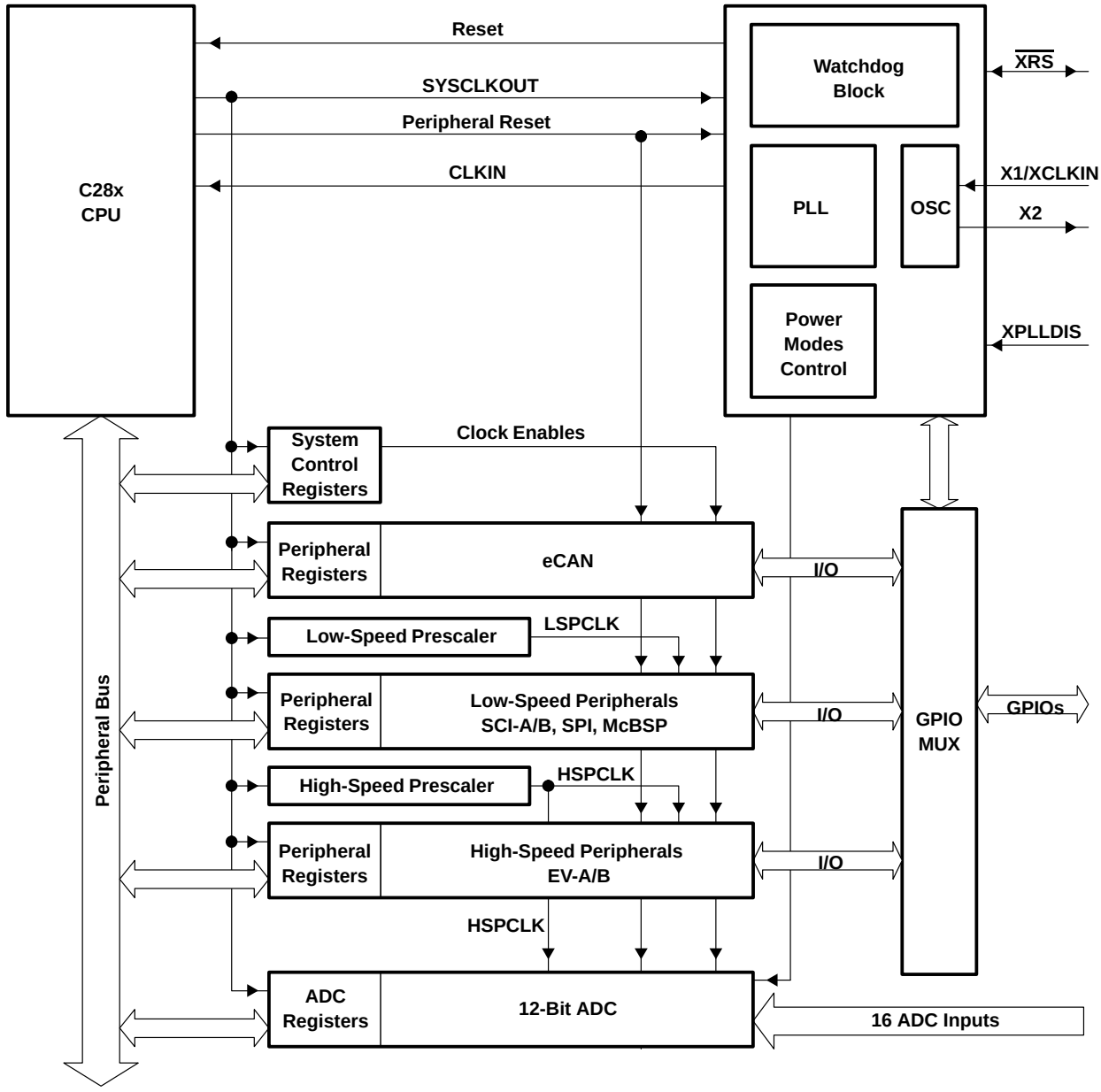


Figure 9. Clock and Reset Domains

system control (continued)

The PLL, clocking, watchdog and low-power modes, are controlled by the registers listed in Table 31.

Table 31. PLL, Clocking, Watchdog, and Low-Power Mode Registers†

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
reserved	0x0000–7010 0x0000–7017	8	
reserved	0x0000–7018	1	
reserved	0x0000–7019	1	
HISPCP	0x0000–701A	1	High-Speed Peripheral Clock Prescaler Register for HSPCLK clock
LOSPCP	0x0000–701B	1	Low-Speed Peripheral Clock Prescaler Register for HSPCLK clock
PCLKCR	0x0000–701C	1	Peripheral Clock Control Register
reserved	0x0000–701D	1	
LPMCR0	0x0000–701E	1	Low Power Mode Control Register 0
LPMCR1	0x0000–701F	1	Low Power Mode Control Register 1
reserved	0x0000–7020	1	
PLLCR	0x0000–7021	1	PLL Control Register‡
SCSR	0x0000–7022	1	System Control & 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	

† All of the above registers can only be accessed, by executing the EALLOW instruction.
‡ The PLL control register (PLLCR) is reset to a known state by the $\overline{\text{XRS}}$ signal only.

PRODUCT PREVIEW

system control (continued)

The PCLKCR1 and PCLKCR2 registers basically enable/disable clocks to the various peripheral modules in the F2810 and F2812 devices. Table 32 lists the bit descriptions of the PCLKCR1 and PCLKCR2 registers.

Table 32. PCLKCR Register Bit Definitions†

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
0	EVAENCLK	R/W	0	If this bit is set, it enables the high-speed clock (HSPCLK) within the EV-A peripheral. For low power operation, this bit is set to zero by the user or by reset.
1	EVBENCLK	R/W	0	If this bit is set, it enables the high-speed clock (HSPCLK) within the EV-B peripheral. For low power operation, this bit is set to zero by the user or by reset.
2	reserved	R=0	0	reserved
3	ADCENCLK	R/W	0	If this bit is set, it enables the high-speed clock (HSPCLK) within the ADC peripheral. For low power operation, this bit is set to zero by the user or by reset.
7:4	reserved	R=0	0:0	
8	SPIAENCLK	R/W	0	If this bit is set, it enables the low-speed clock (LSPCLK) within the SPI peripheral. For low power operation, this bit is set to zero by the user or by reset.
9	reserved	R=0	0	reserved
10	SCIAENCLK	R/W	0	If this bit is set, it enables the low-speed clock (LSPCLK) within the SCI-A peripheral. For low power operation, this bit is set to zero by the user or by reset.
11	SCIBENCLK	R/W	0	If this bit is set, it enables the low-speed clock (LSPCLK) within the SCI-B peripheral. For low power operation, this bit is set to zero by the user or by reset.
12	MAENCLK	R/W	0	If this bit is set, it enables the low-speed clock (LSPCLK) within the McBSP peripheral. For low power operation, this bit is set to zero by the user or by reset.
13	reserved	R=0	0	reserved
14	HECCAENCLK	R/W	0	If this bit is set, it enables the system clock within the CAN peripheral. For low power operation, this bit is set to zero by the user or by reset.
15	reserved	R=0	0	reserved

† If a peripheral block is not used, then the clock to that peripheral can be turned off to minimize power consumption.

system control (continued)

The system control and status register contains the watchdog override bit and the watchdog interrupt enable/disable bit. Table 33 describes the bit functions of the SCSR register.

Table 33. SCSR Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
0	WDOVERRIDE	R/W=1	1	If this bit is set to 1, the user is allowed to change the state of the Watchdog disable (WDDIS) bit in the Watchdog Control (WDCR) register (refer to Watchdog Block section of this data sheet). If the WDOVERRIDE bit is cleared, by writing a 1 the WDDIS bit cannot be modified by the user. Writing a 0 will have no effect. If this bit is cleared, then it will remain in this state until a reset occurs. The current state of this bit is readable by the user.
1	WDENINT	R/W	0	If this bit is set to 1, the watchdog reset ($\overline{\text{WDRST}}$) output signal is disabled and the watchdog interrupt ($\overline{\text{WDINT}}$) output signal is enabled. If this bit is zero, then the $\overline{\text{WDRST}}$ output signal is enabled and the $\overline{\text{WDINT}}$ output signal is disabled. This is the default state on reset ($\overline{\text{XRS}}$).
15:2	reserved	R=0	0:0	

The HISPCP and LOSPCP registers are used to configure the high- and low-speed peripheral clocks, respectively. See Table 34 for the HISPCP bit definitions and Table 35 for the LOSPCP bit definitions.

Table 34. HISPCP Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
2:0	HSPCLK	R/W	0,0,1	These bits configure the high-speed peripheral clock (HSPCLK) rate relative to SYSCLKOUT: 000 HSPCLK = SYSCLKOUT / 1 001 HSPCLK = SYSCLKOUT / 2 010 HSPCLK = SYSCLKOUT / 4 011 HSPCLK = SYSCLKOUT / 6 100 HSPCLK = SYSCLKOUT / 8 101 HSPCLK = SYSCLKOUT / 10 110 HSPCLK = SYSCLKOUT / 12 111 HSPCLK = SYSCLKOUT / 14 HSPCLK = SYSCLKOUT / (HSPCLK x 2) = SYSCLKOUT if HISPCP value is zero
15:3	reserved	R=0	0:0	

system control (continued)

Table 35. LOSPCP Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
2:0	LSPCLK	R/W	0,1,0	These bits configure the low-speed peripheral clock (LSPCLK) rate relative to SYSCLKOUT: 000 LSPCLK = SYSCLKOUT / 1 001 LSPCLK = SYSCLKOUT / 2 010 LSPCLK = SYSCLKOUT / 4 011 LSPCLK = SYSCLKOUT / 6 100 LSPCLK = SYSCLKOUT / 8 101 LSPCLK = SYSCLKOUT / 10 110 LSPCLK = SYSCLKOUT / 12 111 LSPCLK = SYSCLKOUT / 14 LSPCLK = SYSCLKOUT / (LSPCLK x 2) = SYSCLKOUT if LOSPCP value is zero
15:3	reserved	R=0	0:0	

Note: The HSPCLK is set to SYSCLKOUT/2 and LSPCLK is set to SYSCLKOUT/4 on reset.

PRODUCT PREVIEW

OSC and PLL block

The OSC and PLL block on the F2810 and F2812 will use a zero-pin Phase-Locked Loop (ZPLL). Figure 10 shows the implemented features and relevant signals.

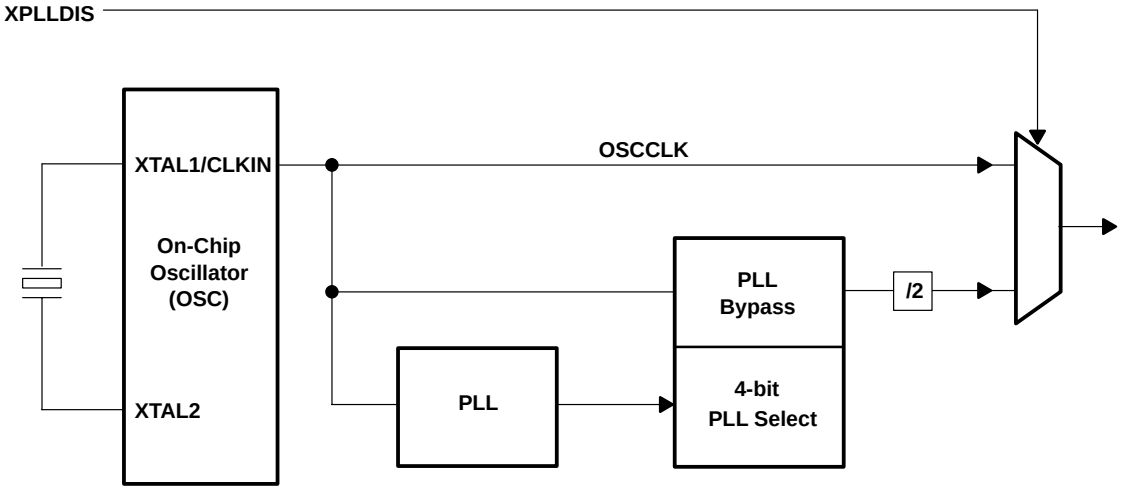


Figure 10. OSC and PLL Block

The OSC circuit enables a crystal to be attached to the F2810 and F2812 devices using the X1 and X2 pins. If a crystal is not used, then an external oscillator can be directly connected to the XCLKIN pin and the X2 pin is left unconnected. The oscillator input range is 20 MHz to 35 MHz.

Table 36. PLLCR Register Bit Definitions

BIT(S)	NAME	TYPE	$\overline{\text{XRS}}$ RESET†	DESCRIPTION
3:0	DIV	R/W	0,0,0,0	These bits set the PLL clocking ratio. The range of values should be between (x 1.0) to (x 10.0). The scale between these values should be as linear as possible: 0000 x 0.5 PLL bypassed but enabled CLKIN = OSCCLK / 2 0001 x 1.0 PLL connected CLKIN (OSCCLK * 1.0) / 2 0010 x 2.0 CLKIN = (OSCCLK * 2.0) / 2 0011 x 3.0 CLKIN = (OSCCLK * 3.0) / 2 0100 x 4.0 CLKIN = (OSCCLK * 4.0) / 2 0101 x 5.0 CLKIN = (OSCCLK * 5.0) / 2 0110 x 6.0 CLKIN = (OSCCLK * 6.0) / 2 0111 x 7.0 CLKIN = (OSCCLK * 7.0) / 2 1000 x 8.0 CLKIN = (OSCCLK * 8.0) / 2 1001 x 9.0 CLKIN = (OSCCLK * 9.0) / 2 1010 x 10.0 CLKIN = (OSCCLK * 10.0) / 2 1011 spare 1100 spare 1101 spare 1110 spare 1111 spare
15:4	reserved	R=0	0:0	

† The PLLCR register is reset to a known state by the $\overline{\text{XRS}}$ reset line. If a reset is issued by the debugger, the PLL clocking ratio is not changed.

PRODUCT PREVIEW

PLL-based clock module

The F2810 and F2812 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 to select different CPU clock rates.

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

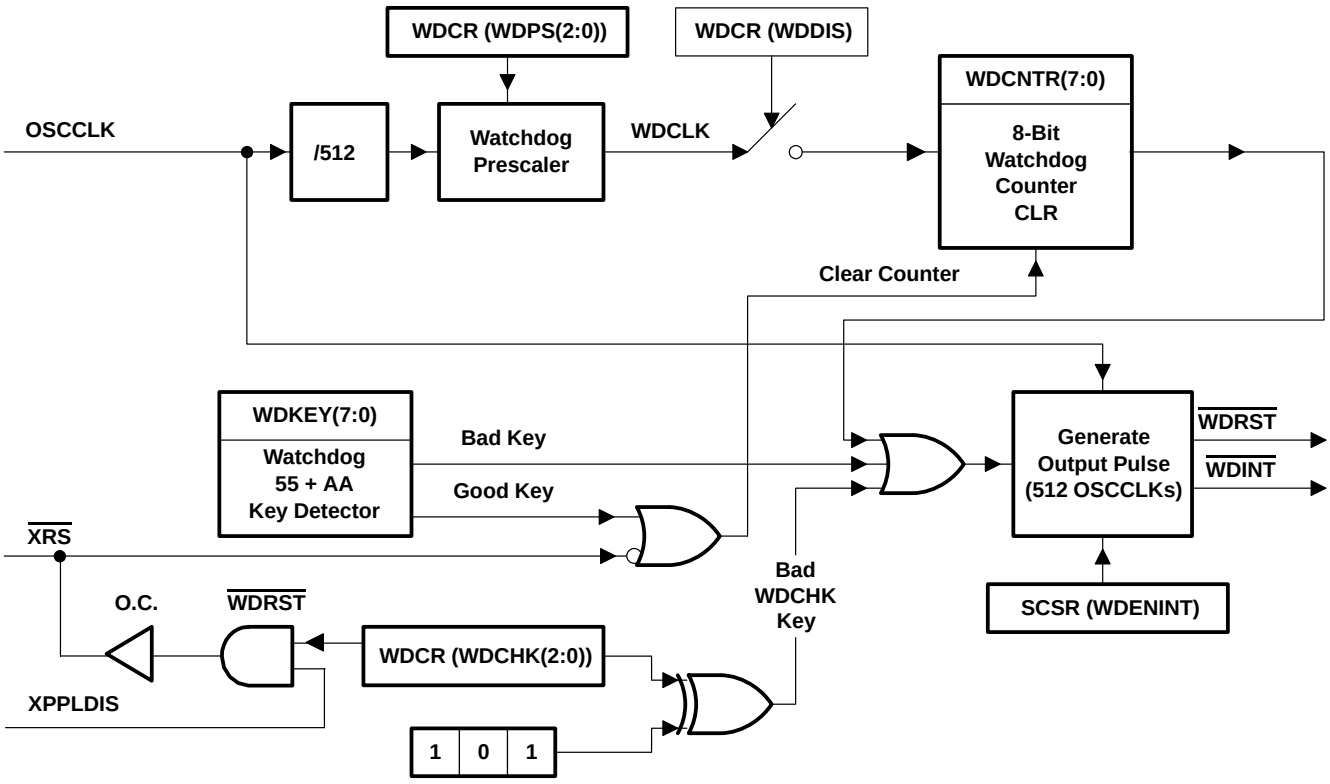
- Crystal-operation
  This mode allows the use of an external crystal/resonator to provide the time base to the device.
- External clock source operation
  This mode allows the internal oscillator to be bypassed. The device clocks are generated from an external clock source input on the XTAL1/CLKIN pin.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

external reference oscillator clock option

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

watchdog block

The watchdog block on the F2810 and F2812 is identical to the one used on the 240x 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 11 shows the various functional blocks within the watchdog module.



NOTE A: The $\overline{\text{WDRST}}$ signal is driven low for 512 OSCCLK cycles (similarly for the $\overline{\text{WDINT}}$ signal if enabled).

Figure 11. Watchdog Module

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

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 the PLL clock or the oscillator clock. The $\overline{\text{WDINT}}$ signal is fed to the LPM block so that it can wake the device from STANDBY (if enabled). Refer to ”Low-Power Modes Block” section of this data sheet 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.

watchdog block (continued)

Table 37. WDCNTR Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
7:0	WDCNTR	R/W	0:0	These bits contain the current value of the WD counter. The 8-bit counter continually increments at the WDCLK rate. If the counter overflows, then the watchdog initiates a reset. If the WDKEY register is written with a valid combination, then the counter is reset to zero.
15:8	reserved	R=0	0:0	

Table 38. WDKEY Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
7:0	WDKEY	W/R=0	0:0	Writing 0x55 followed by 0xAA will cause the WDCNTR bits to be cleared. Writing any other value will cause an immediate watchdog reset to be generated.
15:8	reserved	R=0	0:0	

Table 39. WDCR Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
2:0	WDPS(2:0)	R/W	0:0	These bits configure the watchdog counter clock (WDCLK) rate relative to OSCCLK/512: 000 WDCLK = OSCCLK/512/1 001 WDCLK = OSCCLK/512/1 010 WDCLK = OSCCLK/512/2 011 WDCLK = OSCCLK/512/4 100 WDCLK = OSCCLK/512/8 101 WDCLK = OSCCLK/512/16 110 WDCLK = OSCCLK/512/32 111 WDCLK = OSCCLK/512/64
5:3	WDCHK(2:0)	W/R=0	0:0	The user must ALWAYS write “1,0,1” to these bits whenever a write to this register is performed. Writing any other value will cause an immediate reset to the core (if WD enabled).
6	WDDIS	R/W	0	Writing a 1 to this bit will disable the watchdog module. Writing a 0 will enable the module. This bit can only be modified if the WDOVERRIDE bit in the SCSR2 register is set to 1. On reset, the watchdog module is enabled.
7	WDFLAG	R/W=1		Watchdog reset status flag bit. This bit, if set, indicates a watchdog reset ($\overline{\text{WDRST}}$) generated the reset condition. If 0, then it was an external device or power-up reset condition. This bit remains latched until the user writes a 1 to clear the condition. Writes of 0 will be ignored.
15:8	reserved	R=0	0:0	

PRODUCT PREVIEW

watchdog block (continued)

When the $\overline{\text{XRS}}$ line is low, the WDFLAG bit is forced low. The WDFLAG bit will only be set if a rising edge on $\overline{\text{WDRST}}$ signal is detected (after synch and a 4 cycle delay) and the $\overline{\text{XRS}}$ signal is high. If the $\overline{\text{XRS}}$ signal is low when $\overline{\text{WDRST}}$ goes high, then the WDFLAG bit will remain at 0. In a typical application, the $\overline{\text{WDRST}}$ signal will connect to the $\overline{\text{XRS}}$ input. Hence to distinguish between a watchdog reset and an external device reset, an external reset must be longer in duration then the watchdog pulse.

Emulation Considerations

The watchdog module behaves as follows under various debug conditions:

CPU Suspended:	When the CPU is suspended, the watchdog clock (WDCLK) is suspended.
Run-Free Mode:	When the CPU is placed in run-free mode, then the watchdog module resumes operation as normal.
Real-Time Single-Step Mode:	When the CPU is in real-time single-step mode, the watchdog clock (WDCLK) is suspended. The watchdog remains suspended even within real-time interrupts.
Real-Time Run-Free Mode:	When the CPU is in real-time run-free mode, the watchdog operates as normal.

PRODUCT PREVIEW

low-power modes block

The low-power modes on the F2810 and F2812 are similar to the 240x devices. Table 40 summarizes the various modes.

Table 40. F2810 and F2812 Low-Power Modes

MODE	IDLES	LPM(1:0)	OSCCLK	CLKIN	SYSCLKOUT	EXIT†
Normal	low	X,X	on	on	on	–
IDLE	high	0,0	on	on	on‡	$\overline{\text{XRS}}$, $\overline{\text{WDINT}}$, Any Enabled Interrupt, XNMI
STANDBY	high	0,1	on (watchdog still running)	off	off	$\overline{\text{XRS}}$, $\overline{\text{WDINT}}$, XINT1, XNMI, $\overline{\text{T1/2/3/4CTRIP}}$, $\overline{\text{C1/2/3/4/5/6TRIP}}$, SCIRXDA, SCIRXDB, CANRX, Debugger§
HALT	high	1,X	off (oscillator and PLL turned off, watchdog not functional)	off	off	$\overline{\text{XRS}}$, XNMI, Debugger§

† 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.

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

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

PRODUCT PREVIEW

The various low-power modes operate as follows:

IDLE Mode: This mode is, exited by any enabled interrupt or an NMI that is recognized by the processor. The LPM block performs no tasks during this mode as long as the LPMCR(LPM) bits are set to 0,0.

HALT Mode: Only the $\overline{\text{XRS}}$ and XNMI external signals can wake the device from HALT mode. The XNMI input to the core has an enable/disable bit. Hence, it is safe to use the XNMI signal for this function.

STANDBY Mode: All other signals (including XNMI) will wake the device from STANDBY mode if selected by the LPMCR1 register. The user will need to select which signal(s) will wake the device. The selected signal(s) are also qualified by the OSCCLK before waking the device. The number of OSCCLKs is specified in the LPMCR0 register.

low-power modes block (continued)

The low-power modes are controlled by the LPMCR0 register (see Table 41) and the LPMCR1 register (see Table 42).

Table 41. LPMCR0 Register Bit Definitions

BIT(S)	NAME	TYPE	RESET†	DESCRIPTION
1,0	LPM‡	R/W	0,0	These bits set the low power mode for the device.
7:2	QUALSTDBY	R/W	1:1	Select number of OSCCLK clock cycles to qualify the selected inputs when waking the LPM from STANDBY mode: 000000 = 2 OSCCLKs 000001 = 3 OSCCLKs . 111111 = 65 OSCCLKs
15:8	reserved	R=0	0:0	

† These bits are cleared by a reset ($\overline{\text{XRS}}$).
‡ The low power mode bits (LPM) are only valid when the IDLE instruction is executed. Therefore, the user must set the LPM bits to the appropriate mode before executing the IDLE instruction.

Table 42. LPMCR1 Register Bit Definitions

BIT(S)	NAME	TYPE	RESET†	DESCRIPTION
0	XINT1	R/W	0	If the respective bit is set to 1, it will enable the selected signal to wake the device from STANDBY mode. If the bit is cleared, the signal will have no effect.
1	XNMI	R/W	0	
2	$\overline{\text{WDINT}}$	R/W	0	
3	T1CTRIP	R/W	0	
4	T2CTRIP	R/W	0	
5	T3CTRIP	R/W	0	
6	T4CTRIP	R/W	0	
7	C1TRIP	R/W	0	
8	C2TRIP	R/W	0	
9	C3TRIP	R/W	0	
10	C4TRIP	R/W	0	
11	C5TRIP	R/W	0	
12	C6TRIP	R/W	0	
13	SCIRXA	R/W	0	
14	SCIRXB	R/W	0	
15	CANRX	R/W	0	

† These bits are cleared by a reset ($\overline{\text{XRS}}$).

PRODUCT PREVIEW

PERIPHERALS

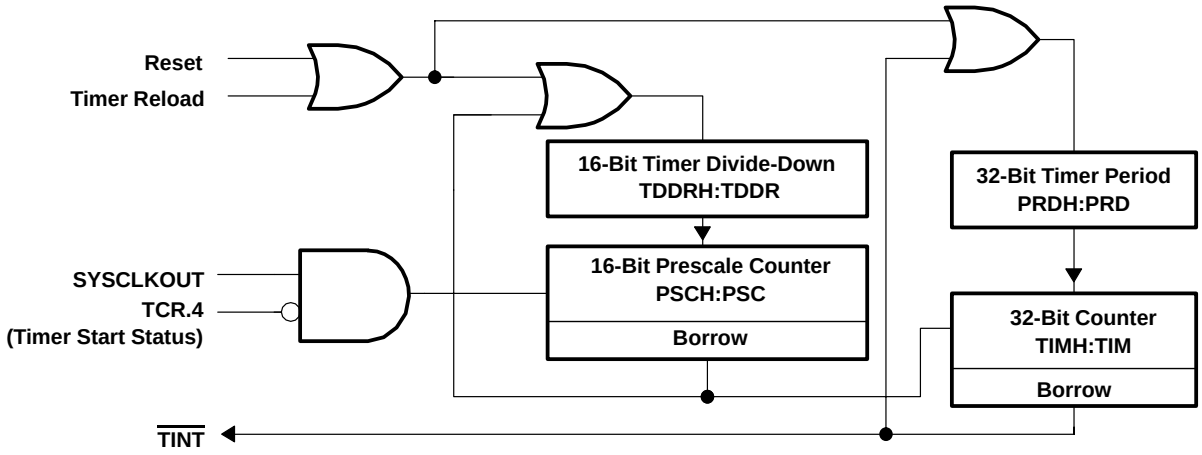
The integrated peripherals of the TMS320F2810 and TMS320F2812 are described in the following subsections:

- Three 32-bit CPU-Timers
- Two event-manager modules (EVA, EVB)
- Enhanced analog-to-digital converter (ADC) module
- Controller area network (CAN) module
- Serial communications interface modules (SCI-A, SCI-B)
- Serial peripheral interface (SPI) module
- PLL-based clock module
- Digital I/O and shared pin functions
- External memory interfaces (TMS320F2812 only)
- Watchdog (WD) timer module

32-bit CPU-Timers 0/1/2

This section describes the three 32-bit CPU-timers on the F2810 and F2812 devices (TIMER0/1/2).

CPU-Timers 1 and 2 are reserved for the Real-Time OS (such as DSP-BIOS).† CPU-Timer 0 can be used in user applications.



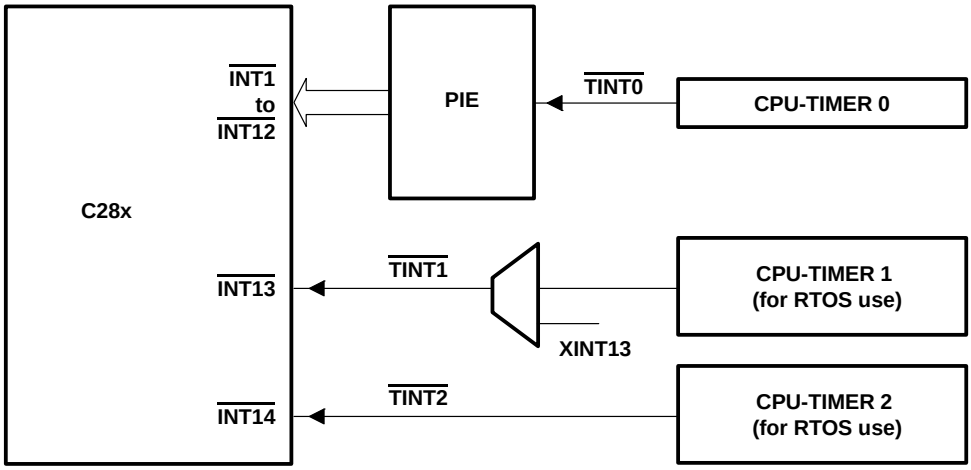
NOTE A: The CPU-Timers are different from the general-purpose (GP) timers that are present in the Event Manager modules (EVA, EVB).

Figure 12. CPU-Timers

† If the application is not using BIOS, then CPU-Timers 1 and 2 can be used in the application.

32-bit CPU-Timers 0/1/2 (continued)

In the F2810 and F2812 devices, the timer interrupt signals ($\overline{\text{TINT0}}$, $\overline{\text{TINT1}}$, $\overline{\text{TINT2}}$) are connected as shown in Figure 13.



NOTES: 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 13. CPU-Timer Interrupts 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 43 are used to configure the timers.

Table 43. CPU-Timers 0, 1, 2 Configuration and Control Registers

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
TIMER0TIM	0x0000–0C00	1	CPU-Timer 0, Counter Register
TIMER0TIMH	0x0000–0C01	1	CPU-Timer 0, Counter Register High
TIMER0PRD	0x0000–0C02	1	CPU-Timer 0, Period Register
TIMER0PRDH	0x0000–0C03	1	CPU-Timer 0, Period Register High
TIMER0TCR	0x0000–0C04	1	CPU-Timer 0, Control Register
reserved	0x0000–0C05	1	
TIMER0TPR	0x0000–0C06	1	CPU-Timer 0, Prescale Register
TIMER0TPRH	0x0000–0C07	1	CPU-Timer 0, Prescale Register High
TIMER1TIM	0x0000–0C08	1	CPU-Timer 1, Counter Register
TIMER1TIMH	0x0000–0C09	1	CPU-Timer 1, Counter Register High
TIMER1PRD	0x0000–0C0A	1	CPU-Timer 1, Period Register
TIMER1PRDH	0x0000–0C0B	1	CPU-Timer 1, Period Register High
TIMER1TCR	0x0000–0C0C	1	CPU-Timer 1, Control Register
reserved	0x0000–0C0D	1	
TIMER1TPR	0x0000–0C0E	1	CPU-Timer 1, Prescale Register
TIMER1TPRH	0x0000–0C0F	1	CPU-Timer 1, Prescale Register High

PRODUCT PREVIEW

32-bit CPU-Timers 0/1/2 (continued)

Table 43. CPU-Timers 0, 1, 2 Configuration and Control Registers (Continued)

NAME	ADDRESS	SIZE (x16)	DESCRIPTION
TIMER2TIM	0x0000–0C10	1	CPU-Timer 2, Counter Register
TIMER2TIMH	0x0000–0C11	1	CPU-Timer 2, Counter Register High
TIMER2PRD	0x0000–0C12	1	CPU-Timer 2, Period Register
TIMER2PRDH	0x0000–0C13	1	CPU-Timer 2, Period Register High
TIMER2TCR	0x0000–0C14	1	CPU-Timer 2, Control Register
reserved	0x0000–0C15	1	
TIMER2TPR	0x0000–0C16	1	CPU-Timer 2, Prescale Register
TIMER2TPRH	0x0000–0C17	1	CPU-Timer 2, Prescale Register High
reserved	0x0000–0C18 0x0000–0C3F	40	

Table 44. TIMERxTIM Register Bit Definitions†

BITS	NAME	R/W	RESET	DESCRIPTION
15:0	TIM	R/W	0xFFFF	Timer Counter Registers (TIMH:TIM): The TIM register holds the low 16 bits of the current 32-bit count of the timer. The TIMH register holds the high 16 bits of the current 32-bit count of the timer. The TIMH:TIM decrements by one every (TDDRH:TDDR+1) clock cycles, where TDDRH:TDDR is the timer prescale divide-down value. When the TIMH:TIM decrements to zero, the TIMH:TIM register is reloaded with the period value contained in the PRDH:PRD registers. The timer interrupt ($\overline{\text{TINT}}$) signal is generated.

† x = 0, 1, or 2

Table 45. TIMERxTIMH Register Bit Definitions†

BITS	NAME	R/W	RESET	DESCRIPTION
15:0	TIMH	R/W	0x0000	See description for TIMERxTIM.

† x = 0, 1, or 2

Table 46. TIMERxPRD Register Bit Definitions†

BITS	NAME	R/W	RESET	DESCRIPTION
15:0	PRD	R/W	0xFFFF	Timer Period Registers (PRDH:PRD): The PRD register holds the low 16 bits of the 32-bit period. The PRDH register holds the high 16 bits of the 32-bit period. When the TIMH:TIM decrements to zero, the TIMH:TIM register is reloaded with the period value contained in the PRDH:PRD registers, at the start of the next timer input clock cycle (the output of the prescaler). The PRDH:PRD contents are also loaded into the TIMH:TIM when you set the timer reload bit (TRB) in the Timer Control Register (TCR).

† x = 0, 1, or 2

Table 47. TIMERxPRDH Register Bit Definitions†

BITS	NAME	R/W	RESET	DESCRIPTION
15:0	PRDH	R/W	0x0000	See description for TIMERxPRD

† x = 0, 1, or 2

32-bit CPU-Timers 0/1/2 (continued)

Table 48. TIMERxTCR Register Bit Definitions†

BIT	NAME	R/W	RESET	DESCRIPTION
15	TIF	R/W=1	0	Timer Interrupt Flag. This flag gets set when the timer decrements to zero. This bit can be cleared by software writing a 1, but it can only be set by the timer reaching zero. Writing a 1 to this bit will clear it, writing a zero has no effect.
14	TIE	R/W	0	Timer Interrupt Enable. If the timer decrements to zero, and this bit is set, the timer will assert its interrupt request.
13:12	Reserved	R	0	Reserved
11	FREE	R/W	0	Timer Emulation Modes: These bits are special emulation bits that determine the state of the timer when a breakpoint is encountered in the high-level language debugger. If the FREE bit is set to 1, then, upon a software breakpoint, the timer continues to run (that is, free runs). In this case, SOFT is a *don't care*. But if FREE is 0, then SOFT takes effect. In this case, if SOFT = 0, the timer halts the next time the TIMH:TIM decrements. If the SOFT bit is 1, then the timer halts when the TIMH:TIM has decremented to zero. **FREE SOFT Timer Emulation Mode** 0 0 Stop after the next decrement of the TIMH:TIM (hard stop) 0 1 Stop after the TIMH:TIM decrements to 0 (soft stop) 1 0 Free run 1 1 Free run Note: That in the SOFT STOP mode, the timer will generate an interrupt before shutting down (since reaching 0 is the interrupt causing condition).
10	SOFT	R/W	0	
9:6	Reserved	R/W	0	Reserved
5	TRB	W/R=0	0	Timer Reload bit. When you write a 1 to TRB, the TIMH:TIM is loaded with the value in the PRDH:PRD, and the prescaler counter (PSCH:PSC) is loaded with the value in the timer divide-down register (TDDRH:TDDR). The TRB bit is always read as zero.
4	TSS	R/W	0	Timer stop status bit. TSS is a 1-bit flag that stops or starts the timer. To stop the timer, set TSS to 1. To start or restart the timer, set TSS to 0. At reset, TSS is cleared to 0 and the timer immediately starts.
3:0	Reserved	R/W	0	Reserved

† x = 0, 1, or 2

32-bit CPU-Timers 0/1/2 (continued)

Table 49. TIMERxTPR Register Bit Definitions†

BITS	NAME	R/W	RESET	DESCRIPTION
7:0	TDDR	R/W	0x00	Timer Divide-Down. Every (TDDRH:TDDR + 1) timer clock source cycles, the timer counter register (TIMH:TIM) decrements by one. At reset, the TDDRH:TDDR bits are cleared to 0. To increase the overall timer count by an integer factor, write this factor minus one to the TDDRH:TDDR bits. When the prescaler counter (PSCH:PSC) value is 0, one timer clock source cycle later, the contents of the TDDRH:TDDR reload the PSCH:PSC, and the TIMH:TIM decrements by one. TDDRH:TDDR also reloads the PSCH:PSC whenever the timer reload bit (TRB) is set by software.
15:8	PSC	R	0x00	Timer Prescale Counter. These bits hold the current prescale count for the timer. For every timer clock source cycle that the PSCH:PSC value is greater than 0, the PSCH:PSC decrements by one. One timer clock (output of the timer prescaler) cycle after the PSCH:PSC reaches 0, the PSCH:PSC is loaded with the contents of the TDDRH:TDDR, and the timer counter register (TIMH:TIM) decrements by one. The PSCH:PSC is also reloaded whenever the timer reload bit (TRB) is set by software. The PSCH:PSC can be checked by reading the register, but it cannot be set directly. It must get its value from the timer divide-down register (TDDRH:TDDR). At reset, the PSCH:PSC is set to 0.

† x = 0, 1, or 2

Table 50. TIMERxTPRH Register Bit Definitions†

BIT	NAME	R/W	RESET	DESCRIPTION
7:0	TDDRH	R/W	0x00	See description of TIMERxTPR.
15:8	PSCH	R	0x00	See description of TIMERxTPR.

† x = 0, 1, or 2

PRODUCT PREVIEW

event manager modules (EVA, EVB)

The event-manager modules include general-purpose (GP) timers, full-compare/PWM units, capture units, and quadrature-encoder pulse (QEP) circuits. EVA's and EVB's timers, compare units, and capture units function identically. However, timer/unit names differ for EVA and EVB. Table 51 shows the module and signal names used. Table 51 shows the features and functionality available for the event-manager modules and highlights EVA nomenclature.

Event managers A and B have identical peripheral register sets with EVA starting at 7400h and EVB starting at 7500h. The paragraphs in this section describe the function of GP timers, compare units, capture units, and QEPs using EVA nomenclature. These paragraphs are applicable to EVB with regard to function—however, module/signal names would differ.

Table 51. Module and Signal Names for EVA and EVB

EVENT MANAGER MODULES	EVA		EVB	
	MODULE	SIGNAL	MODULE	SIGNAL
GP Timers	GP Timer 1 GP Timer 2	T1PWM/T1CMP T2PWM/T2CMP	GP Timer 3 GP Timer 4	T3PWM/T3CMP T4PWM/T4CMP
Compare Units	Compare 1 Compare 2 Compare 3	PWM1/2 PWM3/4 PWM5/6	Compare 4 Compare 5 Compare 6	PWM7/8 PWM9/10 PWM11/12
Capture Units	Capture 1 Capture 2 Capture 3	CAP1 CAP2 CAP3	Capture 4 Capture 5 Capture 6	CAP4 CAP5 CAP6
QEP Channels	QEP1 QEP2 QEPI1	QEP1 QEP2	QEP3 QEP4 QEPI2	QEP3 QEP4
External Clock Inputs	Direction External Clock	TDIRA TCLKINA	Direction External Clock	TDIRB TCLKINB
External Compare Inputs	Compare	$\overline{\text{C1TRIP}}$ $\overline{\text{C2TRIP}}$ $\overline{\text{C3TRIP}}$		$\overline{\text{C4TRIP}}$ $\overline{\text{C5TRIP}}$ $\overline{\text{C6TRIP}}$
External Trip Inputs		$\overline{\text{T1CTRIP\_PDPINTA}}$† $\overline{\text{T2CTRIP/EVASOC}}$		$\overline{\text{T3CTRIP\_PDPINTB}}$† $\overline{\text{T4CTRIP/EVBSOC}}$

† In the 24x/240x-compatible mode, the $\overline{\text{T1CTRIP\_PDPINTA}}$ pin functions as $\overline{\text{PDPINTA}}$ and the $\overline{\text{T3CTRIP\_PDPINTB}}$ pin functions as $\overline{\text{PDPINTB}}$.

PRODUCT PREVIEW

event manager modules (EVA, EVB) (continued)

Table 52. EV-A Registers†

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
GPTCONA	0x0000–7400	1	GP Timer Control Register A
T1CNT	0x0000–7401	1	GP Timer 1 Counter Register
T1CMPR	0x0000–7402	1	GP Timer 1 Compare Register
T1PR	0x0000–7403	1	GP Timer 1 Period Register
T1CON	0x0000–7404	1	GP Timer 1 Control Register
T2CNT	0x0000–7405	1	GP Timer 2 Counter Register
T2CMPR	0x0000–7406	1	GP Timer 2 Compare Register
T2PR	0x0000–7407	1	GP Timer 2 Period Register
T2CON	0x0000–7408	1	GP Timer 2 Control Register
EXTCONA‡	0x0000–7409	1	GP Extension Control Register A
COMCONA	0x0000–7411	1	Compare Control Register A
ACTRA	0x0000–7413	1	Compare Action Control Register A
DBTCONA	0x0000–7415	1	Dead-Band Timer Control Register A
CMPR1	0x0000–7417	1	Compare Register 1
CMPR2	0x0000–7418	1	Compare Register 2
CMPR3	0x0000–7419	1	Compare Register 3
CAPCONA	0x0000–7420	1	Capture Control Register A
CAPFIFOA	0x0000–7422	1	Capture FIFO Status Register A
CAP1FIFO	0x0000–7423	1	Two-Level Deep Capture FIFO Stack 1
CAP2FIFO	0x0000–7424	1	Two-Level Deep Capture FIFO Stack 2
CAP3FIFO	0x0000–7425	1	Two-Level Deep Capture FIFO Stack 3
CAP1FBOT	0x0000–7427	1	Bottom Register Of Capture FIFO Stack 1
CAP2FBOT	0x0000–7428	1	Bottom Register Of Capture FIFO Stack 2
CAP2FBOT	0x0000–7429	1	Bottom Register Of Capture FIFO Stack 3
EVAIMRA	0x0000–742C	1	Interrupt Mask Register A
EVAIMRB	0x0000–742D	1	Interrupt Mask Register B
EVAIMRC	0x0000–742E	1	Interrupt Mask Register C
EVAIFRA	0x0000–742F	1	Interrupt Flag Register A
EVAIFRB	0x0000–7430	1	Interrupt Flag Register B
EVAIFRC	0x0000–7431	1	Interrupt Flag Register C

† The EV-B register set is identical except the address range is from 0x0000–7500 to 0x0000–753F. The above registers are mapped to Zone 2. This space allows only 16-bit accesses. 32-bit accesses produce undefined results.
‡ New register compared to 24x/240x

event manager modules (EVA, EVB) (continued)

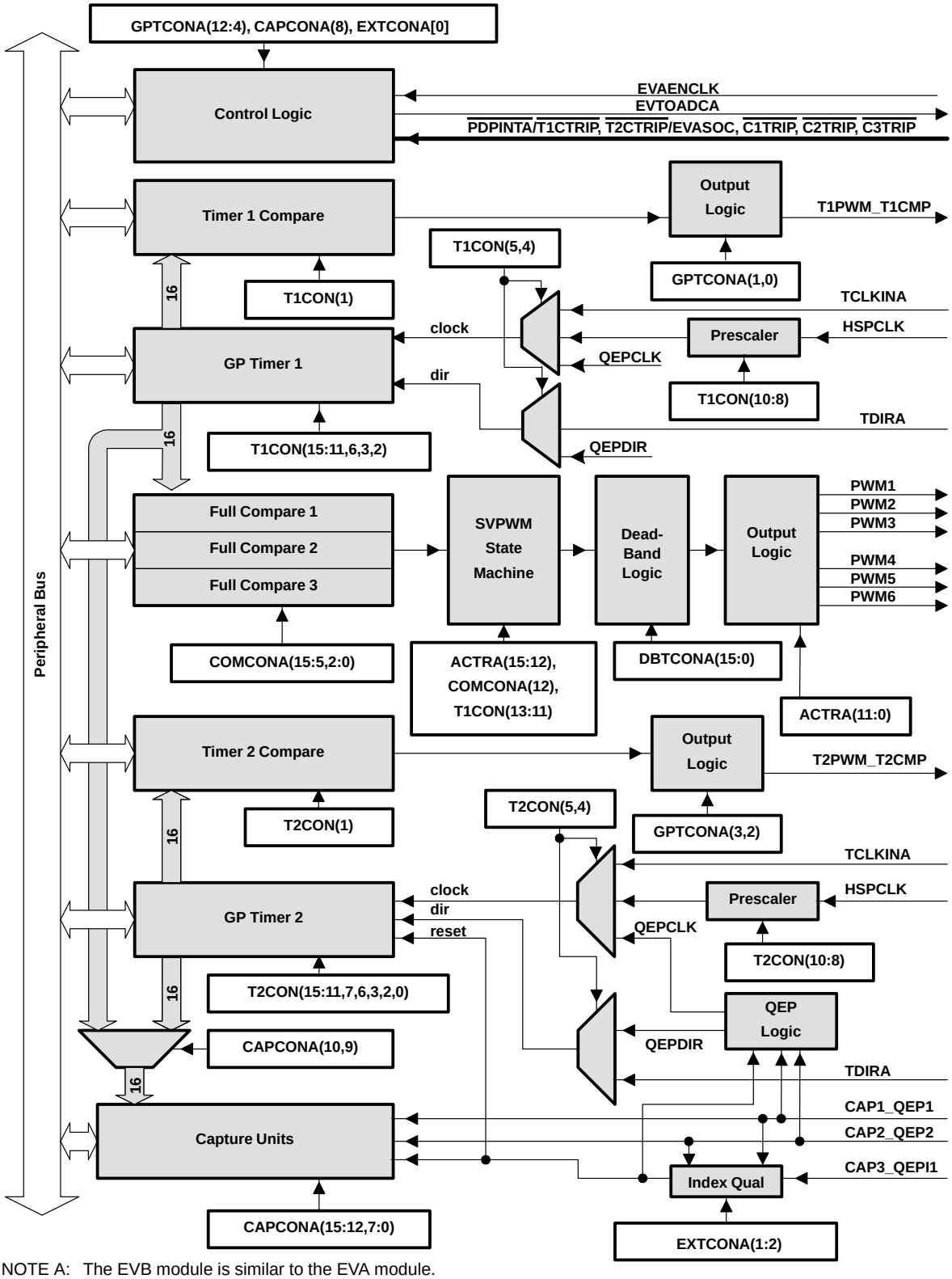
EXTCONA is an added control register to enable and disable the added/modified features. It is required for compatibility with 24x EV. EXTCONA enables and disables the additions and modifications in features. All additions and modifications are disabled by default to keep compatibility with 24x EV (see Table 53).

Table 53. EXTCONA Register Bit Definitions

BIT(S)	NAME	TYPE	RESET	DESCRIPTION
0	INDCOE	R/W	0	Independent Compare Output Enable Mode: This bit, when set to one, allows compare outputs to be enabled and disabled independently. 0 Independent Compare Output Enable mode is disabled. Time 1 and 2 compare outputs are enabled and disabled at the same time by GPTCONA(6). Full Compare 1, 2, and 3 outputs are enabled and disabled at the same time by COMCONA(9). GPTCONA(12,11,5,4) and COMCONA(7:5, 2:0) are reserved. EVIFRA(0) enables and disables all the compare outputs at the same time. EVIMR(0) enables and disables PDP interrupt and the direct path of $\overline{\text{PDPINT}}$ signal at the same time. 1 Independent Compare Output Enable mode is enabled. Compare outputs are enabled and disabled respectively by GPTCONA(5,4) and COMCONA(7:5). Compare trips are enabled and disabled respectively by GPTCONA(12,11) and COMCONA(2:0). GPTCONA(6) and COMCONA(9) are reserved. EVIFRA[0] is set to one when any trip input is low and is also enabled. EVIMRA(0) functions only as interrupt enable and disable.
1	QEPIQUAL	R/W	0	QEP/CAP3 Index Qualification Mode: This bit turns on and off QEP index qualifier. 0 QEPI/CAP3 qualification mode is off. QEPI/CAP3 is allowed to pass the qualifier unaffected. 1 QEPI/CAP3 qualification mode is on. A zero-to-one transition is allowed to pass the qualifier only when both QEPA and QEPB are high. Otherwise the output of the qualifier stays low.
2	QEPIE	R/W	0	QEP Index Enable: This bit enables and disables the QEPI input. The QEPI input when enabled can cause Timer 2 to reset: 0 Disable QEPI. Transitions on QEPI don't affect Timer 2. 1 Enable QEPI. Either a zero-to-one transition on QEPI alone (when EXTCONA[1] = 0), or a zero-to-one transition plus QEPA and QEPB are both high (when EXTCONA[1] = 1), causes Timer 2 to reset to zero.
3	EVSOCE	R/W	0	EV Start-of-Conversion Output Enable. This bit enables and disables the EV ADC start-of-conversion output. When enabled, a negative (active-low) pulse of 32 x HSPCLK is generated on selected EV ADC start-of-conversion event. This bit does not affect the EVTOADC signal routed to the ADC module as optional SOC trigger. 0 Disable EVSOC output. EVSOC is in Hi-Z state. 1 Enable EVSOC output.
15:4	reserved	R = 0	0:0	

PRODUCT PREVIEW

event manager modules (EVA, EVB) (continued)



NOTE A: The EVB module is similar to the EVA module.

Figure 14. Event Manager A Functional Block Diagram

PRODUCT PREVIEW

general-purpose (GP) timers

There are two GP timers. The GP timer x (x = 1 or 2 for EVA; x = 3 or 4 for EVB) includes:

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

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

full-compare units

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

programmable deadband generator

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

PWM waveform generation

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

PWM characteristics

Characteristics of the PWMs are as follows:

- 16-bit registers
- Wide range of programmable deadband for the PWM output pairs
- Change of the PWM carrier frequency for PWM frequency wobbling as needed
- Change of the PWM pulse widths within and after each PWM period as needed
- External-maskable power and drive-protection interrupts
- Pulse-pattern-generator circuit, for programmable generation of asymmetric, symmetric, and four-space vector PWM waveforms
- Minimized CPU overhead using auto-reload of the compare and period registers
- The PWM pins are driven to a high-impedance state when the $\overline{\text{PDPINTx}}$ pin is driven low and **after** $\overline{\text{PDPINTx}}$ signal qualification. The $\overline{\text{PDPINTx}}$ pin (after qualification) is reflected in bit 8 of the COMCONx register.
  - $\overline{\text{PDPINTA}}$ pin status is reflected in bit 8 of COMCONA register.
  - $\overline{\text{PDPINTB}}$ pin status is reflected in bit 8 of COMCONB register.

capture unit

The capture unit provides a logging function for different events or transitions. The values of the selected GP timer counter is captured and stored in the two-level-deep FIFO stacks when selected transitions are detected on capture input pins, CAPx (x = 1, 2, or 3 for EVA; and x = 4, 5, or 6 for EVB). The capture unit consists of three capture circuits.

- Capture units include the following features:
  - One 16-bit capture control register, CAPCONx (R/W)
  - One 16-bit capture FIFO status register, CAPFIFOx
  - Selection of GP timer 1/2 (for EVA) or 3/4 (for EVB) as the time base
  - Three 16-bit 2-level-deep FIFO stacks, one for each capture unit
  - Three capture input pins (CAP1/2/3 for EVA, CAP4/5/6 for EVB)—one input pin per capture unit. [All inputs are synchronized with the device (CPU) clock. In order for a transition to be captured, the input must hold at its current level to meet two rising edges of the device clock. The input pins CAP1/2 and CAP4/5 can also be used as QEP inputs to the QEP circuit.]
  - User-specified transition (rising edge, falling edge, or both edges) detection
  - Three maskable interrupt flags, one for each capture unit

quadrature-encoder pulse (QEP) circuit

Two capture inputs (CAP1 and CAP2 for EVA; CAP4 and CAP5 for EVB) can be used to interface the on-chip QEP circuit with a quadrature encoder pulse. Full synchronization of these inputs is performed on-chip. Direction or leading-quadrature pulse sequence is detected, and GP timer 2/4 is incremented or decremented by the rising and falling edges of the two input signals (four times the frequency of either input pulse).

external ADC start-of-conversion

EVA/EVB start-of-conversion (SOC) can be sent to an external pin (ESOCA/B) for external ADC interface. EVASOC and EVBSOC are muxed with $\overline{\text{T2CTRIP}}$ and $\overline{\text{T4CTRIP}}$, respectively.

enhanced analog-to-digital converter (ADC) module

A simplified functional block diagram of the ADC module is shown in Figure 15. 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 V to 2.5 V
- Fast conversion time:
  - Single conversion time: 200 ns
  - Pipelined conversion time: 60 ns
- 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:

$$\text{Digital Value} = 4095 \times \frac{\text{Input Analog Voltage} - \text{ADCLO}}{2.5}$$

- Multiple triggers as sources for the start-of-conversion (SOC) sequence
  - S/W – software immediate start
  - EVA – Event manager A (multiple event sources within EVA)
  - EVB – Event manager B (multiple event sources within EVB)
- 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
- EVA and EVB triggers can operate independently in dual-sequencer mode
- Sample-and-hold (S/H) acquisition time window has separate prescale control
- Calibration mode

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

The ADC module in the F2810 and F2812 has been enhanced to provide flexible interface to event managers A and B. The ADC interface is built around a fast, 12-bit ADC module with a total minimum conversion time of 200 ns (S/H + conversion) per conversion. The ADC module has 16 channels, configurable as two independent 8-channel modules to service event managers A and B. 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 15 shows the block diagram of the F2810 and F2812 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.

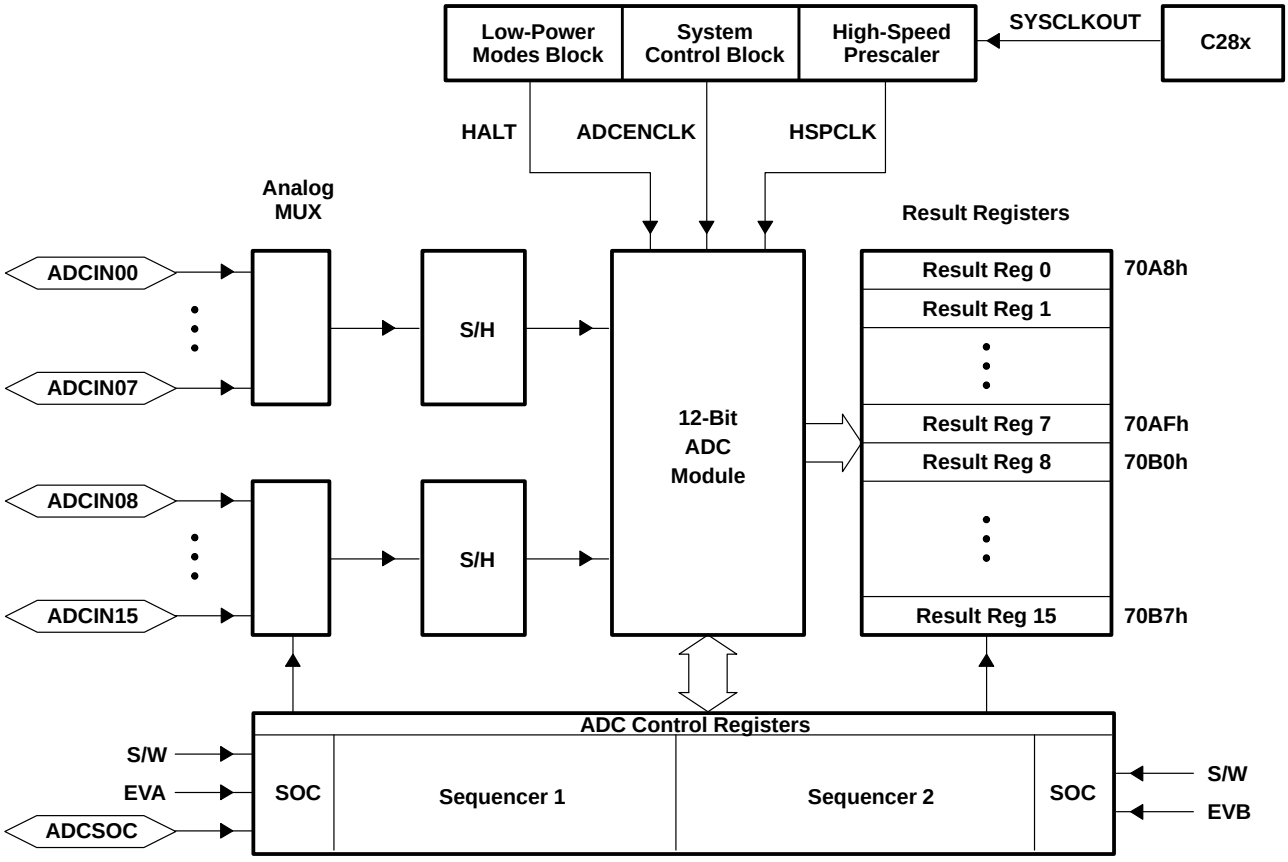


Figure 15. Block Diagram of the F2810 and F2812 ADC Module

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

To obtain the specified accuracy of the ADC, proper board layout is very critical. To the best extent possible, traces leading to the $\overline{\text{ADCIN}}$ 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 (such as V_{CCA}, V_{REFHI}, and V_{SSA}) from the digital supply.

Notes:

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 ($\overline{\text{XRS}}$) 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 will however 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 signal only affects the analog module. It does not affect the registers. If low, the ADC module is powered. If high, the ADC module goes into low-power mode. The HALT mode will stop the clock to the CPU, which will stop the HSPCLK. Therefore the ADC register logic will be turned off indirectly.

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

The ADC operation is configured, controlled, and monitored by the registers listed in Table 54.

Table 54. ADC Registers†

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
ADCTRL1	0x0000–7100	1	ADC Control Register 1
ADCTRL2	0x0000–7101	1	ADC Control Register 2
ADCMAXCONV	0x0000–7102	1	ADC Maximum Conversion Channels Register
ADCCHSELSEQ1	0x0000–7103	1	ADC Channel Select Sequencing Control Register 1
ADCCHSELSEQ2	0x0000–7104	1	ADC Channel Select Sequencing Control Register 2
ADCCHSELSEQ3	0x0000–7105	1	ADC Channel Select Sequencing Control Register 3
ADCCHSELSEQ4	0x0000–7106	1	ADC Channel Select Sequencing Control Register 4
ADCASEQSR	0x0000–7107	1	ADC Auto–Sequence Status Register
ADCRESULT0	0x0000–7108	1	ADC Conversion Result Buffer Register 0
ADCRESULT1	0x0000–7109	1	ADC Conversion Result Buffer Register 1
ADCRESULT2	0x0000–710A	1	ADC Conversion Result Buffer Register 2
ADCRESULT3	0x0000–710B	1	ADC Conversion Result Buffer Register 3
ADCRESULT4	0x0000–710C	1	ADC Conversion Result Buffer Register 4
ADCRESULT5	0x0000–710D	1	ADC Conversion Result Buffer Register 5
ADCRESULT6	0x0000–710E	1	ADC Conversion Result Buffer Register 6
ADCRESULT7	0x0000–710F	1	ADC Conversion Result Buffer Register 7
ADCRESULT8	0x0000–7110	1	ADC Conversion Result Buffer Register 8
ADCRESULT9	0x0000–7111	1	ADC Conversion Result Buffer Register 9
ADCRESULT10	0x0000–7112	1	ADC Conversion Result Buffer Register 10
ADCRESULT11	0x0000–7113	1	ADC Conversion Result Buffer Register 11
ADCRESULT12	0x0000–7114	1	ADC Conversion Result Buffer Register 12
ADCRESULT13	0x0000–7115	1	ADC Conversion Result Buffer Register 13
ADCRESULT14	0x0000–7116	1	ADC Conversion Result Buffer Register 14
ADCRESULT15	0x0000–7117	1	ADC Conversion Result Buffer Register 15
ADCCALOFF0	0x0000–7118	1	ADC Calibration Offset Result 0
ADCCALOFF1	0x0000–7119	1	ADC Calibration Offset Result 1
ADCTRL3	0x0000–711A	1	ADC Control Register 3
ADCST	0x0000–711B	1	ADC Status Register
reserved	0x0000–711C 0x0000–711F	4	

† The above registers are mapped to peripheral bus 16 space. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.

enhanced controller area network (eCAN) module

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.

enhanced controller area network (eCAN) module (continued)

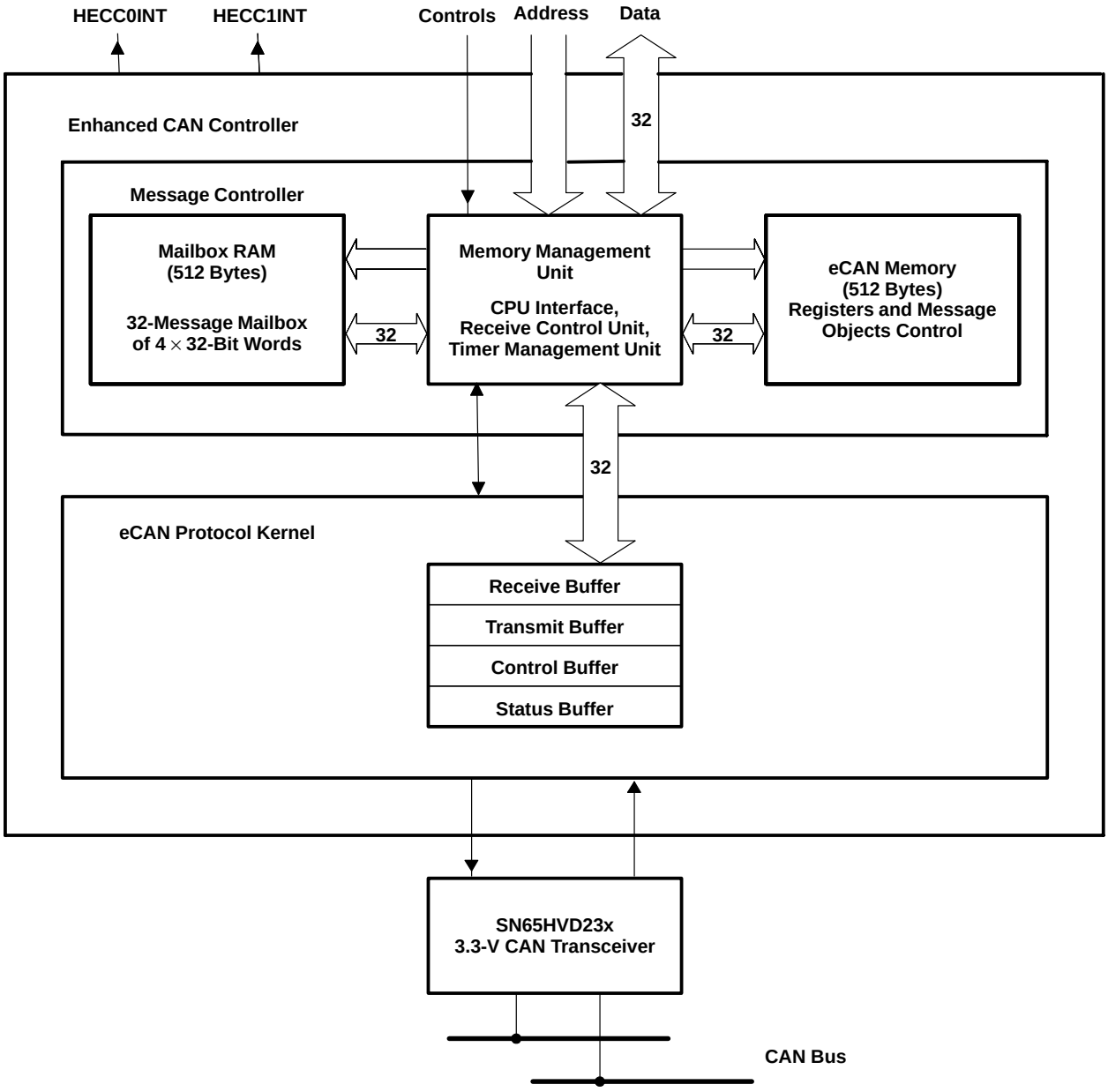


Figure 16. eCAN Block Diagram and Interface Circuit

enhanced controller area network (eCAN) module (continued)

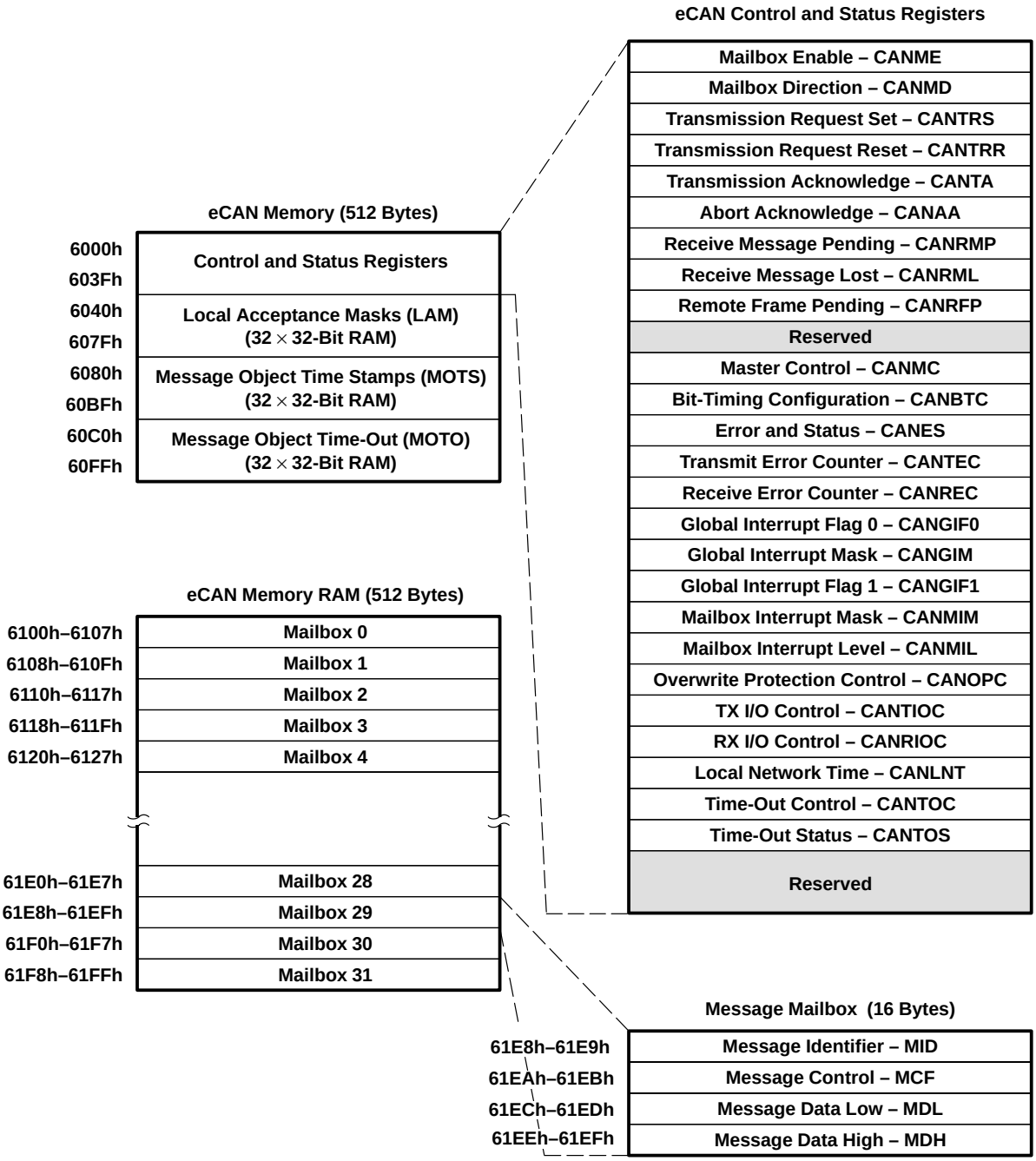


Figure 17. eCAN Memory Map

enhanced controller area network (eCAN) module (continued)

The CAN registers listed in Table 55 are used by the CPU to configure and control the CAN controller and the message objects.

Table 55. CAN Registers Map†

REGISTER NAME	ADDRESS	DESCRIPTION
CANME	0x00 6000	Mailbox enable
CANMD	0x00 6002	Mailbox direction
CANTRS	0x00 6004	Transmit request set
CANTRR	0x00 6006	Transmit request reset
CANTA	0x00 6008	Transmission acknowledge
CANAA	0x00 600A	Abort acknowledge
CANRMP	0x00 600C	Receive message pending
CANRML	0x00 600E	Receive message lost
CANRFP	0x00 6010	Remote frame pending
Reserved	0x00 6012	Reserved
CANMC	0x00 6014	Master control
CANBTC	0x00 6016	Bit-timing configuration
CANES	0x00 6018	Error and status
CANTEC	0x00 601A	Transmit error counter
CANREC	0x00 601C	Receive error counter
CANGIF0	0x00 601E	Global interrupt flag 0
CANGIM	0x00 6020	Global interrupt mask
CANGIF1	0x00 6022	Global interrupt flag 1
CANMIM	0x00 6024	Mailbox interrupt mask
CANMIL	0x00 6026	Mailbox interrupt level
CANOPC	0x00 6028	Overwrite protection control
CANTIOC	0x00 602A	TX I/O control
CANRIOC	0x00 602C	RX I/O control
CANLNT	0x00 602E	Local network time (Reserved in SCC mode)
CANTOC	0x00 6030	Time-out control (Reserved in SCC mode)
CANTOS	0x00 6032	Time-out status (Reserved in SCC mode)

† These registers are mapped to Peripheral Frame 1. This space allows 16-bit and 32-bit accesses.

multichannel buffered serial port (McBSP) module

The McBSP module has the following features:

- Compatible to McBSP in TMS320C54x™ /TMS320C55x™ DSP devices
- Full-duplex communication
- Double-buffered data registers which 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
- Support A-bis mode
- 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 at 75 Mbps maximum for 150-MHz SYSCLKOUT
- Two 16 x 16-level FIFO for Transmit channel
- Two 16 x 16-level FIFO for Receive channel

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

TMS320C54x and TMS320C55x are trademarks of Texas Instruments.

multichannel buffered serial port (McBSP) module (continued)

Figure 18 shows the block diagram of the McBSP module with FIFO, interfaced to the F2810 and F2812 version of Peripheral Frame 2.

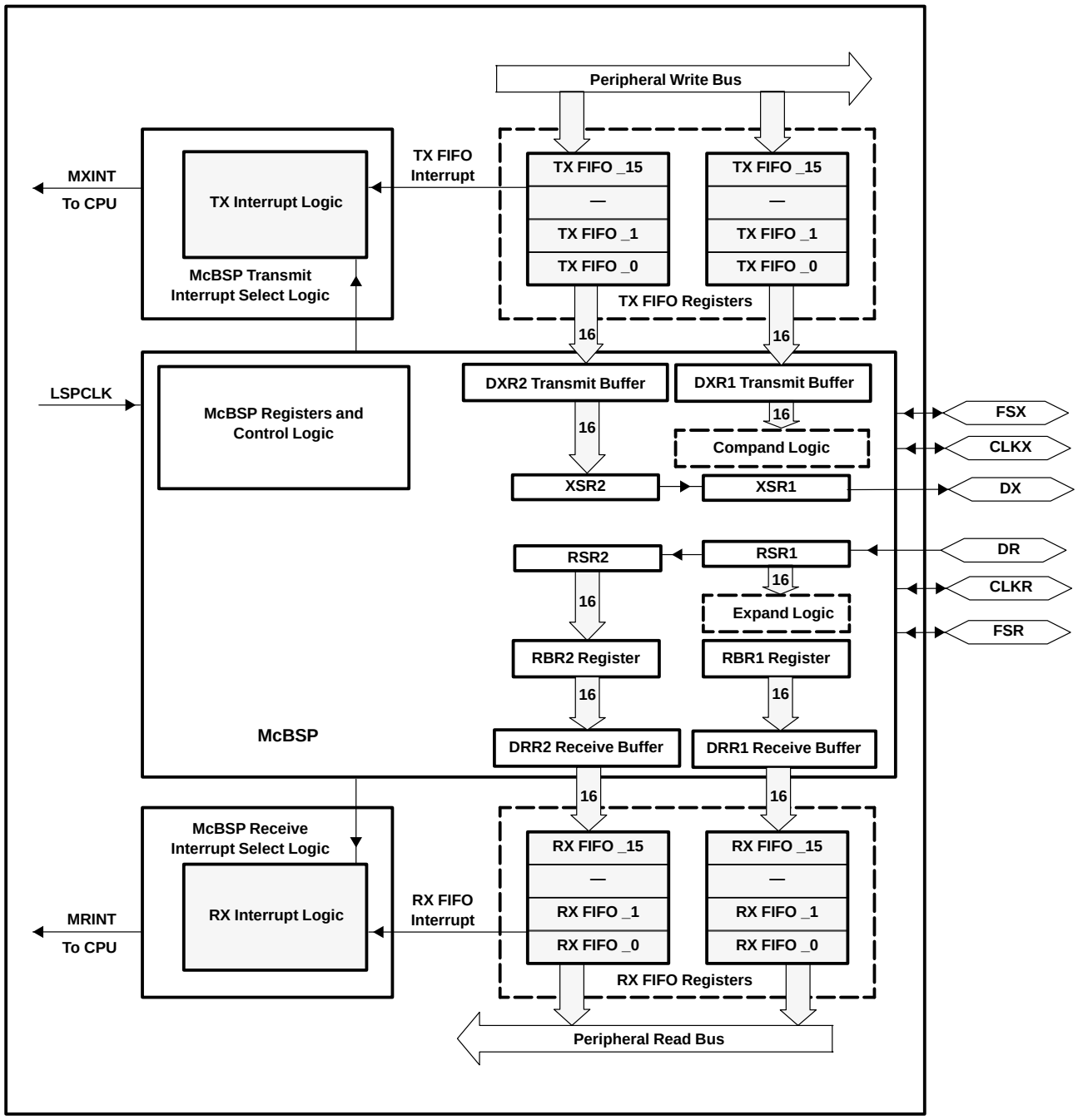


Figure 18. McBSP Module With FIFO

multichannel buffered serial port (McBSP) module (continued)

Table 56 provides a summary of the McBSP registers.

Table 56. McBSP Register Summary

NAME	ADDRESS 0x000 xxh	TYPE (R/W)	RESET VALUE (HEX)	DESCRIPTION
DATA REGISTERS, RECEIVE, TRANSMIT†				
–	–	–	0x0000	McBSP Receive Buffer Register
–	–	–	0x0000	McBSP Receive Shift Register
–	–	–	0x0000	McBSP Transmit Shift Register
DRR2	00	R	0x0000	McBSP Data Receive Register 2 – Read First if the word size is greater than 16 bits, else ignore DRR2
DRR1	01	R	0x0000	McBSP Data Receive Register 1 – Read Second if the word size is greater than 16 bits, else read DRR1 only
DXR2	02	W	0x0000	McBSP Data Transmit Register 2 – Write First if the word size is greater than 16 bits, else ignore DXR2
DXR1	03	W	0x0000	McBSP Data Transmit Register 1 – Write Second if the word size is greater than 16 bits, else write to DXR1 only
McBSP CONTROL REGISTERS				
SPCR2	04	R/W	0x0000	McBSP Serial Port Control Register 2
SPCR1	05	R/W	0x0000	McBSP Serial Port Control Register 1
RCR2	06	R/W	0x0000	McBSP Receive Control Register 2
RCR1	07	R/W	0x0000	McBSP Receive Control Register 1
XCR2	08	R/W	0x0000	McBSP Transmit Control Register 2
XCR1	09	R/W	0x0000	McBSP Transmit Control Register 1
SRGR2	0A	R/W	0x0000	McBSP Sample Rate Generator Register 2
SRGR1	0B	R/W	0x0000	McBSP Sample Rate Generator Register 1
MULTICHANNEL CONTROL REGISTERS				
MCR2	0C	R/W	0x0000	McBSP Multichannel Register 2
MCR1	0D	R/W	0x0000	McBSP Multichannel Register 1
RCERA	0E	R/W	0x0000	McBSP Receive Channel Enable Register Partition A
RCERB	0F	R/W	0x0000	McBSP Receive Channel Enable Register Partition B
XCERA	10	R/W	0x0000	McBSP Transmit Channel Enable Register Partition A
XCERB	11	R/W	0x0000	McBSP Transmit Channel Enable Register Partition B
PCR1	12	R/W	0x0000	McBSP Pin Control Register
RCERC	13	R/W	0x0000	McBSP Receive Channel Enable Register Partition C
RCERD	14	R/W	0x0000	McBSP Receive Channel Enable Register Partition D
XCERC	15	R/W	0x0000	McBSP Transmit Channel Enable Register Partition C
XCERD	16	R/W	0x0000	McBSP Transmit Channel Enable Register Partition D

† DRR2/DRR1 and DXR2/DXR1 share the same addresses of receive and transmit FIFO registers in FIFO mode.
‡ FIFO pointers advancing is based on order of access to DRR2/DRR1 and DXR2/DXR1 registers.

multichannel buffered serial port (McBSP) module (continued)

Table 56. McBSP Register Summary (Continued)

NAME	ADDRESS 0x000 xxh	TYPE (R/W)	RESET VALUE (HEX)	DESCRIPTION
MULTICHANNEL CONTROL REGISTERS (CONTINUED)				
RCERE	17	R/W	0x0000	McBSP Receive Channel Enable Register Partition E
RCERF	18	R/W	0x0000	McBSP Receive Channel Enable Register Partition F
XCERE	19	R/W	0x0000	McBSP Transmit Channel Enable Register Partition E
XCERF	1A	R/W	0x0000	McBSP Transmit Channel Enable Register Partition F
RCERG	1B	R/W	0x0000	McBSP Receive Channel Enable Register Partition G
RCERH	1C	R/W	0x0000	McBSP Receive Channel Enable Register Partition H
XCERG	1D	R/W	0x0000	McBSP Transmit Channel Enable Register Partition G
XCERH	1E	R/W	0x0000	McBSP Transmit Channel Enable Register Partition H
FIFO MODE REGISTERS (applicable only in FIFO mode)				
FIFO Data Registers‡				
DRR2	00	R	0x0000	McBSP Data Receive Register 2 – Top of receive FIFO – Read First FIFO pointers will not advance
DRR1	01	R	0x0000	McBSP Data Receive Register 1 – Top of receive FIFO – Read Second for FIFO pointers to advance
DXR2	02	W	0x0000	McBSP Data Transmit Register 2 – Top of transmit FIFO – Write First FIFO pointers will not advance
DXR1	03	W	0x0000	McBSP Data Transmit Register 1 – Top of transmit FIFO – Write Second for FIFO pointers to advance
FIFO Control Registers				
MFFTX	20	R/W	0xA000	McBSP Transmit FIFO Register
MFFRX	21	R/W	0x201F	McBSP Receive FIFO Register
MFFCT	22	R/W	0x0000	McBSP FIFO Control Register
MFFINT	23	R/W	0x0000	McBSP FIFO Interrupt Register
MFFST	24	R/W	0x0000	McBSP FIFO Status Register

† DRR2/DRR1 and DXR2/DXR1 share the same addresses of receive and transmit FIFO registers in FIFO mode.
‡ FIFO pointers advancing is based on order of access to DRR2/DRR1 and DXR2/DXR1 registers.

serial communications interface (SCI) module

The F2810 and F2812 devices include two 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 the SCI module include:

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

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

- Baud rate programmable to 64K different rates
  - Up to 9.3 Mbps at 150-MHz SYSCLKOUT
- 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)
- 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

serial communications interface (SCI) module (continued)

The SCI port operation is configured and controlled by the registers listed in Table 57 and Table 58.

Table 57. SCI-A Registers†

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
SCICCR	0x0000–7050	1	SCI-A Communications Control Register
SCICTL1	0x0000–7051	1	SCI-A Control Register 1
SCIHBAUD	0x0000–7052	1	SCI-A Baud Register, High Bits
SCILBAUD	0x0000–7053	1	SCI-A Baud Register, Low Bits
SCICTL2	0x0000–7054	1	SCI-A Control Register 2
SCIRXST	0x0000–7055	1	SCI-A Receive Status Register
SCIRXEMU	0x0000–7056	1	SCI-A Receive Emulation Data Buffer Register
SCIRXBUF	0x0000–7057	1	SCI-A Receive Data Buffer Register
SCITXBUF	0x0000–7059	1	SCI-A Transmit Data Buffer Register
SCIFFTX	0x0000–705A	1	SCI-A FIFO Transmit Register
SCIFFRX	0x0000–705B	1	SCI-A FIFO Receive Register
SCIFFCT	0x0000–705C	1	SCI-A FIFO Control Register
SCIPRI	0x0000–705F	1	SCI-A Priority Control Register

† Shaded registers are new registers for the FIFO mode.

Table 58. SCI-B Registers†

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
SCICCR	0x0000–7750	1	SCI-B Communications Control Register
SCICTL1	0x0000–7751	1	SCI-B Control Register 1
SCIHBAUD	0x0000–7752	1	SCI-B Baud Register, High Bits
SCILBAUD	0x0000–7753	1	SCI-B Baud Register, Low Bits
SCICTL2	0x0000–7754	1	SCI-B Control Register 2
SCIRXST	0x0000–7755	1	SCI-B Receive Status Register
SCIRXEMU	0x0000–7756	1	SCI-B Receive Emulation Data Buffer Register
SCIRXBUF	0x0000–7757	1	SCI-B Receive Data Buffer Register
SCITXBUF	0x0000–7759	1	SCI-B Transmit Data Buffer Register
SCIFFTX	0x0000–775A	1	SCI-B FIFO Transmit Register
SCIFFRX	0x0000–775B	1	SCI-B FIFO Receive Register
SCIFFCT	0x0000–775C	1	SCI-B FIFO Control Register
SCIPRI	0x0000–775F	1	SCI-B Priority Control Register

† Shaded registers are new registers for the FIFO mode.

Note:
The above registers are mapped to peripheral bus 16 space. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.

serial communications interface (SCI) module (continued)

Figure 19 shows the SCI module block diagram.

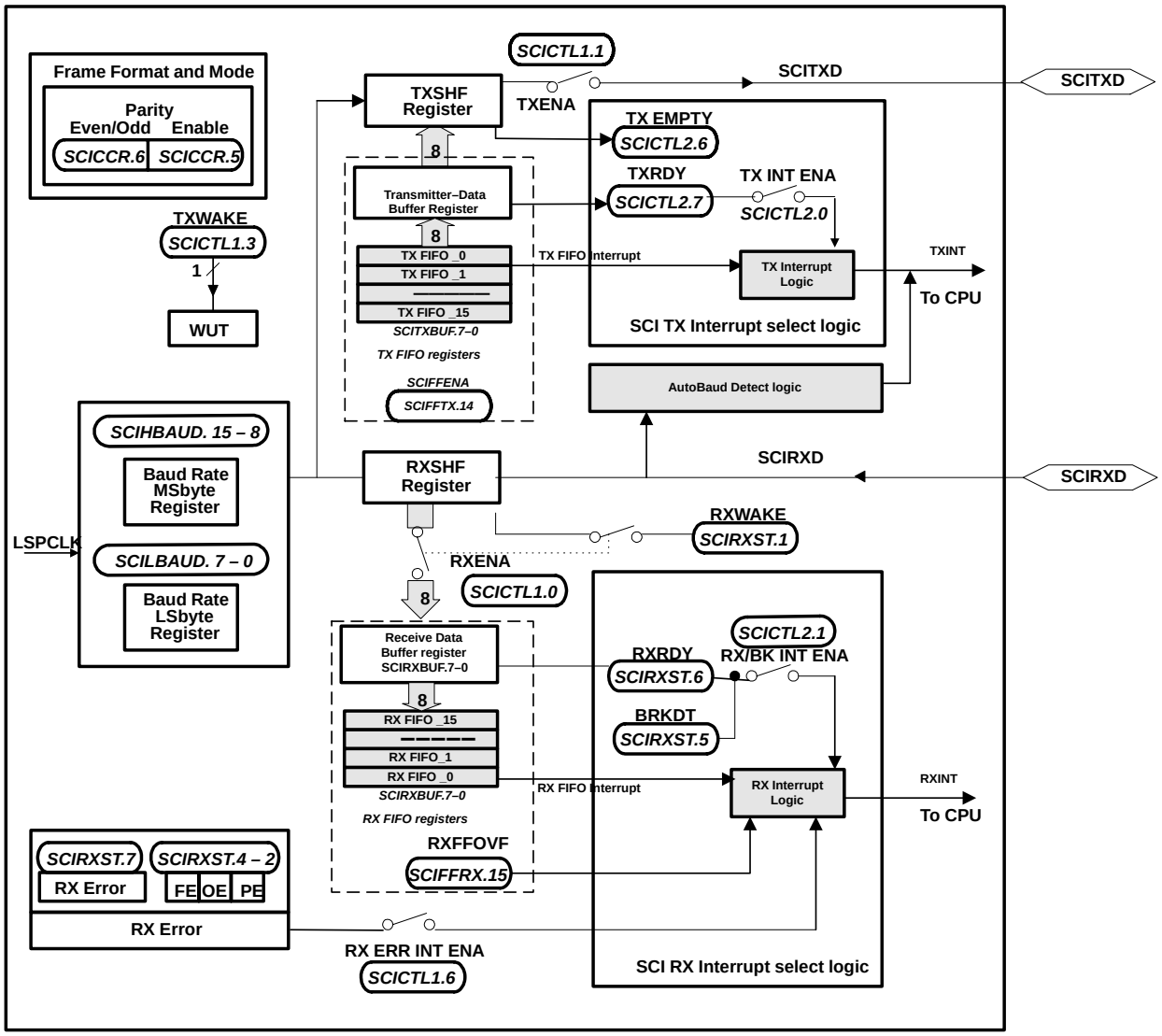


Figure 19. Serial Communications Interface (SCI) Module Block Diagram

serial peripheral interface (SPI) module

The F2810 and F2812 devices include the four-pin serial peripheral interface (SPI) module. The SPI is a high-speed, synchronous serial I/O port that allows a serial bit stream of programmed length (one to 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 DSP controller and external peripherals or another processor. Typical applications include external I/O or peripheral expansion through devices such as shift registers, display drivers, and ADCs. Multidevice communications are supported by the master/slave operation of the SPI.

The SPI module features include:

- 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/37.5 Mbps at 150-MHz SYSCLKOUT
- 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

serial peripheral interface (SPI) module (continued)

The SPI port operation is configured and controlled by the registers listed in Table 59.

Table 59. SPI Registers

NAME	ADDRESS RANGE	SIZE (x16)	DESCRIPTION
SPICCR	0x0000–7040	1	SPI Configuration Control Register
SPICTL	0x0000–7041	1	SPI Operation Control Register
SPIST	0x0000–7042	1	SPI Status Register
SPIBRR	0x0000–7044	1	SPI Baud Rate Register
SPIEMU	0x0000–7046	1	SPI Emulation Buffer Register
SPIRXBUF	0x0000–7047	1	SPI Serial Input Buffer Register
SPITXBUF	0x0000–7048	1	SPI Serial Output Buffer Register
SPIDAT	0x0000–7049	1	SPI Serial Data Register
SPIFFTX	0x0000–704A	1	SPI FIFO Transmit Register
SPIFFRX	0x0000–704B	1	SPI FIFO Receive Register
SPIFFCT	0x0000–704C	1	SPI FIFO Control Register
SPIPRI	0x0000–704F	1	SPI Priority Control Register

Note:
The above registers are mapped to Peripheral Frame 2. This space only allows 16-bit accesses. 32-bit accesses produce undefined results.

serial peripheral interface (SPI) module (continued)

Figure 20 is a block diagram of the SPI in slave mode.

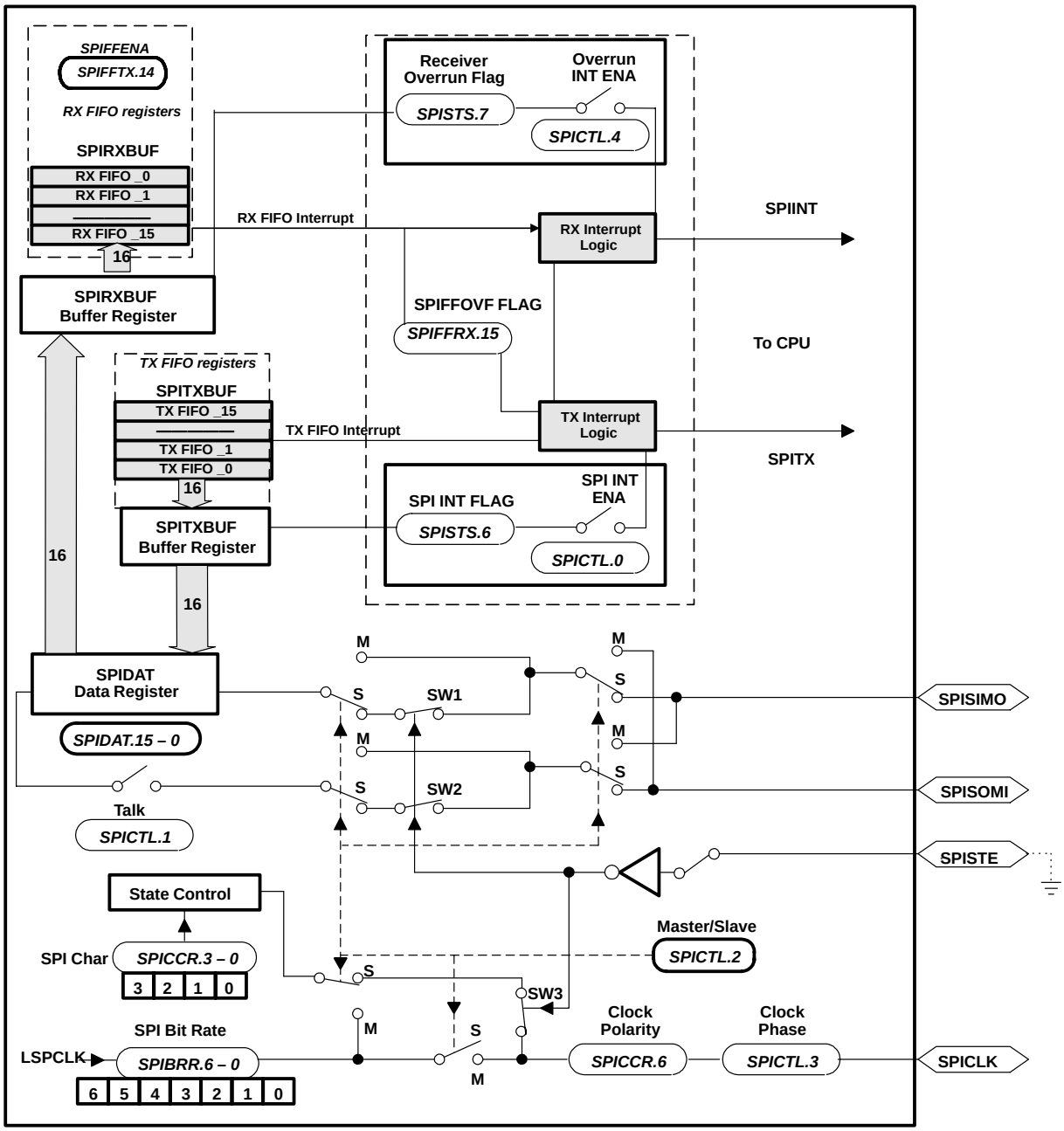


Figure 20. Serial Peripheral Interface Module Block Diagram

GPIO mux

The GPIO Mux registers, are used to select the operation of shared pins on the F2810 and F2812 devices. The pins can be individually selected to operate as “Digital I/O” or connected to “Peripheral I/O” signals (via the GPxMUX registers). If selected for “Digital I/O” mode, registers are provided to configure the pin direction (via the GPxDIR registers) and to qualify the input signal to remove unwanted noise (via the GPxQUAL) registers). Table 60 lists the GPIO Mux Registers.

Table 60. GPIO Mux Registers†‡§

NAME	ADDRESS	SIZE (x16)	REGISTER DESCRIPTION
GPAMUX	0x0000–70C0	1	GPIO A Mux Control Register
GPADIR	0x0000–70C1	1	GPIO A Direction Control Register
GPAQUAL	0x0000–70C2	1	GPIO A Input Qualification Control Register
reserved	0x0000–70C3	1	
GPBMUX	0x0000–70C4	1	GPIO B Mux Control Register
GPBDIR	0x0000–70C5	1	GPIO B Direction Control Register
GPBQUAL	0x0000–70C6	1	GPIO B Input Qualification Control Register
reserved	0x0000–70C7	1	
reserved	0x0000–70C8	1	
reserved	0x0000–70C9	1	
reserved	0x0000–70CA	1	
reserved	0x0000–70CB	1	
GPDMUX	0x0000–70CC	1	GPIO D Mux Control Register
GPDDIR	0x0000–70CD	1	GPIO D Direction Control Register
GPDQUAL	0x0000–70CE	1	GPIO D Input Qualification Control Register
reserved	0x0000–70CF	1	
GPEMUX	0x0000–70D0	1	GPIO E Mux Control Register
GPEDIR	0x0000–70D1	1	GPIO E Direction Control Register
GPEQUAL	0x0000–70D2	1	GPIO E Input Qualification Control Register
reserved	0x0000–70D3	1	
GPFMUX	0x0000–70D4	1	GPIO F Mux Control Register
GPFDIR	0x0000–70D5	1	GPIO F Direction Control Register
reserved	0x0000–70D6	1	
reserved	0x0000–70D7	1	
GPGMUX	0x0000–70D8	1	GPIO F Mux Control Register
GPGDIR	0x0000–70D9	1	GPIO F Direction Control Register
reserved	0x0000–70DA	1	
reserved	0x0000–70DB	1	
reserved	0x0000–70DC 0x0000–70DF	4	

† Registers that are not implemented will return undefined values and writes will be ignored.
‡ Not all inputs will support input signal qualification.
§ These registers are EALLOW protected. This prevents spurious writes from overwriting the contents and corrupting the system.

GPIO mux (continued)

If configured for ”Digital I/O” mode, additional registers are provided for setting individual I/O signals (via the GPxSET registers), for clearing individual I/O signals (via the GPxCLEAR registers), for toggling individual I/O signals (via the GPxTOGGLE registers), or for reading/writing to the individual I/O signals (via the GPxDAT registers). Table 61 lists the GPIO Data Registers.

Table 61. GPIO Data Registers†‡

NAME	ADDRESS	SIZE (x16)	REGISTER DESCRIPTION
GPADAT	0x0000–70E0	1	GPIO A Data Register
GPASET	0x0000–70E1	1	GPIO A Set Register
GPACLEAR	0x0000–70E2	1	GPIO A Clear Register
GPATOGGLE	0x0000–70E3	1	GPIO A Toggle Register
GPBDAT	0x0000–70E4	1	GPIO B Data Register
GPBSET	0x0000–70E5	1	GPIO B Set Register
GPBCLEAR	0x0000–70E6	1	GPIO B Clear Register
GPBTOGGLE	0x0000–70E7	1	GPIO B Toggle Register
reserved	0x0000–70E8	1	
reserved	0x0000–70E9	1	
reserved	0x0000–70EA	1	
reserved	0x0000–70EB	1	
GPDDAT	0x0000–70EC	1	GPIO D Data Register
GPDSET	0x0000–70ED	1	GPIO D Set Register
GPDCLEAR	0x0000–70EE	1	GPIO D Clear Register
GPDTOGGLE	0x0000–70EF	1	GPIO D Toggle Register
GPEDAT	0x0000–70F0	1	GPIO E Data Register
GPESET	0x0000–70F1	1	GPIO E Set Register
GPECLEAR	0x0000–70F2	1	GPIO E Clear Register
GPETOGGLE	0x0000–70F3	1	GPIO E Toggle Register
GPFDAT	0x0000–70F4	1	GPIO F Data Register
GPFSET	0x0000–70F5	1	GPIO F Set Register
GPFCLEAR	0x0000–70F6	1	GPIO F Clear Register
GPFTOGGLE	0x0000–70F7	1	GPIO F Toggle Register
GPGDAT	0x0000–70F8	1	GPIO G Data Register
GPGSET	0x0000–70F9	1	GPIO G Set Register
GPGCLEAR	0x0000–70FA	1	GPIO G Clear Register
GPGTOGGLE	0x0000–70FB	1	GPIO G Toggle Register
reserved	0x0000–70FC 0x0000–70FF	4	

† Reserved locations will return undefined values and writes will be ignored.
‡ These registers are NOT EALLOW protected. The above registers will typically be accessed regularly by the user.

GPIO mux (continued)

Figure 21 shows how the various register bits select the various modes of operation.

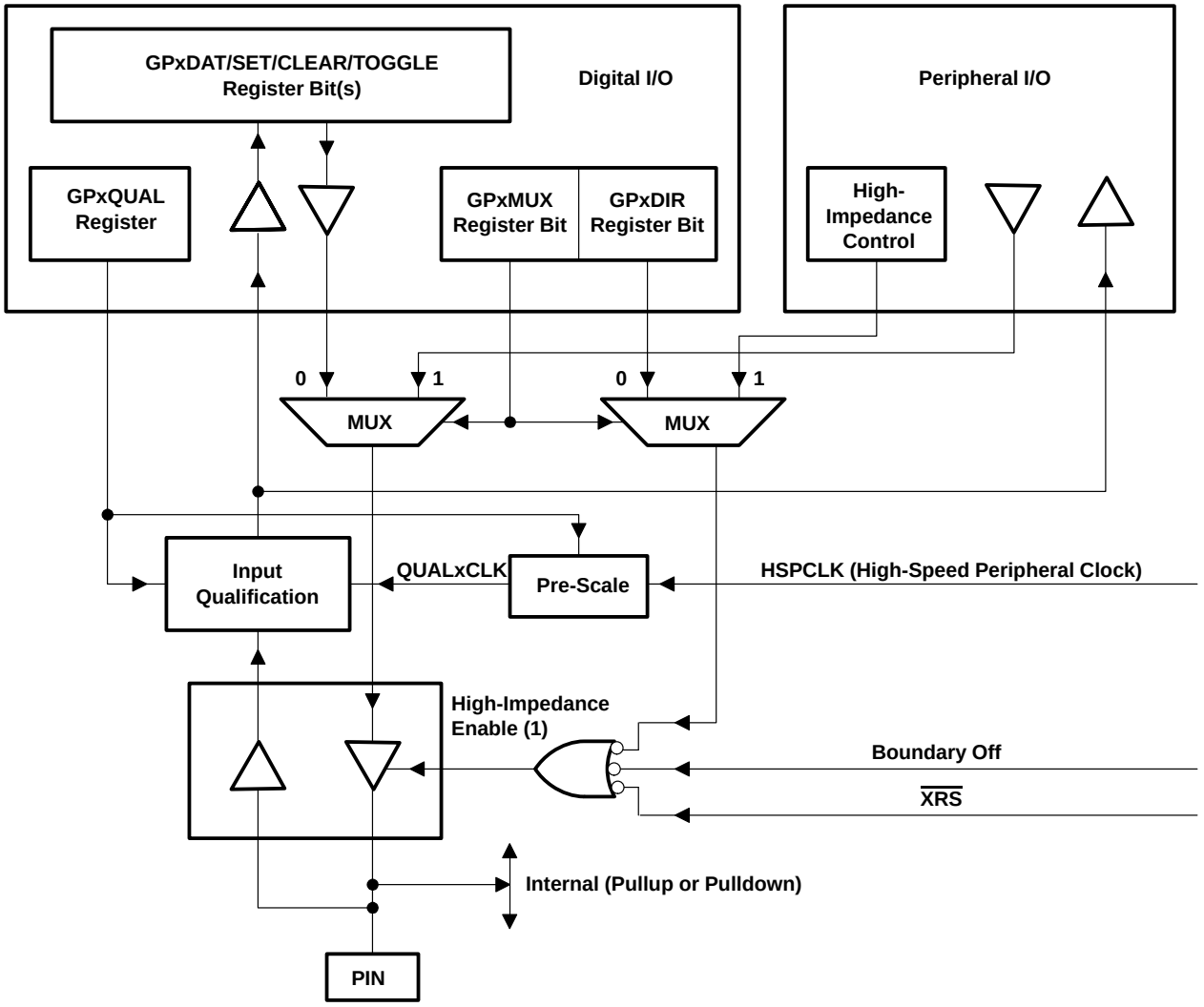


Figure 21. Modes of Operation

Notes:

1. Via the GPxDAT register, the state of any PIN can be read, regardless of the operating mode.
2. Some selected input signals are, qualified by the QUALxCLK, which is a prescaled version of the high-speed peripheral clock (HSPCLK). The GPxQUAL register specifies the qualification sampling period. The sampling window is 6 samples wide and the output is only changed when all samples are the same (all 0's or all 1's) as shown in Figure 22. This feature removes unwanted spikes from the input signal.

PRODUCT PREVIEW

GPIO mux (continued)

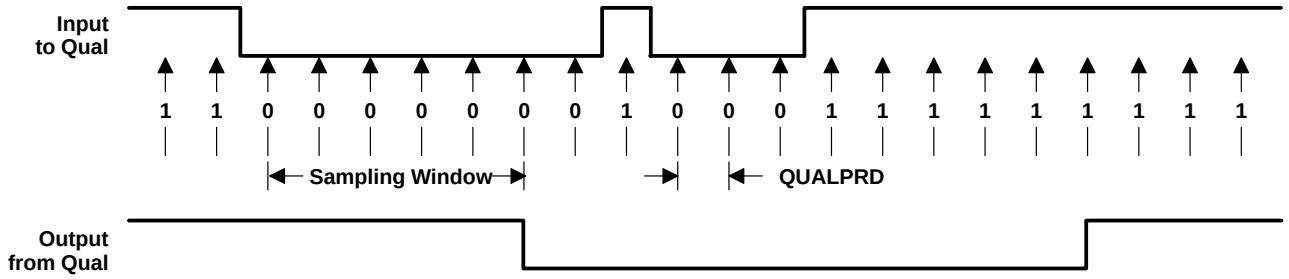


Figure 22. I/P Qualifier Clock Cycles

Table 62. GPAMUX, GPADIR Register Bit Definitions

GPAMUX BIT	PERIPHERAL NAME (BIT = 1)	GPIO NAME (BIT = 0)	GPADIR BIT	TYPE	RESET	INPUT QUAL
EV-A Peripheral						
0	PWM1 (O)	GPIOA0	0	R/W	0	yes
1	PWM2 (O)	GPIOA1	1	R/W	0	yes
2	PWM3 (O)	GPIOA2	2	R/W	0	yes
3	PWM4 (O)	GPIOA3	3	R/W	0	yes
4	PWM5 (O)	GPIOA4	4	R/W	0	yes
5	PWM6 (O)	GPIOA5	5	R/W	0	yes
6	T1PWM_T1CMP (0)	GPIOA6	6	R/W	0	yes
7	T2PWM_T2CMP (0)	GPIOA7	7	R/W	0	yes
8	CAP1_QEP1 (I)	GPIOA8	8	R/W	0	yes
9	CAP2_QEP2 (I)	GPIOA9	9	R/W	0	yes
10	CAP3_QEPI1 (I)	GPIOA10	10	R/W	0	yes
11	TDIRA (I)	GPIOA11	11	R/W	0	yes
12	TCLKINA (I)	GPIOA12	12	R/W	0	yes
13	$\overline{\text{C1TRIP}}$ (I)	GPIOA13	13	R/W	0	yes
14	$\overline{\text{C2TRIP}}$ (I)	GPIOA14	14	R/W	0	yes
15	$\overline{\text{C3TRIP}}$ (I)	GPIOA15	15	R/W	0	yes

GPIO mux (continued)

Table 63. GPAQUAL Register Bit Definitions

BIT	NAME	TYPE	RESET	DESCRIPTION
7:0	QUALPRD	R/W	0:0	Specifies the qualification sampling period: 0x00 no qualification (just SYNC to SYSCLKOUT) 0x01 QUALPRD = SYSCLKOUT/2 0x02 QUALPRD = SYSCLKOUT/4 . 0xFF QUALPRD = SYSCLKOUT/510
15:8	reserved	R=0	0:0	

Notes:

1. GPADIR bit = 0, configures corresponding GPIO pin as an input. GPADIR bit = 1, configures corresponding GPIO pin as an output.
2. The GPADAT, GPASET, GPACLEAR, GPATOGGLE registers have the same bit to I/O signal mapping as the GPAMUX and GPADIR registers.
3. The GPADAT register is a R/W register. Reading the register will reflect the current state of the input I/O signal (after qualification). Writing to the register will set the corresponding state of any I/O signal configured as an output.
4. The GPASET register is a write only register (reads back 0). Writing a 1 to the corresponding bit of an I/O signal will cause the I/O signal to go high. Writing a 0 will have no effect.
5. The GPACLEAR register is a write only register (reads back 0). Writing a 1 to the corresponding bit of an I/O signal will cause the I/O signal to go low. Writing a 0 will have no effect.
6. The GPATOGGLE register is a write only register (reads back 0). Writing a 1 to the corresponding bit of an I/O signal will cause the I/O signal to toggle. Writing a 0 will have no effect.

GPIO mux (continued)

Table 64. GPBMUX, GPBDIR Register Bit Definitions

GPBMUX BIT	PERIPHERAL NAME (BIT = 1)	GPIO NAME (BIT = 0)	GPBDIR BIT	TYPE	RESET	INPUT QUAL
EV-B Peripheral						
0	PWM7 (O)	GPIOB0	0	R/W	0	yes
1	PWM8 (O)	GPIOB1	1	R/W	0	yes
2	PWM9 (O)	GPIOB2	2	R/W	0	yes
3	PWM10 (O)	GPIOB3	3	R/W	0	yes
4	PWM11 (O)	GPIOB4	4	R/W	0	yes
5	PWM12 (O)	GPIOB5	5	R/W	0	yes
6	T3PWM_T3CMP (0)	GPIOB6	6	R/W	0	yes
7	T4PWM_T4CMP (0)	GPIOB7	7	R/W	0	yes
8	CAP4_QEP3 (I)	GPIOB8	8	R/W	0	yes
9	CAP5_QEP4 (I)	GPIOB9	9	R/W	0	yes
10	CAP6_QEPI2 (I)	GPIOB10	10	R/W	0	yes
11	TDIRB (I)	GPIOB11	11	R/W	0	yes
12	TCLKINB (I)	GPIOB12	12	R/W	0	yes
13	$\overline{\text{C4TRIP}}$ (I)	GPIOB13	13	R/W	0	yes
14	$\overline{\text{C5TRIP}}$ (I)	GPIOB14	14	R/W	0	yes
15	$\overline{\text{C6TRIP}}$ (I)	GPIOB15	15	R/W	0	yes

Table 65. GPBQUAL Register Bit Definitions

BIT	NAME	TYPE	RESET	DESCRIPTION
7:0	QUALPRD	R/W	0:0	Specifies the qualification sampling period: 0x00 no qualification (just SYNC to SYSCLKOUT) 0x01 QUALPRD = SYSCLKOUT/2 0x02 QUALPRD = SYSCLKOUT/4 . 0xFF QUALPRD = SYSCLKOUT/510
15:8	reserved	R=0	0:0	

PRODUCT PREVIEW

GPIO mux (continued)

Table 66. GPDMUX, GPDDIR Register Bit Definitions

GPDMUX BIT	PERIPHERAL NAME (BIT = 1)	GPIO NAME (BIT = 0)	GPDDIR BIT	TYPE	RESET	INPUT QUAL
EV-A Peripheral:						
0	$\overline{\text{T1CTRIP\_PDPINTA}}$ (I)	GPIOD0	0	R/W	0	yes
1	$\overline{\text{T2CTRIP}}$ (I)	GPIOD1	1	R/W	0	yes
2	reserved	GPIOD2	2	R/W	0	–
3	reserved	GPIOD3	3	R/W	0	–
4	reserved	GPIOD4	4	R/W	0	–
EV-B Peripheral:						
5	$\overline{\text{T3CTRIP\_PDPINTB}}$ (I)	GPIOD5	5	R/W	0	yes
6	$\overline{\text{T4CTRIP}}$ (I)	GPIOD6	6	R/W	0	yes
7	reserved	GPIOD7	7	R=0	0	–
8	reserved	GPIOD8	8	R=0	0	–
9	reserved	GPIOD9	9	R=0	0	–
10	reserved	GPIOD10	10	R/W	0	–
11	reserved	GPIOD11	11	R/W	0	–
12	reserved	GPIOD12	12	R=0	0	–
13	reserved	GPIOD13	13	R=0	0	–
14	reserved	GPIOD14	14	R=0	0	–
15	reserved	GPIOD15	15	R=0	0	–

Table 67. GPDQUAL Register Bit Definitions

BIT	NAME	TYPE	RESET	DESCRIPTION
7:0	QUALPRD	R/W	0:0	Specifies the qualification sampling period: 0x00 no qualification (just SYNC to SYSCLKOUT) 0x01 QUALPRD = SYSCLKOUT/2 0x02 QUALPRD = SYSCLKOUT/4 . 0xFF QUALPRD = SYSCLKOUT/510
15:8	reserved	R=0	0:0	

PRODUCT PREVIEW

GPIO mux (continued)

Table 68. GPEMUX, GPEDIR Register Bit Definitions

GPEMUX BIT	PERIPHERAL NAME (BIT = 1)	GPIO NAME (BIT = 0)	GPEDIR BIT	TYPE	RESET	INPUT QUAL
Interrupts:						
0	XINT1_$\overline{\text{XBIO}}$ (I)	GPIOE0	0	R/W	0	yes
1	XINT2_ADCSOC (I)	GPIOE1	1	R/W	0	yes
2	XNMI_XINT13 (I)	GPIOE2	2	R/W	0	yes
3	reserved	GPIOE3	3	R/W	0	–
4	reserved	GPIOE4	4	R/W	0	–
5	reserved	GPIOE5	5	R=0	0	–
6	reserved	GPIOE6	6	R=0	0	–
7	reserved	GPIOE7	7	R=0	0	–
8	reserved	GPIOE8	8	R=0	0	–
9	reserved	GPIOE9	9	R=0	0	–
10	reserved	GPIOE10	10	R=0	0	–
11	reserved	GPIOE11	11	R=0	0	–
12	reserved	GPIOE12	12	R=0	0	–
13	reserved	GPIOE13	13	R=0	0	–
14	reserved	GPIOE14	14	R=0	0	–
15	reserved	GPIOE15	15	R=0	0	–

Table 69. GPEQUAL Register Bit Definitions

BIT	NAME	TYPE	RESET	DESCRIPTION
7:0	QUALPRD	R/W	0:0	Specifies the qualification sampling period: 0x00 no qualification (just SYNC to SYSCLKOUT) 0x01 QUALPRD = SYSCLKOUT/2 0x02 QUALPRD = SYSCLKOUT/4 . 0xFF QUALPRD = SYSCLKOUT/510
15:8	reserved	R=0	0:0	

GPIO mux (continued)

Table 70. GPFMUX, GPFDIR Register Bit Defintions

GPFMUX BIT	PERIPHERAL NAME (BIT = 1)	GPIO NAME (BIT = 0)	GPFDIR BIT	TYPE	RESET	INPUT QUAL
SPI Peripheral:						
0	SPISIMO (O)	GPIOF0	0	R/W	0	no
1	SPISOMI (I)	GPIOF1	1	R/W	0	no
2	SPICLK (I/O)	GPIOF2	2	R/W	0	no
3	SPISTE (I/O)	GPIOF3	3	R/W	0	no
SCIA Peripheral:						
4	SCITXDA (O)	GPIOF4	4	R/W	0	no
5	SCIRXDA (I)	GPIOF5	5	R/W	0	no
CAN Peripheral:						
6	CANTX (O)	GPIOF6	6	R/W	0	no
7	CANRX (I)	GPIOF7	7	R/W	0	no
McBSP Peripheral:						
8	MCLKX (I/O)	GPIOF8	8	R/W	0	no
9	MCLKR (I/O)	GPIOF9	9	R/W	0	no
10	MFSX (I/O)	GPIOF10	10	R/W	0	no
11	MFSR (I/O)	GPIOF11	11	R/W	0	no
12	MDX (O)	GPIOF12	12	R/W	0	no
13	MDR (I)	GPIOF13	13	R/W	0	no
XINT I/O Space Strobe & XF CPU Output Signal:						
14	XF (0)	GPIOF14	14	R/W	0	no

GPIO mux (continued)

Table 71. GPGMUX, GPGDIR Register Bit Definitions

GPGMUX BIT	PERIPHERAL NAME (BIT = 1)	GPIO NAME (BIT = 0)	GPGDIR BIT	TYPE	RESET	INPUT QUAL
0	reserved	GPIOG0	0	R/W	0	–
1	reserved	GPIOG1	1	R/W	0	–
2	reserved	GPIOG2	2	R/W	0	–
3	reserved	GPIOG3	3	R/W	0	–
SCI-B Peripheral:						
4	SCITXDB (O)	GPIOG4	4	R/W	0	no
5	SCIRXDB (I)	GPIOG5	5	R/W	0	no
6	reserved	GPIOG6	6	R/W	0	–
7	reserved	GPIOG7	7	R/W	0	–
8	reserved	GPIOG8	8	R/W	0	–
9	reserved	GPIOG9	9	R/W	0	–
10	reserved	GPIOG10	10	R/W	0	–
11	reserved	GPIOG11	11	R/W	0	–
12	reserved	GPIOG12	12	R/W	0	–
13	reserved	GPIOG13	13	R/W	0	–
14	reserved	GPIOG14	14	R/W	0	–
15	reserved	GPIOG15	15	R/W	0	–

development support

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

The following products support development of F2810- and F2812-based applications:

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

Hardware Development Tools:
Emulator XDS510™/XDS510PP™ (supports x24x/28x multiprocessor system debug)
SPI515 (third-party tool)
XDS510PP (third-party tool)

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

Development tools for the 28x are as follows:

- Code Composer Studio™ Integrated Development Environment (IDE) Version 2.0
  - Code Composer Studio Version 2.0 Debugger
  - Code Generation Tools
    - Assembler/Linker
    - C/C++ Compiler
    - Cycle Accurate Simulator
- JTAG-Based Emulator
- Sample Applications Code
- Universal 5-V DC Power Supply
- Documentation and Cables

PRODUCT PREVIEW

XDS510, XDS510PP, TMS320, and Code Composer Studio are trademarks of Texas Instruments.

device and development support tool nomenclature

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

Support tool development evolutionary flow:

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

TMDS Fully qualified development support product

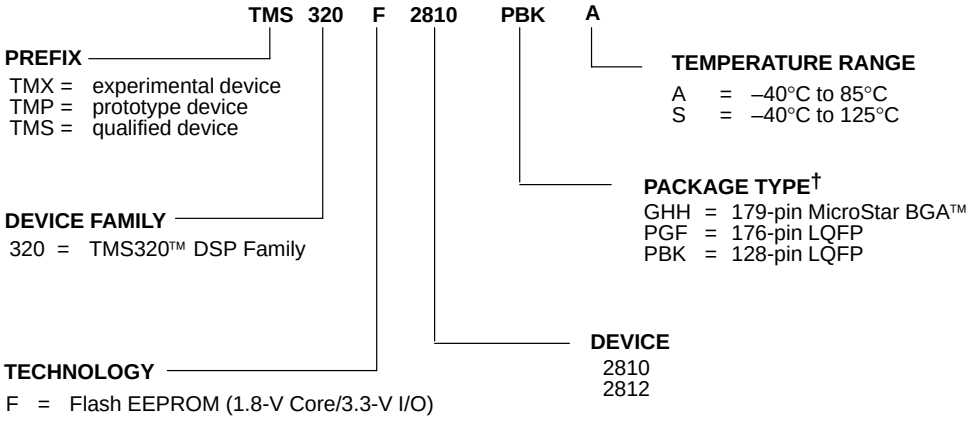
TMX and TMP devices and TMDX development support tools are shipped against the following disclaimer:

"Developmental product is intended for internal evaluation purposes."

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

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 23 provides a legend for reading the complete device name for any TMS320x28x family member. Refer to the timing section for specific options that are available on F2810 and F2812 devices.

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.



† BGA = Ball Grid Array
LQFP = Low-Profile Quad Flatpack

Figure 23. TMS320x28x Device Nomenclature

documentation support

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

- Application Reports
  - *3.3V DSP for Digital Motor Control* (literature number SPRA550)

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 TMS320F2810/TMS320F2812 data sheet (literature number SPRS174), use the *comments@books.sc.ti.com* email address, which is a repository for feedback. For questions and support, contact the Product Information Center listed at the **http://www.ti.com/sc/docs/pic/home.htm** site.

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

Supply voltage range, V_{DD}, $PLLV_{CCA}$, V_{DDO}, and V_{CCA} (see Note 1) . . . . . – 0.3 V to 4.6 V
Supply voltage range, CV_{DD} . . . . . – 0.5 V to 2 V
V_{CCP} range . . . . . – 0.3 V to 3.6 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
Output voltage range,V_O . . . . . – 0.3 V to 4.6 V
Input clamp current, I_{IK} ($V_{IN} < 0$ or $V_{IN} > V_{CC}$) . . . . . ± 20 mA
Output clamp current, I_{OK} ($V_O < 0$ or $V_O > V_{CC}$) . . . . . ± 20 mA
Operating free-air temperature ranges, T_A: A version . . . . . – 40°C to 85°C
S version . . . . . – 40°C to 125°C
Junction temperature range, T_J . . . . . – 40°C to 150°C
Storage temperature range, T_{stg} . . . . . – 65°C to 150°C

† Clamp current stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to V_{SS}.

recommended operating conditions‡§

				MIN	NOM	MAX	UNIT
V_{DD}/V_{DDO}	Supply voltage		$V_{DDO} = V_{DD} \pm 0.3$ V	3	3.3	3.6	V
CV_{DD}	Device supply voltage, CPU/core			1.65	1.8	1.95	V
V_{SS}	Supply ground			0	0	0	V
$PLLV_{CCA}$	PLL supply voltage			3	3.3	3.6	V
V_{CCA}¶	ADC supply voltage			3	3.3	3.6	V
V_{CCP}	Flash programming supply voltage			3	3.3	3.6	V
f_{CLKOUT}	Device clock frequency (system clock)			2		150	MHz
V_{IH}	High-level input voltage		All inputs	2			V
V_{IL}	Low-level input voltage		All inputs			0.8	V
I_{OH}	High-level output source current, V_{OH} = 2.4 V					– 2	mA
I_{OL}	Low-level output sink current, $V_{OL} = V_{OL}$ MAX					2	mA
T_A	Free-air temperature	A version		– 40		85	°C
		S version		– 40		125	°C
T_J	Junction temperature			– 40	25	150	°C
N_f	Flash endurance for the array (Write/erase cycles)		– 40°C to 85°C		TBD		cycles
N_{OTP}	OTP endurance for the array (Write cycles)		– 40°C to 85°C		1		cycles

‡ Refer to the mechanical data package page for thermal resistance values, Θ_{JA} (junction-to-ambient) and Θ_{JC} (junction-to-case).
§ The drive strength of the EVA PWM pins and the EVB PWM pins are *not* identical.
¶ V_{CCA} should not exceed V_{DD} by 0.3 V.

PRODUCT PREVIEW

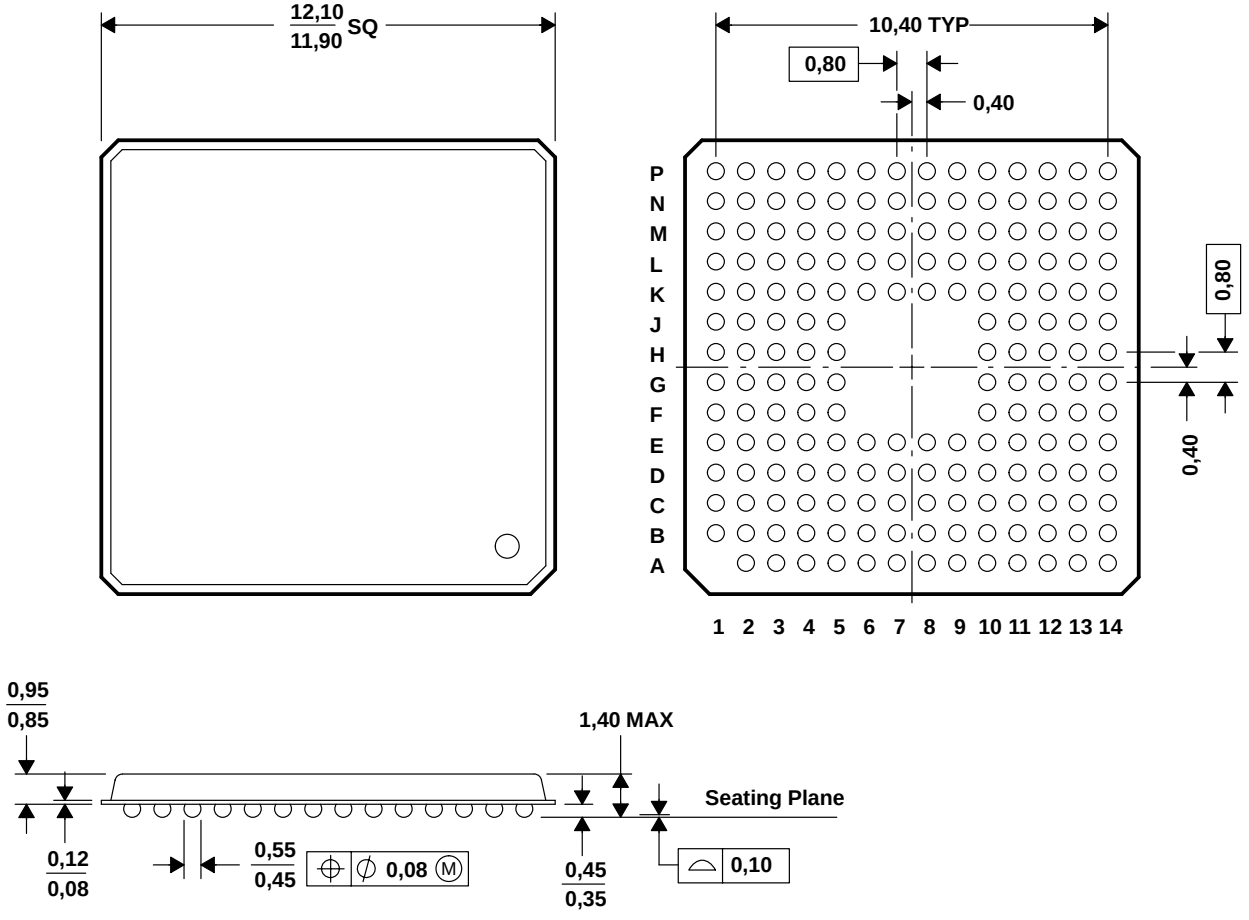
electrical characteristics over recommended operating free-air temperature ranges (unless otherwise noted)

PARAMETER			TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage		V_{DD} = 3.0 V, I_{OH} = I_{OH}MAX	2.4			V
			All outputs at 50 µA	V_{DDO} – 0.2			
V_{OL}	Low-level output voltage		I_{OL} = I_{OL}MAX			0.4	V
I_{IL}	Input current (low level)	With pullup	V_{DD} = 3.3 V, V_{IN} = 0 V		–16		µA
		With pulldown				±2	
I_{IH}	Input current (high level)	With pullup	V_{DD} = 3.3 V, V_{IN} = V_{DD}			±2	µA
		With pulldown			16		
I_{OZ}	Output current, high-impedance state (off-state)		V_O = V_{DD} or 0 V			±2	µA
C_i	Input capacitance				2		pF
C_o	Output capacitance				3		pF

PRODUCT PREVIEW

MECHANICAL DATA

GHH (S-PBGA-N179) **PLASTIC BALL GRID ARRAY**



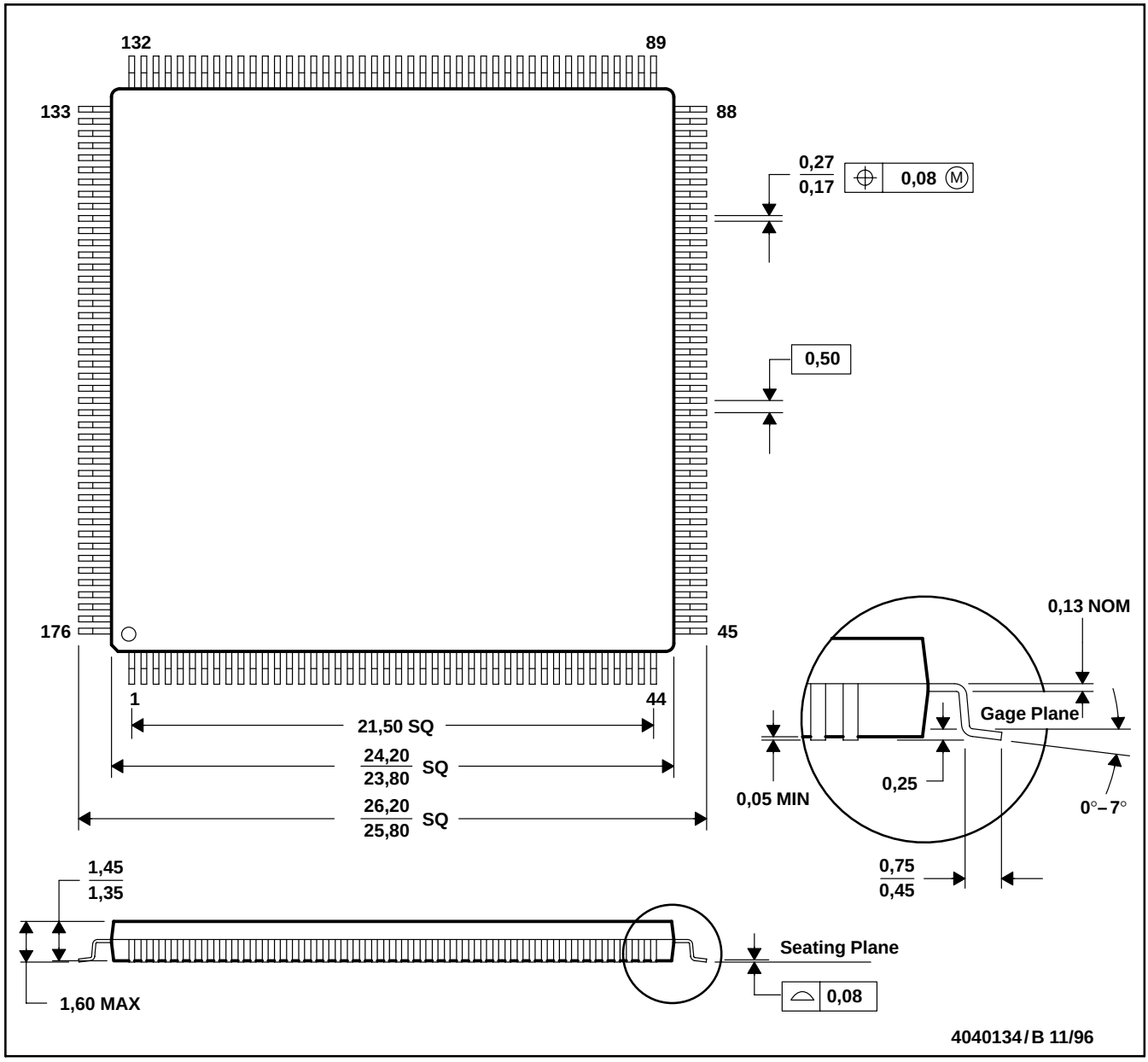
NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. MicroStar BGA™ configuration

PRODUCT PREVIEW

MicroStar BGA is a trademark of Texas Instruments.

MECHANICAL DATA

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

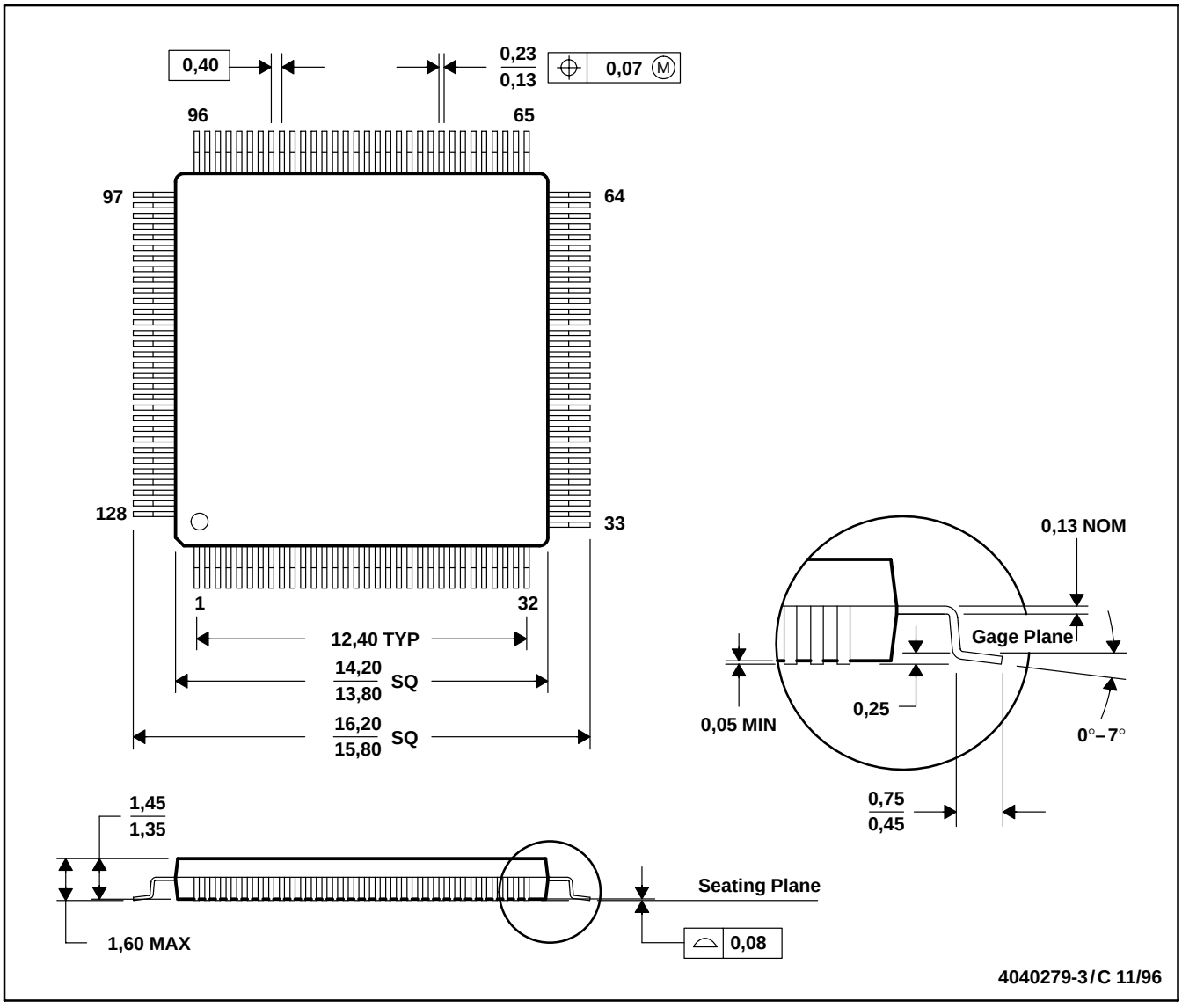


NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

PRODUCT PREVIEW

MECHANICAL DATA

PBK (S-PQFP-G128) **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MS-026

PRODUCT PREVIEW

IMPORTANT NOTICE

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

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

Customers are responsible for their applications using TI components.

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

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

Reproduction of 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. Representation or reproduction of this information with alteration voids all warranties provided for an associated TI product or service, is an unfair and deceptive business practice, and TI is not responsible nor liable for any such use.

Resale of TI's 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, is an unfair and deceptive business practice, and TI is not responsible nor liable for any such use.

Also see: Standard Terms and Conditions of Sale for Semiconductor Products. www.ti.com/sc/docs/stdterms.htm

Mailing Address:

Texas Instruments
Post Office Box 655303
Dallas, Texas 75265