Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

Mixed-Signal Control Processor with ARM Cortex-M4 and 16-Bit ADCs

ADSP-CM402F/CM403F/CM407F/CM408F/CM409F

SYSTEM FEATURES

- Up to 240 MHz ARM Cortex-M4 with floating-point unit
- 24-channel analog front end (AFE) with 16-bit ADCs
- 128K Byte to 384K Byte zero-wait-state L1 SRAM with 16K Byte L1 cache
- Up to 2M Byte flash memory
- Single 3.3 V power supply
- Package Options:
  - 176-lead (24 mm × 24 mm) LQFP package
  - 120-lead (14 mm × 14 mm) LQFP package
  - 212-ball (19 mm × 19 mm) BGA package
- Static memory controller (SMC) with asynchronous memory interface that supports 8-bit and 16-bit memories
- Enhanced PWM units
- Four 3rd/4th order SINC filter pairs for glueless connection of sigma-delta modulators
- Hardware-based harmonic analysis engine
- 10/100 Ethernet MAC with IEEE 1588v2 support
- Full Speed USB on-the-go (OTG)
- Two CAN (controller area network) 2.0B interfaces
- Three UART ports
- Two serial peripheral interface (SPI-compatible) ports
- Three/four synchronous serial ports
- Eight 32-bit GP timers, three capture timing units
- Four encoder interfaces, 2 with frequency division
- One TWI unit, fully compatible with I^2C bus standard
- Lightweight security

ANALOG FRONT END

- Two 16-bit SAR ADCs with up to 24 multiplexed inputs, supporting dual simultaneous conversion in 380 ns (16-bit, no missing codes)
- ADC controller (ADCC) and DAC controller (DACC)
- Two 12-bit DACs
- Two 2.5 V precision voltage reference outputs
- (For details, see ADC/DAC Specifications on Page 68)

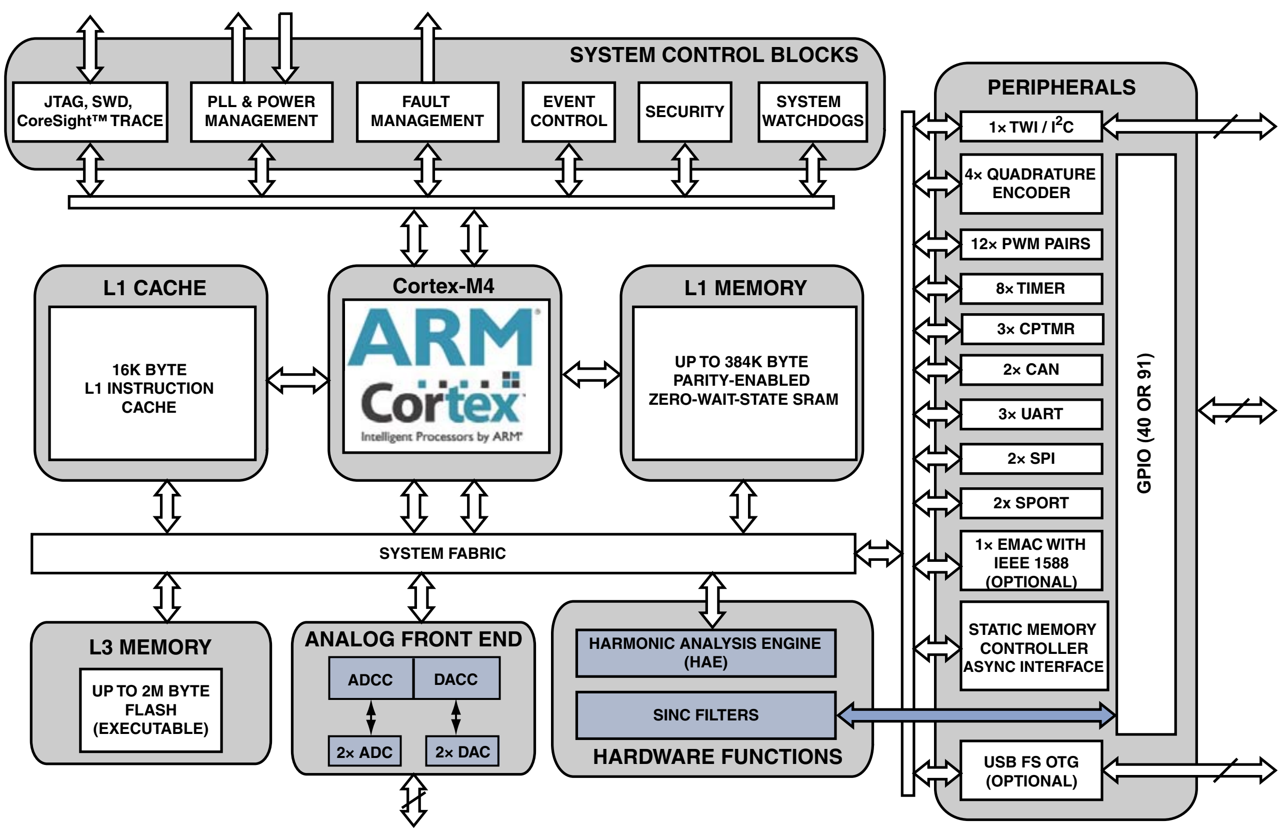

Figure 1. Block Diagram

Rev. A Document Feedback

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106 U.S.A.
Tel: 781.329.4700
Technical Support www.analog.com

TABLE OF CONTENTS

REVISION HISTORY

11/15—Rev. 0 to Rev. A

Change to equation in Serial Ports 83

Change to equation in Serial Peripheral Interface (SPI) Port—Master Timing 89

Changes to Ordering Guide 124

GENERAL DESCRIPTION

The ADSP-CM40xF family of mixed-signal control processors is based on the ARM® Cortex-M4™ processor core with floating-point unit operating at frequencies up to 240 MHz and integrating up to 384 kB of SRAM memory, 2 MB of flash memory, accelerators and peripherals optimized for motor control and photo-voltaic (PV) inverter control and an analog module consisting of two 16-bit SAR ADCs and two 12-bit DACs. The ADSP-CM40xF family operates from a single voltage supply (VDD_EXT/VDD_ANA), generating its own internal voltage supplies using internal voltage regulators and an external pass transistor.

This family of mixed-signal control processors offers low static power consumption and is produced with a low power and low voltage design methodology, delivering world class processor and ADC performance with lower power consumption.

By integrating a rich set of industry-leading system peripherals and memory (shown in Table 1), the ADSP-CM40xF mixed-signal control processors are the platform of choice for next-generation applications that require RISC programmability, advanced communications and leading-edge signal processing in one integrated package. These applications span a wide array of markets including power/motor control, embedded industrial, instrumentation, medical and consumer.

Each ADSP-CM40xF family member contains the following modules.

- 8 GP timers with PWM output
- 3-phase PWM units with up to 4 output pairs per unit
- 2 CAN modules
- 1 two-wire interface (TWI) module
- 3 UARTs
- 1 ADC controller (ADCC) to control on-chip ADCs
- 1 DAC controller (DACC) to control on-chip DACs
- 4 Sinus Cardinalis (SINC) filter pairs
- 1 harmonic analysis engine (HAE)
- 2 SPI (1 connected to internal SPI flash memory)
- 3 half-SPORTs
- 1 watchdog timer unit
- 3 capture timer units
- 1 cyclic redundancy check (CRC)

Table 1 provides the additional product features shown by model.

Table 1. ADSP-CM4 0xF Family Product Features

<table>
<tr><th>Generic</th><th colspan="2">ADSP-CM402F</th><th colspan="3">ADSP-CM403F</th><th colspan="3">ADSP-CM407F</th><th colspan="2">ADSP-CM408F</th><th>ADSP-CM409F</th></tr>
<tr><td>Package</td><td colspan="5">120-Lead LQFP</td><td colspan="5">176-Lead LQFP</td><td>212-Ball BGA</td></tr>
<tr><td>GPIOs</td><td colspan="5">40</td><td colspan="6">91</td></tr>
<tr><td>SMC</td><td colspan="5">16-Bit Asynchronous/5 Address</td><td colspan="6">16-Bit Asynchronous/24 Address</td></tr>
<tr><td>ADC ENOB (No Averaging)</td><td colspan="2">11+</td><td colspan="3">13+</td><td colspan="3">11+</td><td colspan="3">13+</td></tr>
<tr><td>ADC Inputs</td><td colspan="5">24</td><td colspan="5">16</td><td>24</td></tr>
<tr><td>DAC Outputs</td><td colspan="5">2</td><td colspan="5">N/A</td><td>2</td></tr>
<tr><td>SPORTs</td><td colspan="5">3 Half-SPORTs</td><td colspan="6">4 Half-SPORTs</td></tr>
<tr><td>Ethernet</td><td colspan="5">N/A</td><td>1</td><td>N/A</td><td>N/A</td><td>1</td><td>N/A</td><td>1</td></tr>
<tr><td>USB</td><td colspan="5">N/A</td><td>1</td><td>1</td><td>N/A</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>External SPI</td><td colspan="5">1</td><td colspan="6">2</td></tr>
<tr><td>HAE</td><td colspan="11">1</td></tr>
<tr><td>CAN</td><td colspan="11">2</td></tr>
<tr><td>UART</td><td colspan="11">3</td></tr>
<tr><td>Feature Set Code</td><td>E</td><td>F</td><td>C</td><td>E</td><td>F</td><td>A</td><td>B</td><td>D</td><td>A</td><td>B</td><td>A</td></tr>
<tr><td>L1 SRAM (kB)</td><td>128</td><td>128</td><td>384</td><td>128</td><td>128</td><td>384</td><td>384</td><td>128</td><td>384</td><td>384</td><td>384</td></tr>
<tr><td>Flash (kB)</td><td>512</td><td>256</td><td>2048</td><td>512</td><td>256</td><td>2048</td><td>2048</td><td>1024</td><td>2048</td><td>2048</td><td>2048</td></tr>
<tr><td>Core Clock (MHz)</td><td>150</td><td>100</td><td>240</td><td>150</td><td>100</td><td>240</td><td>240</td><td>150</td><td>240</td><td>240</td><td>240</td></tr>
<tr><td>Model</td><td>ADSP-CM402CSWZ-EF</td><td>ADSP-CM402CSWZ-FF</td><td>ADSP-CM403CSWZ-CF</td><td>ADSP-CM403CSWZ-EF</td><td>ADSP-CM403CSWZ-FF</td><td>ADSP-CM407CSWZ-AF</td><td>ADSP-CM407CSWZ-BF</td><td>ADSP-CM407CSWZ-DF</td><td>ADSP-CM408CSWZ-AF</td><td>ADSP-CM408CSWZ-BF</td><td>ADSP-CM409CBCZ-AF</td></tr>
</table>

ANALOG FRONT END

The mixed-signal controllers contain two ADCs and two DACs. Control of these data converters is simplified by a powerful on-chip analog-to-digital conversion controller (ADCC) and a digital-to-analog conversion controller (DACC). The ADCC and DACC are integrated seamlessly into the software programming model, and they efficiently manage the configuration and real-time operation of the ADCs and DACs.

For technical details, see ADC/DAC Specifications on Page 68.

The ADCC provides the mechanism to precisely control execution of timing and analog sampling events on the ADCs. The ADCC supports two-channel (one each—ADC0, ADC1) simultaneous sampling of ADC inputs and can deliver 16 channels of ADC data to memory in 3 µs. Conversion data from the ADCs may be either routed via DMA to memory, or to a destination register via the processor. The ADCC can be configured so that the two ADCs sample and convert both analog inputs simultaneously or at different times and may be operated in asynchronous or synchronous modes. The best performance can be achieved in synchronous mode.

Likewise, the DACC interfaces to two DACs and has purpose of managing those DACs. Conversion data to the DACs may be either routed from memory through DMA, or from a source register via the processor.

Functional operation and programming for the ADCC and DACC are described in detail in the *ADSP-CM40x Mixed-Signal Control Processor with ARM Cortex-M4 Hardware Reference*.

ADC and DAC features and performance specifications differ by processor model. Simplified block diagrams of the ADCC/DACC and the ADC/DAC are shown in Figure 2 and Figure 3.

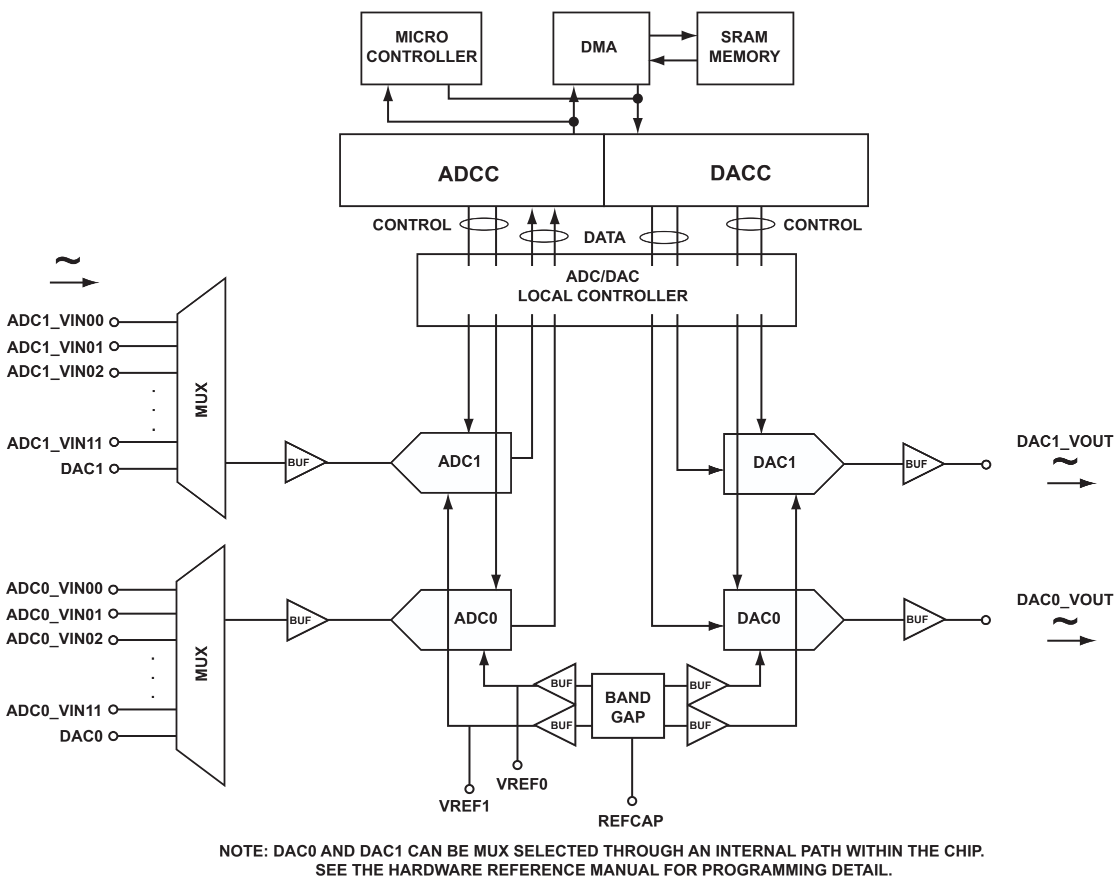

Figure 2. ADSP-CM402F/ADSP-CM403F/ADSP-CM409F Analog Front End Block Diagram

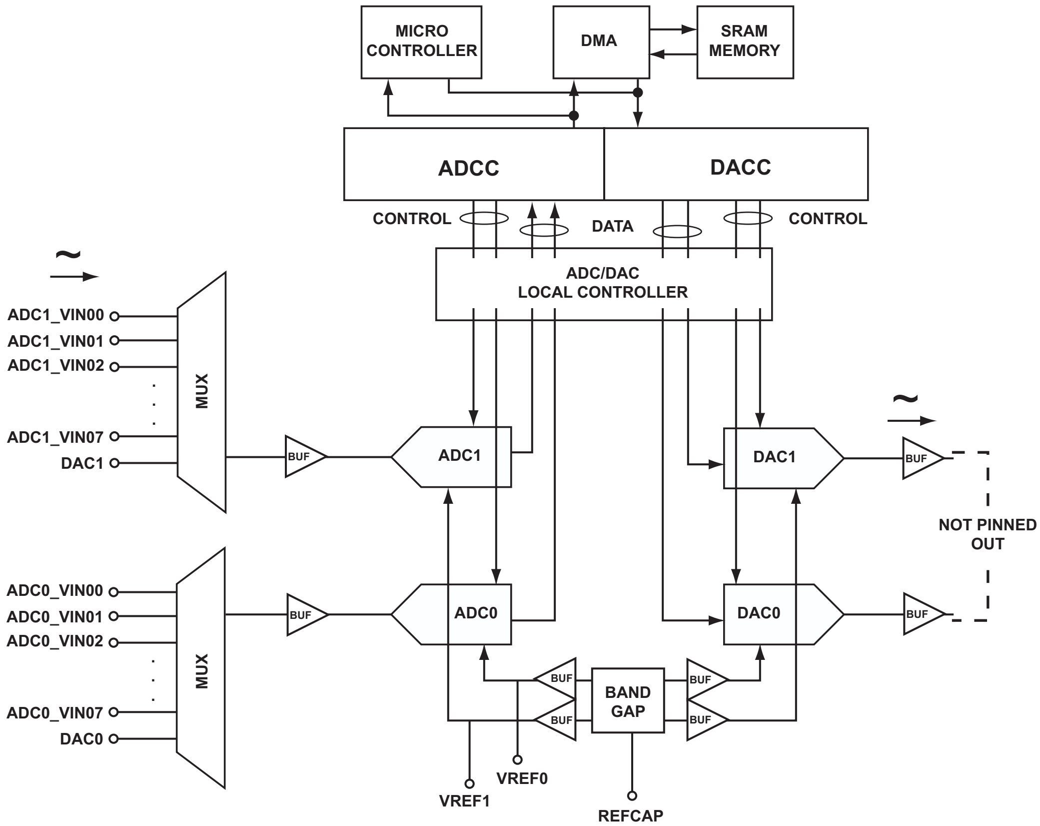

Figure 3. ADSP-CM407F/ADSP-CM408F Analog Subsystem Block Diagram

Considerations for Best Converter Performance

As with any high performance analog/digital circuit, to achieve best performance, good circuit design and board layout practices should be followed. The power supply and its noise bypass (decoupling), ground return paths and pin connections, and analog/digital routing channel paths and signal shielding, are all of first-order consideration. For application hints on design best practice, see Figure 4 and the *ADSP-CM40x Mixed-Signal Control Processor with ARM Cortex-M4 Hardware Reference*. For more information about the VREG circuit, see Figure 9.

ADC Module

The ADC module contains two 16-bit, high speed, low power successive approximation register (SAR) ADCs, allowing for dual simultaneous sampling with each ADC preceded by a 12-channel multiplexer. See ADC Specifications on Page 68 for detailed performance specifications. Input multiplexers enable conversion of up to a combined 26 analog input sources to the ADCs (12 analog inputs plus 1 DAC loopback input per ADC).

The voltage input range requirement for those analog inputs is from 0 V to 2.5 V. All analog inputs are of single-ended design. As with all single-ended inputs, signals from high impedance sources are the most difficult to measure, and depending on the electrical environment, may require an external buffer circuit for signal conditioning (see Figure 5). An on-chip pre-buffer between the multiplexer and ADC reduces the need for additional signal conditioning external to the processor. Additionally, each ADC has an on-chip 2.5 V reference that can be overdriven when an external voltage reference is preferred.

DAC Module

The DAC is a 12-bit, low power, string DAC design. The output of the DAC is buffered, and can drive an R/C load to either ground or VDD_ANA. See DAC Specifications on Page 70 for detailed performance specifications. It should be noted that on some models of the processor, the DAC outputs are not pinned out. However, these outputs are always available as one of the multiplexed inputs to the ADCs. This feature may be useful for functional self-check of the converters.

Note: On the ADSP-CM402F/CM403F/CM409F processors, the DAC output is available to the ADC as channel 12; whereas on the ADSP-CM407F/CM408F processors, the DAC output is available to the ADC as Channel 8.

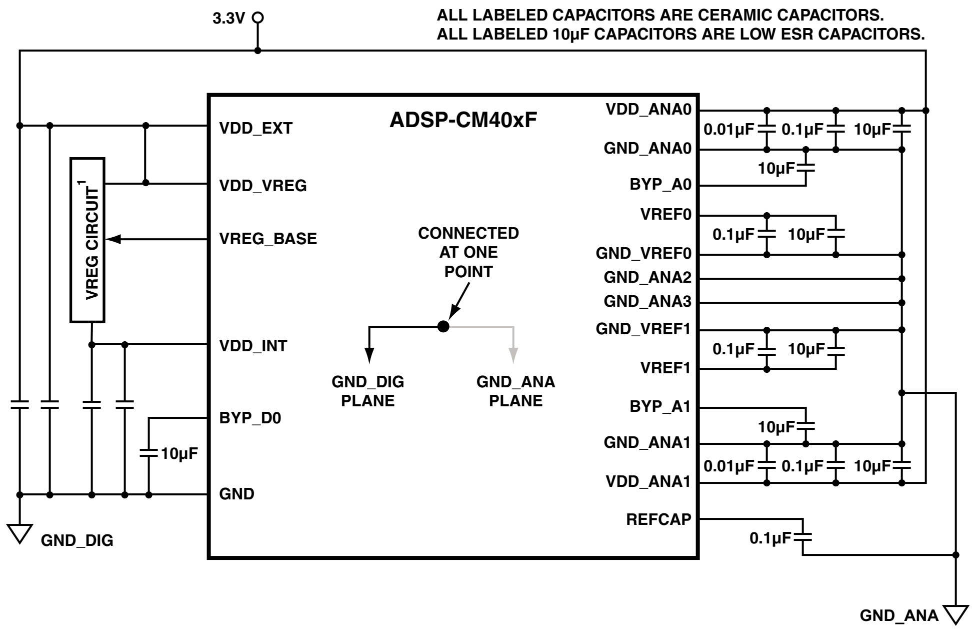

Figure 4. Typical Power Supply Configuration

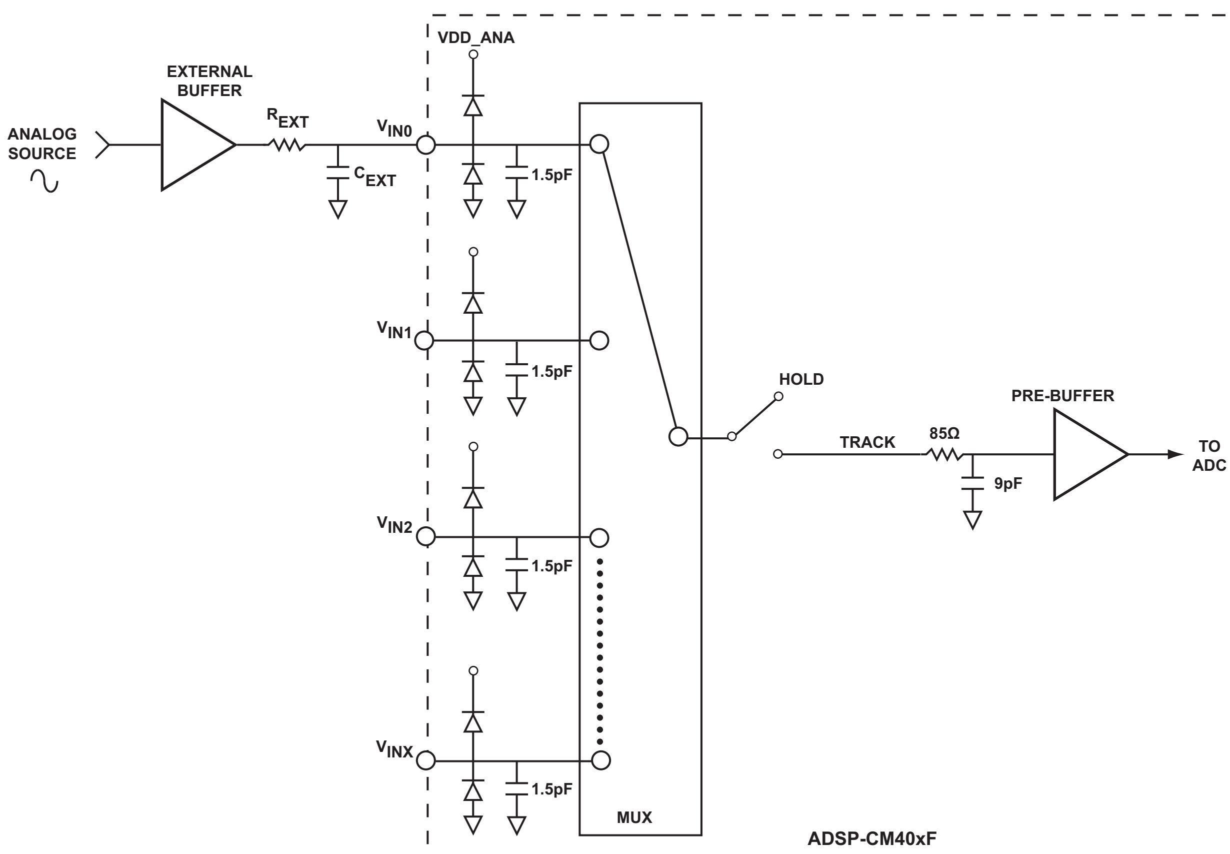

Figure 5. Equivalent Single-Ended Input (Simplified)

ARM CORTEX-M4 CORE

The ARM Cortex-M4, core shown in Figure 6, is a 32-bit reduced instruction set computer (RISC). It uses 32-bit buses for instruction and data. The length of the data can be 8 bits, 16 bits, or 32 bits. The length of the instruction word is 16 or 32 bits. The controller has the following features.

Cortex-M4 Architecture

- Thumb-2 ISA technology
- DSP and SIMD extensions
- Single cycle MAC (Up to 32 × 32 + 64 → 64)
- Hardware divide instructions
- Single-precision FPU
- NVIC interrupt controller (129 interrupts and 16 priorities)
- Memory protection unit (MPU)
- Full CoreSight™ debug, trace, breakpoints, watchpoints, and cross-triggers

Microarchitecture

- 3-stage pipeline with branch speculation
- Low-latency interrupt processing with tail chaining

Configurable For Ultra Low Power

- Deep sleep mode, dynamic power management
- Programmable clock generator unit

EmbeddedICE

EmbeddedICE™ provides integrated on-chip support for the core. The EmbeddedICE module contains the breakpoint and watchpoint registers that allow code to be halted for debugging purposes. These registers are controlled through the JTAG test port.

When a breakpoint or watchpoint is encountered, the processor halts and enters debug state. Once in a debug state, the processor registers can be inspected as well as the Flash/EE, SRAM, and memory-mapped registers.

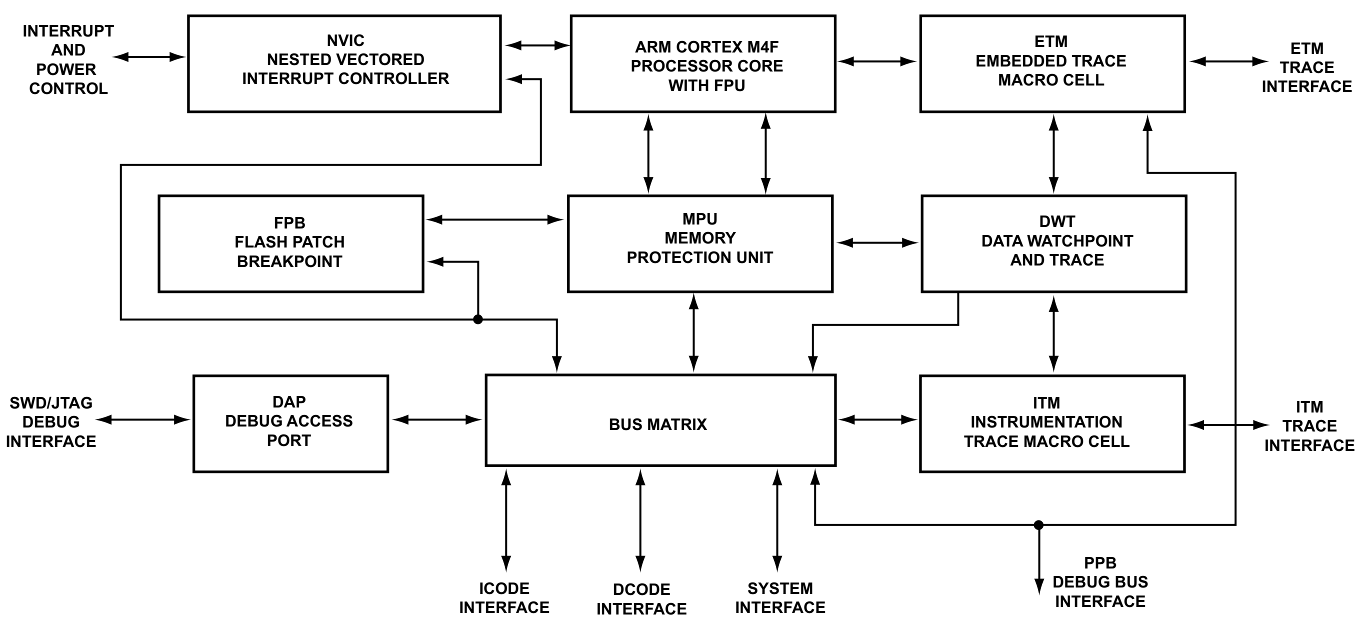

Figure 6. Cortex-M4 Block Diagram

PROCESSOR INFRASTRUCTURE

The following sections provide information on the primary infrastructure components of the ADSP-CM40xF processors.

DMA Controllers (DDEs)

The processor contains 17 independent and concurrently operating peripheral DMA channels plus two MDMA streams. DDE Channel 0 to Channel 16 are for peripherals and Channel 17 to Channel 20 are for MDMA.

System Event Controller (SEC)

The SEC manages the enabling and routing of system fault sources through its integrated fault management unit.

Trigger Routing Unit (TRU)

The TRU provides system-level sequence control without core intervention. The TRU maps trigger masters (generators of triggers) to trigger slaves (receivers of triggers). Slave endpoints can be configured to respond to triggers in various ways. Common applications enabled by the TRU include:

- Initiating the ADC sampling periodically in each PWM period or based on external events
- Automatically triggering the start of a DMA sequence after a sequence from another DMA channel completes
- Software triggering
- Synchronization of concurrent activities

Pin Interrupts (PINT)

Every port pin on the processor can request interrupts in either an edge-sensitive or a level-sensitive manner with programmable polarity. Interrupt functionality is decoupled from GPIO operation. Six system-level interrupt channels (PINT0 to PINT5) are reserved for this purpose. Each of these interrupt channels can manage up to 32 interrupt pins. The assignment from pin to interrupt is not performed on a pin-by-pin basis. Rather, groups of eight pins (half ports) can be flexibly assigned to interrupt channels.

Every pin interrupt channel features a special set of 32-bit memory-mapped registers that enable half-port assignment and interrupt management. This includes masking, identification, and clearing of requests. These registers also enable access to the respective pin states and use of the interrupt latches, regardless of whether the interrupt is masked or not. Most control registers feature multiple MMR address entries to write-one-to-set or write-one-to-clear them individually.

General-Purpose I/O (GPIO)

Each general-purpose port pin can be individually controlled by manipulation of the port control, status, and interrupt registers:

- GPIO direction control register—Specifies the direction of each individual GPIO pin as input or output.
- GPIO control and status registers —A write one to modify mechanism allows any combination of individual GPIO pins to be modified in a single instruction, without affecting the level of any other GPIO pins.
- GPIO interrupt mask registers—Allow each individual GPIO pin to function as an interrupt to the processor. GPIO pins defined as inputs can be configured to generate hardware interrupts, while output pins can be triggered by software interrupts.
- GPIO interrupt sensitivity registers—Specify whether individual pins are level- or edge-sensitive and specify—if edge-sensitive—whether just the rising edge or both the rising and falling edges of the signal are significant.

Pin Multiplexing

The processor supports a flexible multiplexing scheme that multiplexes the GPIO pins with various peripherals. A maximum of five peripherals plus GPIO functionality is shared by each GPIO pin. All GPIO pins have a bypass path feature—that is, when the output enable and the input enable of a GPIO pin are both active, the data signal before the pad driver is looped back to the receive path for the same GPIO pin.

For more information, see:

- ADSP-CM402F/ADSP-CM403F GPIO Multiplexing for 120-Lead LQFP on Page 27.
- ADSP-CM407F/ADSP-CM408F GPIO Multiplexing for 176-Lead LQFP on Page 37.
- ADSP-CM409F GPIO Multiplexing for 212-Ball BGA on Page 48.

MEMORY ARCHITECTURE

The internal and external memory of the ADSP-CM40xF processor is shown in Figure 7 and described in the following sections.

ARM Cortex-M4 Memory Subsystem

The memory map of the ADSP-CM40xF family is based on the Cortex-M4 model from ARM. By retaining the standardized memory mapping, it becomes easier to port applications across M4 platforms. Only the physical implementation of memories inside the model differs from other vendors.

ADSP-CM40xF application development is typically based on memory blocks across CODE/SRAM and external memory regions. Sufficient internal memory is available via internal SRAM and internal flash. Additional external memory devices may be interfaced via the SMC asynchronous memory port, as well as through the SPI0 serial memory interface.

Code Region

Accesses in this region (0x0000_0000 to 0x1FFF_FFFF) are performed by the core on its ICODE and DCODE interfaces, and they target the memory and cache resources within the Cortex-M4F platform integration component.

- **Boot ROM.** A 32K byte boot ROM executed at system reset. This space supports read-only access by the M4F core only. Note that ROM memory contents cannot be modified by the user.

- **Internal SRAM Code Region.** This memory space contains the application instructions and literal (constant) data which must be executed real time. It supports read/write access by the M4F core and read/write DMA access by system devices. Internal SRAM can be partitioned between CODE and DATA (SRAM region in M4 space) in 64K byte blocks. Access to this region occurs at core clock speed, with no wait states.
- **Integrated Flash.** This contains the 2M byte flash memory space interfaced via the SPI2 port of the processor. This memory space contains the application instructions and literal (constant) data. Reads from flash memory are directly cached via internal code cache. Direct memory-mapped reads are permitted through SPI memory-mapped protocol. Internal flash memory ships from the factory in an erased state except for Sector 0 and Sector 1 of the main flash array. Sector 0 and Sector 1 of the main flash array ships from the factory in an unknown state. An erase operation should be performed prior to programming this sector.
- **Internal Code Cache.** A zero-wait-state code cache SRAM memory is available internally (not visible in the memory map) to cache instruction access from internal flash as well as any externally connected serial flash and asynchronous memory.
- **MEM-X/MEM-Y.** These are virtual memory blocks which are used as cacheable memory for the code cache. No physical memory device resides inside these blocks. The application code must be compiled against these memory blocks to utilize the cache.

SRAM Region

Accesses in this region (0x2000_0000 to 0x3FFF_FFFF) are performed by the ARM Cortex-M4F core on its SYS interface. The SRAM region of the core can otherwise act as a data region for an application.

- **Internal SRAM Data Region.** This space can contain read/write data. Internal SRAM can be partitioned between CODE and DATA (SRAM region in M4 space) in 64K byte blocks. Access to this region occurs at core clock speed, with no wait states. It supports read/write access by the M4F core and read/write DMA access by system devices. It supports exclusive memory accesses via the global exclusive access monitor within the Cortex-M4F platform. Bit-banding support is also available.

System Memory Spaces

- **External SPI Flash.** Up to 16M byte of external serial quad flash memory optionally connected to the SPI0 port of the processor. Reads from flash memory are directly cached via internal code cache. Direct memory-mapped reads are permitted via SPI memory-mapped protocol.
- **System MMRs.** Various system MMRs reside in this region. Bit-banding support is available for MMRs.

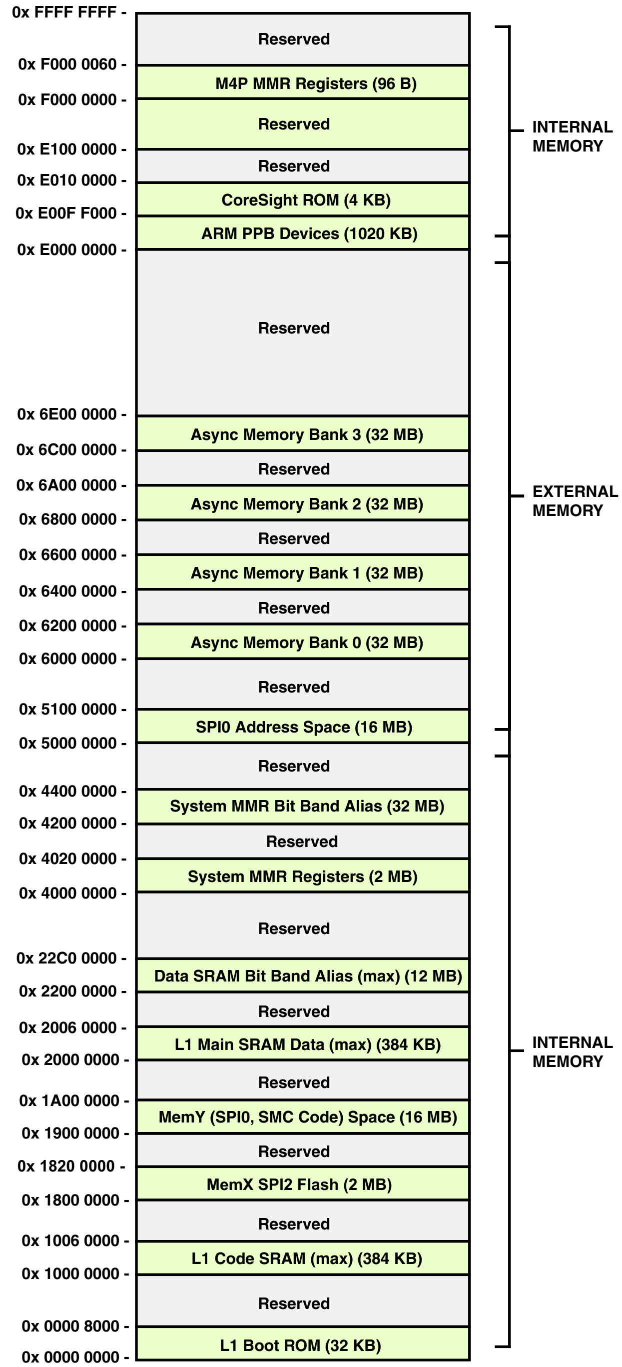

Figure 7. ADSP-CM40xF Memory Map

External Asynchronous Parallel Flash/RAM

- **L2 Asynchronous Memory.** Up to 32M byte × 4 banks of external memory can be optionally connected to the asynchronous memory port (SMC). Code execution from these memory blocks can be optionally cached via internal code cache. Direct R/W data access is also possible.

System Region

Accesses in this region (0xE000_0000 to 0xF7FF_FFFF) are performed by the ARM Cortex-M4F core on its SYS interface, and are handled within the Cortex-M4F platform. The MPU may be programmed to limit access to this space to privileged mode only.

- **CoreSight ROM.** The ROM table entries point to the debug components of the processor.
- **ARM PPB Peripherals.** This space is defined by ARM and occupies the bottom 256K byte of the SYS region (0xE000_0000 to 0xE004_0000). The space supports read/write access by the M4F core to the ARM core's internal peripherals (MPU, ITM, DWT, FPB, SCS, TPIU, ETM) and the CoreSight ROM. It is not accessible by system DMA.
- **Platform Control Registers.** This space has registers within the Cortex-M4F platform integration component that control the ARM core, its memory, and the code cache. It is accessible by the M4F core via its SYS port (but is not accessible by system DMA).

Static Memory Controller (SMC)

The SMC can be programmed to control up to four banks of external memories or memory-mapped devices, with very flexible timing parameters. On ADSP-CM407F/CM408F/CM409F processors, each bank can occupy a 32M byte segment regardless of the size of the device used.

Booting (BOOT)

The processor has several mechanisms for automatically loading internal and external memory after a reset. The boot mode is defined by the SYS_BMODE input pins dedicated for this purpose. There are two categories of boot modes. In master boot modes, the processor actively loads data from a serial memory. In slave boot modes, the processor receives data from external host devices.

The boot modes are shown in Table 2. These modes are implemented by the SYS_BMODE bits of the RCU_CTL register and are sampled during power-on resets and software-initiated resets.

Table 2. Boot Modes

SYS_BMODE[1:0] Setting	Description
00	No Boot/Idle. The processor does not boot. Rather the boot kernel executes an IDLE instruction.
01	Flash Boot. Boot from integrated Flash memory through the SPI2.
10	SPI Slave Boot. Boot through the SPI0 peripheral configured as a slave.
11	UART Boot. Boot through the UART0 peripheral configured as a slave.

SYSTEM ACCELERATION

The following sections describe the system acceleration blocks of the ADSP-CM40xF processors.

Harmonic Analysis Engine (HAE)

The harmonic analysis engine (HAE) block receives 8 kHz input samples from two source signals whose frequencies are between 45 Hz and 65 Hz. The HAE will then process the input samples and produce output results. The output results consist of power quality measurements of the fundamental and up to 12 selectable harmonics.

Sinus Cardinalis Filter (SINC)

The SINC module processes four bit streams using a pair of configurable SINC filters for each bitstream. The purpose of the primary SINC filter of each pair is to produce the filtered and decimated output for the pair. The output may be decimated to any integer rate between 8 and 256 times lower than the input rate. Greater decimation allows greater removal of noise and therefore greater ENOB.

Optional additional filtering outside the SINC module may be used to further increase ENOB. The primary SINC filter output is accessible through transfer to processor memory, or to another peripheral, via DMA.

Each of the four channels is also provided with a low-latency secondary filter with programmable positive and negative overrange detection comparators. These limit detection events can be used to interrupt the core, generate a trigger, or signal a system fault.

SECURITY FEATURES

The processor provides lightweight security functionality which protects sensitive data and IP located in the internal flash memory. It includes password-protected slave boot modes (SPI and UART), as well as password-protected JTAG/SWD debug interfaces. One of the safeguards of the security feature is the ability to perform bulk erase of the entire flash memory. Another security measure provides the ability to control which boot modes are allowed so as to protect the flash contents from untrusted or non-secure boot modes. Programs can enable or disable security features depending upon the secure header configured in internal flash memory.

CAUTION
This product includes security features that can be used to protect embedded nonvolatile memory contents and prevent execution of unauthorized code. When security is enabled on this device (either by the ordering party or the subsequent receiving parties), the ability of Analog Devices to conduct failure analysis on returned devices is limited. Contact Analog Devices for details on the failure analysis limitations for this device.

PROCESSOR RELIABILITY FEATURES

The processor provides the following features which can enhance or help achieve certain levels of system safety and reliability. While the level of safety is mainly dominated by system considerations, the following features are provided to enhance robustness.

Multi-Parity-Bit-Protected L1 Memories

In the processor's SRAM and cache L1 memory space, each word is protected by multiple parity bits to detect the single event upsets that occur in all RAMs.

Cortex MPU

The MPU divides the memory map into a number of regions, and allows the system programmer to define the location, size, access permissions, and memory attributes of each region. It supports independent attribute settings for each region, overlapping regions, and export of memory attributes to the system.

For more information, refer to the ARM Infocenter web page.

System Protection Unit (SPU)

All system resources and L2 memory banks can be controlled by either the processor core, memory-to-memory DMA, or the debug unit. A system protection unit (SPU) enables write accesses to specific resources that are locked to a given master. System protection is enabled in greater granularity for some modules through a global lock concept.

Watchpoint Protection

The primary purpose of watchpoints and hardware breakpoints is to serve emulator needs. When enabled, they signal an emulator event whenever user-defined system resources are accessed or a core executes from user-defined addresses. Watchdog events can be configured such that they signal the events to the core or to the SEC.

Software Watchdog

The on-chip watchdog timer can provide software-based supervision of the ADSP-CM40xF core.

Signal Watchdogs

The eight general-purpose timers feature two modes to monitor off-chip signals. The watchdog period mode monitors whether external signals toggle with a period within an expected range. The watchdog width mode monitors whether the pulse widths of external signals are in an expected range. Both modes help to detect incorrect undesired toggling (or lack thereof) of system-level signals.

Oscillator Watchdog

The oscillator watchdog monitors the external clock oscillator, and can detect the absence of clock as well as incorrect harmonic oscillation. The oscillator watchdog detection signal is routed to the fault management portion of the system event controller.

Low-Latency Sinc Filter Over-range Detection

The SINC filter units provide a low-latency secondary filter with programmable positive and negative limit detectors for each input channel. These may be used to monitor an isolation ADC bitstream for overrange or underrange conditions with a filter group delay as low as 0.7 μs on a 10 MHz bitstream. The secondary SINC filter events can be used to interrupt the core, to trigger other events directly in hardware using the trigger routing unit (TRU), or to signal the fault management unit of a system fault.

Up/Down Count Mismatch Detection

The GP counter can monitor external signal pairs, such as request/grant strobes. If the edge count mismatch exceeds the expected range, the up/down counter can flag this to the processor or to the system event controller (SEC).

Fault Management

The fault management unit is part of the system event controller (SEC). Most system events can be defined as faults. If defined as such, the SEC forwards the event to its fault management unit which may automatically reset the entire device for reboot, or simply toggle the $\overline{\text{SYS\_FAULT}}$ output pin to signal off-chip hardware. Optionally, the fault management unit can delay the action taken via a keyed sequence, to provide a final chance for the core to resolve the crisis and to prevent the fault action from being taken.

ADDITIONAL PROCESSOR PERIPHERALS

The processor contains a rich set of peripherals connected to the core via several concurrent high-bandwidth buses, providing flexibility in system configuration as well as excellent overall system performance (see Figure 1, Block Diagram).

The processor contains high speed serial and parallel ports, an interrupt controller for flexible management of interrupts from the on-chip peripherals or external sources, and power management control functions to tailor the performance and power characteristics of the processor and system to many application scenarios.

The following sections describe additional peripherals that were not described in the previous sections.

Timers

The processor includes several timers which are described in the following sections.

General-Purpose Timers (TIMER)

The general-purpose (GP) timer unit provides eight general-purpose programmable timers. Each timer has an external pin that can be configured either as a pulse width modulator (PWM) or timer output, as an input to clock the timer, or as a mechanism for measuring pulse widths and periods of external events. These timers can be synchronized to an external clock input on the TM0_ACLKx pins, an external signal on the TM0_CLK input pin, or to the internal SCLK.

The timer unit can be used in conjunction with the UARTs and the CAN controller to measure the width of the pulses in the data stream to provide a software auto-baud detect function for the respective serial channels.

The timer can generate interrupts to the processor core, providing periodic events for synchronization to either the system clock or to external signals. Timer events can also trigger other peripherals via the TRU (for instance, to signal a fault).

Watchdog Timer (WDT)

The core includes a 32-bit timer, which may be used to implement a software watchdog function. A software watchdog can improve system availability by forcing the processor to a known state, via generation of a general-purpose interrupt, if the timer expires before being reset by software. The programmer initializes the count value of the timer, enables the appropriate interrupt, then enables the timer. Thereafter, the software must reload the counter before it counts to zero from the programmed value. This protects the system from remaining in an unknown state where software, which would normally reset the timer, has stopped running due to an external noise condition or software error. Optionally, the fault management unit (FMU) can directly initiate the processor reset upon the watchdog expiry event.

Capture Timer (CPTMR)

The processor includes three instants of capture timers (CPTMR) to capture total on time. Each capture timer captures total on time of the input signal between two leading edges of the input trigger signal. Capture timer inputs to all the timers come from external pins and the input trigger signal comes from trigger routing unit (TRU).

The core of the timer is a 32-bit counter which is reset at leading edge of the trigger and counts when the input signal level is active. The total on time of the input signal is captured from the counter at the leading edge of the trigger pulse. Capture timer can generate data interrupts to the processor core at leading edges of trigger pulses and status interrupts to indicate counter overflow condition.

3-Phase Pulse Width Modulator Unit (PWM)

The pulse width modulator (PWM) unit provides duty cycle and phase control capabilities to a resolution of one system clock cycle (SCLK). The heightened precision PWM (HPPWM) module provides increased performance to the PWM unit by increasing its resolution by several bits, resulting in enhanced precision levels. Additional features include:

- 16-bit center-based PWM generation unit
- Programmable PWM pulse width
- Single/double update modes
- Programmable dead time and switching frequency
- Twos-complement implementation which permits smooth transition to full on and full off states
- Dedicated asynchronous PWM trip signal

The eight PWM output signals (per PWM unit) consist of four high-side drive signals and four low-side drive signals. The polarity of a generated PWM signal can be set with software, so that either active high or active low PWM patterns can be produced.

Each PWM block integrates a flexible and programmable 3-phase PWM waveform generator that can be programmed to generate the required switching patterns to drive a 3-phase voltage source inverter for ac induction motor (ACIM) or permanent magnet synchronous motor (PMSM) control. In addition, the PWM block contains special functions that considerably simplify the generation of the required PWM switching patterns for control of the electronically commutated motor (ECM) or permanent magnet synchronous motor (PMSM) control. Software can enable a special mode for switched reluctance motors (SRM).

Each PWM unit features a dedicated asynchronous trip pin which (when brought low) instantaneously places all PWM outputs in the off state.

Serial Ports (SPORTs)

The synchronous serial ports provide an inexpensive interface to a wide variety of digital and mixed-signal peripheral devices such as Analog Devices, Inc., audio codecs, ADCs, and DACs. The serial ports are made up of two data lines per direction, a clock, and frame sync. The data lines can be programmed to either transmit or receive and each data line has a dedicated DMA channel.

Serial port data can be automatically transferred to and from on-chip memory/external memory via dedicated DMA channels.For full-duplex operation, two half SPORTs can work in conjunction with clock and frame sync signals shared internally through the SPMUX block. In some operation modes, SPORT supports gated clock.

Serial ports operate in six modes:

- Standard DSP serial mode
- Multichannel (TDM) mode
- I^2S mode
- Packed I^2S mode
- Left-justified mode
- Right-justified mode

General-Purpose Counters

The 32-bit counter can operate in general-purpose up/down count modes and can sense 2-bit quadrature or binary codes as typically emitted by industrial drives or manual thumbwheels. Count direction is either controlled by a level-sensitive input pin or by two edge detectors.

A third counter input can provide flexible zero marker support and can alternatively be used to input the push-button signal of thumb wheels. All three pins have a programmable debouncing circuit.

The GP counter can also support a programmable M/N frequency scaling of the CNT_CUD and CNT_CDG pins onto output pins in quadrature encoding mode.

Internal signals forwarded to each general-purpose timer enable these timers to measure the intervals between count events. Boundary registers enable auto-zero operation or simple system warning by interrupts when programmable count values are exceeded.

Serial Peripheral Interface Ports (SPI)

The processor contains the SPI-compatible port that allows the processor to communicate with multiple SPI-compatible devices.

In its simplest mode, the SPI interface uses three pins for transferring data: two data pins master output-slave input and master input-slave output (SPI_MOSI and SPI_MISO) and a clock pin, SPI_CLK. A SPI chip select input pin ($\overline{\text{SPI\_SS}}$) lets other SPI devices select the processor, and three SPI chip select output pins (SPI_SELn) let the processor select other SPI devices. The SPI select pins are reconfigured general-purpose I/O pins. Using these pins, the SPI provides a full-duplex, synchronous serial interface, which supports both master and slave modes and multimaster environments.

In a multimaster or multislave SPI system, the MOSI and MISO data output pins can be configured to behave as open drain outputs (using the ODM bit) to prevent contention and possible damage to pin drivers. An external pull-up resistor is required on both the MOSI and MISO pins when this option is selected.

When ODM is set and the SPI is configured as a master, the MOSI pin is three-stated when the data driven out on MOSI is a logic high. The MOSI pin is not three-stated when the driven data is a logic low. Similarly, when ODM is set and the SPI is configured as a slave, the MISO pin is three-stated if the data driven out on MISO is a logic high.

The SPI port's baud rate and clock phase/polarities are programmable, and it has integrated DMA channels for both transmit and receive data streams.

Universal Asynchronous Receiver/Transmitter Ports (UART)

The processor provides full-duplex universal asynchronous receiver/transmitter (UART) ports, which are fully compatible with PC-standard UARTs. Each UART port provides a simplified UART interface to other peripherals or hosts, supporting full-duplex, DMA-supported, asynchronous transfers of serial data. A UART port includes support for five to eight data bits, and none, even, or odd parity. Optionally, an additional address bit can be transferred to interrupt only addressed nodes in multi-drop bus (MDB) systems. A frame is terminated by one, one and a half, two or two and a half stop bits.

The UART ports support automatic hardware flow control through the clear to send (CTS) input and request to send (RTS) output with programmable assertion FIFO levels.

To help support the local interconnect network (LIN) protocols, a special command causes the transmitter to queue a break command of programmable bit length into the transmit buffer. Similarly, the number of stop bits can be extended by a programmable inter-frame space.

The capabilities of the UARTs are further extended with support for the infrared data association (IrDA®) serial infrared physical layer link specification (SIR) protocol.

2-Wire Controller Interface (TWI)

The processor includes a 2-wire interface (TWI) module for providing a simple exchange method of control data between multiple devices. The TWI module is compatible with the widely used I^2C bus standard. The TWI module offers the capabilities of simultaneous master and slave operation and support for both 7-bit addressing and multimedia data arbitration. The TWI interface utilizes two pins for transferring clock (TWI_SCL) and data (TWI_SDA) and supports the protocol at speeds up to 400k bits/sec. The TWI interface pins are compatible with 5 V logic levels.

Additionally, the TWI module is fully compatible with serial camera control bus (SCCB) functionality for easier control of various CMOS camera sensor devices.

Controller Area Network (CAN)

The CAN controller implements the CAN 2.0B (active) protocol. This protocol is an asynchronous communications protocol used in both industrial and automotive control systems. The CAN protocol is well suited for control applications due to its capability to communicate reliably over a network. This is because the protocol incorporates CRC checking, message error tracking, and fault node confinement.

The CAN controller offers the following features:

- 32 mailboxes (8 receive only, 8 transmit only, 16 configurable for receive or transmit).
- Dedicated acceptance masks for each mailbox.
- Additional data filtering on first two bytes.
- Support for both the standard (11-bit) and extended (29-bit) identifier (ID) message formats.
- Support for remote frames.
- Active or passive network support.
- Interrupts, including: TX complete, RX complete, error and global.

An additional crystal is not required to supply the CAN clock, as the CAN clock is derived from a system clock through a programmable divider.

10/100 Ethernet MAC (EMAC)

The processor can directly connect to a network by way of an embedded fast Ethernet media access controller (MAC) that supports both 10-BaseT (10M bits/sec) and 100-BaseT (100M bits/sec) operation. The 10/100 Ethernet MAC peripheral on the processor is fully compliant to the IEEE 802.3-2002 standard. It provides programmable features designed to minimize supervision, bus use, or message processing by the rest of the processor system.

Some standard features are:

- Support for RMII protocols for external PHYs
- Full-duplex and half-duplex modes
- Media access management (in half-duplex operation)
- Flow control
- Station management: generation of MDC/MDIO frames for read-write access to PHY registers

Some advanced features are:

- Automatic checksum computation of IP header and IP payload fields of Rx frames
- Independent 32-bit descriptor-driven receive and transmit DMA channels
- Frame status delivery to memory through DMA, including frame completion semaphores for efficient buffer queue management in software
- Tx DMA support for separate descriptors for MAC header and payload to eliminate buffer copy operations
- Convenient frame alignment modes
- 47 MAC management statistics counters with selectable clear-on-read behavior and programmable interrupts on half maximum value
- Advanced power management
- Magic packet detection and wakeup frame filtering
- Support for 802.3Q tagged VLAN frames
- Programmable MDC clock rate and preamble suppression

IEEE 1588 Support

The IEEE 1588 standard is a precision clock synchronization protocol for networked measurement and control systems. The processor includes hardware support for IEEE 1588 with an integrated precision time protocol synchronization engine. This engine provides hardware assisted time stamping to improve the accuracy of clock synchronization between PTP nodes. The main features of the engine are:

- Support for both IEEE 1588-2002 and IEEE 1588-2008 protocol standards
- 64-bit hardware assisted time stamping for transmit and receive frames capable of up to 10 ns resolution
- Identification of PTP message type, version, and PTP payload in frames sent directly over Ethernet and transmission of the status
- Coarse and fine correction methods for system time update
- Alarm features: target time can be set to interrupt when system time reaches target time
- Pulse-Per-Second (PPS) output for physical representation of the system time. Flexibility to control the pulse-per-second output signal including control of start time, stop time, PPS output width and interval
- Automatic detection and time stamping of PTP messages over IPv4, IPv6, and Ethernet packets
- Multiple input clock sources (SCLK, RMII clock, external clock)
- Auxiliary snapshot to time stamp external events

USB 2.0 On-the-Go (OTG) Dual-Role Device Controller

The USB 2.0 on-the go (OTG) dual-role device controller provides a low-cost connectivity solution for the growing adoption of this bus standard in industrial applications, as well as consumer mobile devices such as cell phones, digital still cameras, and MP3 players. The USB 2.0 controller is a full-speed-only (FS) interface that allows these devices to transfer data using a point-to-point USB connection without the need for a PC host. The module can operate in a traditional USB peripheral-only mode as well as the host mode presented in the OTG supplement to the USB 2.0 specification.

CLOCK AND POWER MANAGEMENT

The processor provides three operating modes, each with a different performance/power profile. Control of clocking to each of the processor peripherals also reduces power consumption. See Table 3 for a summary of the power settings for each mode.

Table 3. Power Settings

Mode	CGU PLL	CGU PLL Bypassed	f_{CCLK}	f_{SCLK}	Core Power
Full On	Enabled	No	Enabled	Enabled	On
Active	Enabled	Yes	Enabled	Enabled	On
	Disabled	Yes	Enabled	Enabled	On
Deep Sleep	Disabled	—	Disabled	Disabled	On

Crystal Oscillator (SYS_XTAL)

The processor can be clocked by an external crystal (see Figure 8), a sine wave input, or a buffered, shaped clock derived from an external clock oscillator. If an external clock is used, it should be a TTL compatible signal and must not be halted, changed, or operated below the specified frequency during normal operation. This signal is connected to the processor's SYS_CLKIN pin. When an external clock is used, the SYS_XTAL pin must be left unconnected. Alternatively, because the processor includes an on-chip oscillator circuit, an external crystal may be used.

For fundamental frequency operation, use the circuit shown in Figure 8. A parallel-resonant, fundamental frequency, microprocessor grade crystal is connected across the SYS_CLKIN and XTAL pins. The on-chip resistance between SYS_CLKIN and the XTAL pin is in the 500 kΩ range. Further parallel resistors are typically not recommended.

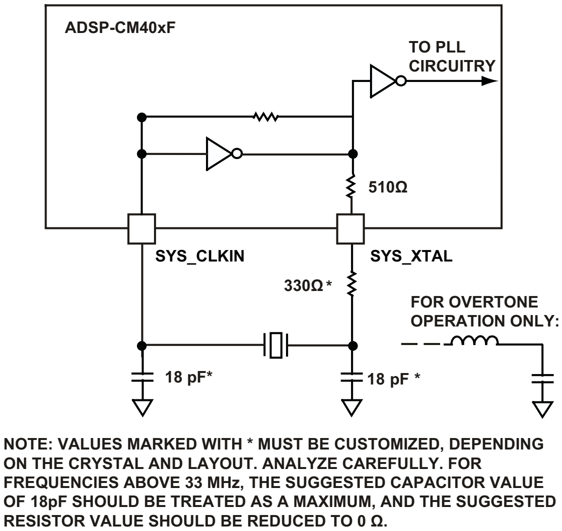

Figure 8. External Crystal Connection

The two capacitors and the 330 Ω series resistor shown in Figure 8 fine tune phase and amplitude of the sine frequency. The capacitor and resistor values shown in Figure 8 are typical values only. The capacitor values are dependent upon the crystal manufacturers' load capacitance recommendations and the PCB physical layout. The resistor value depends on the drive level specified by the crystal manufacturer. The user should verify the customized values based on careful investigations on multiple devices over temperature range.

A third-overtone crystal can be used for frequencies above 25 MHz. The circuit is then modified to ensure crystal operation only at the third overtone by adding a tuned inductor circuit as shown in Figure 8. A design procedure for third-overtone operation is discussed in detail in application note (EE-168) "Using Third Overtone Crystals with the ADSP-218x DSP" (www.analog.com/ee-168).

Oscillator Watchdog

A programmable oscillator watchdog unit is provided to allow verification of proper startup and harmonic mode of the external crystal. This allows the user to specify the expected frequency of oscillation, and to enable detection of non-oscillation and improper-oscillation faults. These events can be routed to the $\overline{\text{SYS\_FAULT}}$ output pin and/or to cause a reset of the part.

Clock Generation Unit (CGU)

The clock generation unit (CGU) generates all on-chip clocks and synchronization signals. Multiplication factors are programmed to the PLLs to define the PLLCLK frequency. Programmable values divide the PLLCLK frequency to generate the core clock (CCLK), the system clocks (SCLK), and the output clock (OCLK). This is illustrated in Figure 10 on Page 64.

Writing to the CGU control registers does not affect the behavior of the PLL immediately. Registers are first programmed with a new value, and the PLL logic executes the changes so that it transitions smoothly from the current conditions to the new ones.

SYS_CLKIN oscillations start when power is applied to the VDD_EXT pins. The rising edge of $\overline{\text{SYS\_HWRST}}$ can be applied as soon as all voltage supplies are within specifications (see Operating Conditions on Page 64), and SYS_CLKIN oscillations are stable.

Clock Out/External Clock

A SYS_CLKOUT output pin has programmable options to output divided-down versions of the on-chip clocks, including USB clocks. By default, the SYS_CLKOUT pin drives a buffered version of the SYS_CLKIN input. Clock generation faults (for example PLL unlock) may trigger a reset by hardware.

SYS_CLKOUT can be used to output one of several different clocks used on the processor. The clocks shown in Table 4 can be outputs from SYS_CLKOUT.

Table 4. SYS_CLKOUT Source and Divider Options

Clock Source	Divider
CCLK (Core Clock)	By 4
OCLK (Output Clock)	Programmable
USBCLK	Programmable
CLKBUF	None, direct from SYS_CLKIN

Power Management

As shown in Table 5 and Figure 4 on Page 6, the processor requires three different power domains, VDD_INT, VDD_EXT, and VDD_ANA. By isolating the internal logic of the processor into its own power domain, separate from other I/O, the processor can take advantage of dynamic power management without affecting the other I/O devices. There are no sequencing requirements for the various power domains, but all domains must be powered according to the appropriate Specifications table for processor operating conditions; even if the feature/peripheral is not used.

The dynamic power management feature of the processor allows the processor's core clock frequency (f_{CCLK}) to be dynamically controlled.

Table 5. Power Domains

Power Domain	V_{DD} Range
All Internal Logic	V_{DD_INT}
Digital I/O	V_{DD_EXT}
Analog	V_{DD_ANA}

The power dissipated by a processor is largely a function of its clock frequency and the square of the operating voltage. For example, reducing the clock frequency by 25% results in a 25% reduction in dynamic power dissipation. For more information on power pins, see Operating Conditions on Page 64.

Full-On Operating Mode—Maximum Performance

In the full-on mode, the PLL is enabled and is not bypassed, providing capability for maximum operational frequency. This is the execution state in which maximum performance can be achieved. The processor core and all enabled peripherals run at full speed.

For more information about PLL controls, see the "Dynamic Power Management" chapter in the *ADSP-CM40x Mixed-Signal Control Processor with ARM Cortex-M4 Hardware Reference*.

Deep Sleep Operating Mode—Maximum Dynamic Power Savings

The deep sleep mode maximizes dynamic power savings by disabling the clocks to the processor core and to all synchronous peripherals. Asynchronous peripherals may still be running but cannot access internal resources or external memory.

Voltage Regulation for VDD_INT

The internal voltage VDD_INT to the ADSP-CM40xF processors can be generated either by using an on-chip voltage regulator or by an external voltage regulator.

The VDD_INT of 1.2 V can be generated using the external I/O supply VDD_VREG of 3.3 V, which is then used to generate VDD_INT of 1.2 V. Figure 9 shows the external components required to complete the power management system for proper operation. For more details regarding component selection, refer to (EE-361) *ADSP-CM40x Power Supply Transistor Selection Guidelines* (www.analog.com/ee-361).

The internal voltage regulator can be bypassed and VDD_INT can be supplied using an external regulator. When an external regulator is used, VDD_VREG and VREG_BASE must be tied to ground for zero current consumption.

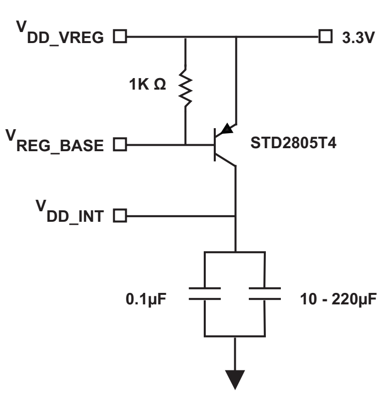

Figure 9. Internal Voltage Regulator Circuit

Reset Control Unit (RCU)

Reset is the initial state of the whole processor or of the core and is the result of a hardware or software triggered event. In this state, all control registers are set to their default values and functional units are idle. Exiting a core only reset starts with the core being ready to boot.

The reset control unit (RCU) controls how all the functional units enter and exit reset. Differences in functional requirements and clocking constraints define how reset signals are generated. Programs must guarantee that none of the reset functions puts the system into an undefined state or causes resources to stall.

From a system perspective reset is defined by both the reset target and the reset source as described below.

Target defined:

- Hardware Reset—All functional units are set to their default states without exception. History is lost.
- System Reset—All functional units except the RCU are set to their default states.

Source defined:

- Hardware Reset—The $\overline{\text{SYS\_HWRST}}$ input signal is asserted active (pulled down).
- System Reset—May be triggered by software (writing to the RCU_CTL register) or by another functional unit such as the dynamic power management (DPM) unit or any of the system event controller (SEC), trigger routing unit (TRU), or emulator inputs.
- Trigger request (peripheral).

SYSTEM DEBUG UNIT (SDU)

The processor includes various features that allow for easy system debug. These are described in the following sections.

JTAG Debug and Serial Wire Debug Port (SWJ-DP)

SWJ-DP is a combined JTAG-DP and SW-DP that enables either a serial wire debug (SWD) or JTAG probe to be connected to a target. SWD signals share the same pins as JTAG. There is an auto detect mechanism that switches between JTAG-DP and SW-DP depending on which special data sequence is used the emulator pod transmits to the JTAG pins.The SWJ-DP behaves as a JTAG target if normal JTAG sequences are sent to it and as a single wire target if the SW_DP sequence is transmitted.

Embedded Trace Macrocell (ETM) and Instrumentation Trace Macrocell (ITM)

The ADSP-CM40xF processors support both embedded trace macrocell (ETM) and instrumentation trace macrocell (ITM). These both offer an optional debug component that enables logging of real-time instruction and data flow within the CPU core. This data is stored and read through special debugger pods that have the trace feature capability. The ITM is a single-data pin feature and the ETM is a 4-data pin feature.

System Watchpoint Unit (SWU)

The system watchpoint unit (SWU) is a single module which connects to a single system bus and provides for transaction monitoring. One SWU is attached to the bus going to each system slave. The SWU provides ports for all system bus address channel signals. Each SWU contains four match groups of registers with associated hardware. These four SWU match groups operate independently, but share common event (interrupt and trigger) outputs.

DEVELOPMENT TOOLS

The ADSP-CM40xF processor is supported with a set of highly sophisticated and easy-to-use development tools for embedded applications. For more information, see the Analog Devices website.

ADDITIONAL INFORMATION

The following publications that describe the ADSP-CM40xF processors (and related processors) can be ordered from any Analog Devices sales office or accessed electronically on our website:

- *ADSP-CM40x Mixed-Signal Control Processor with ARM Cortex-M4 Hardware Reference*
- *ADSP-CM402F/CM403F/CM407F/CM408F/CM409F Anomaly Sheet*

This data sheet describes the ARM Cortex-M4 core and memory architecture used on the ADSP-CM40xF processor, but does not provide detailed programming information for the ARM processor. For more information about programming the ARM processor, visit the ARM Infocenter web page.

The applicable documentation for programming the ARM Cortex-M4 processor include:

- *Cortex®-M4 Devices Generic User Guide*
- *CoreSight™ ETM™-M4 Technical Reference Manual*
- *Cortex®-M4 Technical Reference Manual*

RELATED SIGNAL CHAINS

A signal chain is a series of signal-conditioning electronic components that receive input (data acquired from sampling either real-time phenomena or from stored data) in tandem, with the output of one portion of the chain supplying input to the next. Signal chains are often used in signal processing applications to gather and process data or to apply system controls based on analysis of real-time phenomena.

Analog Devices eases signal processing system development by providing signal processing components that are designed to work together well. A tool for viewing relationships between specific applications and related components is available on the www.analog.com website.

The application signal chains page in the Circuits from the Lab® site (http:\\www.analog.com\circuits) provides:

- Graphical circuit block diagram presentation of signal chains for a variety of circuit types and applications
- Drill down links for components in each chain to selection guides and application information
- Reference designs applying best practice design techniques

SECURITY FEATURES DISCLAIMER

To our knowledge, the Security Features, when used in accordance with the data sheet and hardware reference manual specifications, provide a secure method of implementing code and data safeguards. However, Analog Devices does not guarantee that this technology provides absolute security. ACCORDINGLY, ANALOG DEVICES HEREBY DISCLAIMS ANY AND ALL EXPRESS AND IMPLIED WARRANTIES THAT THE SECURITY FEATURES CANNOT BE BREACHED, COMPROMISED, OR OTHERWISE CIRCUMVENTED AND IN NO EVENT SHALL ANALOG DEVICES BE LIABLE FOR ANY LOSS, DAMAGE, DESTRUCTION, OR RELEASE OF DATA, INFORMATION, PHYSICAL PROPERTY, OR INTELLECTUAL PROPERTY.

ADSP-CM40xF DETAILED SIGNAL DESCRIPTIONS

Table 6 provides a detailed description of each pin.

Table 6. ADSP-CM40xF Detailed Signal Description

Signal Name	Direction	Description
ADC_VINnn	Input	**Channel nn.** Single-Ended Analog Input for ADCs. nn = 00 to 11 for each ADC
BYP_An		**On-chip Analog Power Regulation Bypass Filter Node for ADC.** Connect to decoupling capacitors. n = 0, 1
BYP_D0		**On-chip Digital Power Regulation Bypass Filter Node for Analog Subsystem.** Connect to decoupling capacitors.
CAN_RX	Input	**CAN Receive.** Typically an external CAN transceiver's RX output.
CAN_TX	Output	**CAN Transmit.** Typically an external CAN transceiver's TX input.
CNT_OUTA	Output	**Counter Output Divider A.** Frequency scaled output in Quadrature encoder mode of GP Counter
CNT_OUTB	Output	**Counter Output Divider B.** Frequency scaled output in Quadrature encoder mode of GP Counter
CNT_DG	Input	**CNT Count Down and Gate.** Depending on the mode of operation this input acts either as a count down signal or a gate signal. Count Down: This input causes the GP counter to decrement. Gate: Stops the GP counter from incrementing or decrementing.
CNT_UD	Input	**Count Up and Direction.** Depending on the mode of operation this input acts either as a count up signal or a direction signal. Count Up: This input causes the GP counter to increment. Direction: Selects whether the GP counter is incrementing or decrementing.
CNT_ZM	Input	**Count Zero Marker.** Input that connects to the zero marker output of a rotary device or detects the pressing of a push button.
CPTMR_INn	Input	**Capture Timer Input Pins.** n = 0, 1, 2
DACn_VOUT	Output	**DAC Output.** Analog voltage output. n = 0, 1
ETH_CRS	Input	**EMAC Carrier Sense.** Multiplexed on alternate clock cycles. CRS: Asserted by the PHY when either the transmit or receive medium is not idle. De-asserted when both are idle. RXDV: Asserted by the PHY when the data on RXDn is valid.
ETH_MDC	Output	**EMAC Management Channel Clock.** Clocks the MDC input of the PHY.
ETH_MDIO	I/O	**EMAC Management Channel Serial Data.** Bidirectional data bus for PHY control.
ETH_PTPAUXIN	Input	**EMAC PTP Auxiliary Trigger Input.** Assert this signal to take an auxiliary snapshot of the time and store it in the auxiliary time stamp FIFO.
ETH_PTPCLKIN	Input	**EMAC PTP Clock Input.** Optional external PTP clock input.
ETH_PTPPPS	Output	**EMAC PTP Pulse-Per-Second Output.** When the Advanced Time Stamp feature is enabled, this signal is asserted based on the PPS mode selected. Otherwise, PTPPPS is asserted every time the seconds counter is incremented.
ETH_REFCLK	Input	**EMAC Reference Clock.** Externally supplied Ethernet clock.
ETH_RXDn	Input	**EMAC Receive Data n.** Receive data bus. n = 0, 1
ETH_TXDn	Output	**EMAC Transmit Data n.** Transmit data bus. n = 0, 1
ETH_TXEN	I/O	**EMAC Transmit Enable.** When asserted indicates that the data on TXDn is valid.
JTG_SWCLK	I/O	**Serial Wire Clock.** Clocks data into and out of the target during debug.
JTG_SWDIO	I/O	**Serial Wire Data IO.** Sends and receives serial data to and from the target during debug.
JTG_SWO	Output	**Serial Wire Out.** Provides trace data to the emulator.
JTG_TCK	Input	**JTAG Clock.** JTAG test access port clock.
JTG_TDI	Input	**JTAG Serial Data In.** JTAG test access port data input.
JTG_TDO	Output	**JTAG Serial Data Out.** JTAG test access port data output.
JTG_TMS	Input	**JTAG Mode Select.** JTAG test access port mode select.

Table 6. ADSP-CM40xF Detailed Signal Description (Continued)

Signal Name	Direction	Description
JTG_TRST	Input	**JTAG Reset.** JTAG test access port reset.
Px_nn	I/O	**Position n.** General purpose input/output. See the GP Ports chapter in the processor hardware reference for programming information.
PWM_AH	Output	**PWM Channel A High Side.** High side drive signal.
PWM_AL	Output	**PWM Channel A Low Side.** Low side drive signal.
PWM_BH	Output	**PWM Channel B High Side.** High side drive signal.
PWM_BL	Output	**PWM Channel B Low Side.** Low side drive signal.
PWM_CH	Output	**PWM Channel C High Side.** High side drive signal.
PWM_CL	Output	**PWM Channel C Low Side.** Low side drive signal.
PWM_DH	Output	**PWM Channel D High Side.** High side drive signal.
PWM_DL	Output	**PWM Channel D Low Side.** Low side drive signal.
PWM_SYNC	I/O	**PWM Synchronization signal.** This is an input pin when PWM is configured to receive external sync signal. It is an output pin when PWM Sync is generated internally.
PWM_TRIPn	Input	**PWM Trip Input.** When asserted the selected PWM channel outputs are shut down immediately.
REFCAP	Analog	Output of BandGap Generator Filter Node
SINC_CLKn	Output	**SINC Clock n.** n = 0, 1
SINC_Dn	Input	**SINC Data n.** n = 0 to 3
SMC_Ann	Output	**SMC Address n.** Address bus. n = 0 to 24
SMC_ABEn	Output	**SMC Byte Enable n.** Indicates whether the lower or upper byte of a memory is being accessed. When an asynchronous write is made to the upper byte of a 16-bit memory, SMC_ABE1 = 0 and SMC_ABE0 = 1. When an asynchronous write is made to the lower byte of a 16-bit memory, SMC_ABE1 = 1 and SMC_ABE0 = 0.
SMC_AMSn	Output	**SMC Memory Select n.** Typically connects to the chip select of a memory device. n = 0, 1, 2, 3
SMC_AOE	Output	**SMC Output Enable.** Asserts at the beginning of the setup period of a read access.
SMC_ARDY	Input	**SMC Asynchronous Ready.** Flow control signal used by memory devices to indicate to the SMC when further transactions may proceed.
SMC_ARE	Output	**SMC Read Enable.** Asserts at the beginning of a read access.
SMC_AWE	Output	**SMC Write Enable.** Asserts for the duration of a write access period.
SMC_Dnn	I/O	**SMC Data n.** Bidirectional data bus. n = 0 to 15
SPI_CLK	I/O	**SPI Clock.** Input in slave mode, output in master mode.
SPI_D2	I/O	**SPI Data 2.** Used to transfer serial data in quad mode. Open drain in ODM mode.
SPI_D3	I/O	**SPI Data 3.** Used to transfer serial data in quad mode. Open drain in ODM mode.
SPI_MISO	I/O	**SPI Master In, Slave Out.** Used to transfer serial data. Operates in the same direction as SPI_MOSI in dual and quad modes. Open drain in ODM mode.
SPI_MOSI	I/O	**SPI Master Out, Slave In.** Used to transfer serial data. Operates in the same direction as SPI_MISO in dual and quad modes. Open drain in ODM mode.
SPI_RDY	I/O	**SPI Ready.** Optional flow signal to hold-off faster masters. Output in slave mode, input in master mode.
SPI_SELn	Output	**SPI Slave Select Output n.** Used in master mode to enable the desired slave.
SPI_SS	Input	**SPI Slave Select Input.** Slave mode: acts as the slave select input. Master mode: optionally serves as an error detection input for the SPI when there are multiple masters.
SPT_ACLK	I/O	**SPORT A Channel Clock.** Data and frame sync are driven/sampled with respect to this clock. This signal can be either internally or externally generated.
SPT_AD0	I/O	**SPORT A Channel Data 0.** Primary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_AD1	I/O	**SPORT A Channel Data 1.** Secondary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.

Table 6. ADSP-CM40xF Detailed Signal Description (Continued)

Signal Name	Direction	Description
SPT_AFS	I/O	**SPORT A Channel Frame Sync.** The frame sync pulse initiates shifting of serial data. This signal is either generated internally or externally.
SPT_ATDV	Output	**SPORT A Channel Transmit Data Valid.** This signal is optional and only active when SPORT is configured in multi-channel transmit mode. It is asserted during enabled slots.
SPT_BCLK	I/O	**SPORT B Channel Clock.** Data and frame sync are driven/sampled with respect to this clock. This signal can be either internally or externally generated.
SPT_BD0	I/O	**SPORT B Channel Data 0.** Primary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_BD1	I/O	**SPORT B Channel Data 1.** Secondary bidirectional data I/O. This signal can be configured as an output to transmit serial data, or as an input to receive serial data.
SPT_BFS	I/O	**SPORT B Channel Frame Sync.** The frame sync pulse initiates shifting of serial data. This signal is either generated internally or externally.
SPT_BTDV	Output	**SPORT B Channel Transmit Data Valid.** This signal is optional and only active when SPORT is configured in multi-channel transmit mode. It is asserted during enabled slots.
SYS_BMODEn	Input	**Boot Mode Control n.** Selects the boot mode of the processor. n = 0, 1
SYS_CLKIN	Input	**Processor Clock/Crystal Input.** Connect to an external clock source or crystal.
SYS_CLKOUT	Output	**Processor Clock Output.** Outputs internal clocks. Clocks may be divided down. See the CGU chapter in the processor hardware reference for more details.
SYS_DSWAKEn	Input	**System Deep Sleep Wakeup inputs.** n = 0 to 3
$\overline{\text{SYS\_FAULT}}$	Output	**System Fault.** Indicates system fault.
$\overline{\text{SYS\_HWRST}}$	Input	**Processor Hardware Reset Control.** Resets the device when asserted.
$\overline{\text{SYS\_NMI}}$	Input	**Non-maskable Interrupt.** See the processor hardware and programming references for more details.
$\overline{\text{SYS\_RESOUT}}$	Output	**Processor Reset Output.** Indicates that the device is in the reset state.
SYS_XTAL	Output	**System Crystal Output.** Drives an external crystal. Must be left unconnected if an external clock is driving CLKIN.
TM_ACIn	Input	**GP Timer Alternate Capture Input n.** Provides an additional input for GP Timers in WIDCAP, WATCHDOG, and PININT modes. n = 0 to 5
TM_ACLKn	Input	**GP Timer Alternate Clock n.** Provides an additional time base for use by an individual timer. n = 0 to 5
TM_CLK	Input	**GP Timer Clock.** Provides an additional global time base for use by all the GP timers.
TM_TMRn	I/O	**GP Timer Timer n.** The main input/output signal for each timer. n = 0 to 7. In PWM OUT mode, output is driven on this pin. In Width capture mode, it acts as input and Timer measures width and/or period of incoming signal on this pin. In EXTCLK mode, Timer counts number of incoming signal edges on this pin.
TRACE_CLK	Output	**Embedded Trace Module Clock.** Reference clock for the Trace Unit.
TRACE_Dn	Output	**Embedded Trace Module Data n.** Output data for clocked modes and changes on both edges of TRACE_CLK. n = 0 to 3
TWI_SCL	I/O	**TWI Serial Clock.** Clock output when master, clock input when slave. Compatible with I^{2}C bus standard.
TWI_SDA	I/O	**TWI Serial Data.** Receives or transmits data. Compatible with I^{2}C bus standard.
$\overline{\text{UART\_CTS}}$	Input	**UART Clear to Send.** Input Hardware Flow control signal. Transmitter initiates the transfer only when this signal is active.
$\overline{\text{UART\_RTS}}$	Output	**UART Request to Send.** Output Hardware Flow control signal. Receiver activates this signal when it is ready to receive new transfers.
$\overline{\text{UART\_RX}}$	Input	**UART Receive.** Receive input. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
$\overline{\text{UART\_TX}}$	Output	**UART Transmit.** Transmit output. Typically connects to a transceiver that meets the electrical requirements of the device being communicated with.
USB_DM	I/O	**USB Data –.** Bidirectional differential data line.
USB_DP	I/O	**USB Data +**. Bidirectional differential data line.

Table 6. ADSP-CM40xF Detailed Signal Description (Continued)

Signal Name	Direction	Description
USB_ID	Input	**USB OTG ID.** Senses whether the controller is a host or device. This signal is pulled low when an A-type plug is sensed (signifying that the USB controller is the A device), but the input is high when a B-type plug is sensed (signifying that the USB controller is the B device).
USB_VBC	Output	**USB VBUS Control.** Controls an external voltage source to supply VBUS when in host mode. May be configured as open drain. Polarity is configurable as well.
USB_VBUS	I/O	**USB Bus Voltage.** Connects to bus voltage in host and device modes.
VREFn	I/O	**Voltage Reference for ADC.** When internal reference is selected for ADC, the VREF pin is used for connecting bypass caps. When external reference is selected, an external reference device should be connected to these pins to supply the external reference voltage. n=0,1.
VREG_BASE	Output	**Voltage Regulator Base Node.** Connected to Base of PNP transistor when using internal VDD_INT reference.

ADSP-CM402F/ADSP-CM403F 120-LEAD LQFP SIGNAL DESCRIPTIONS

The processor's pin definitions are shown in Table 7. The columns in this table provide the following information:

- Signal Name: The Signal Name column in the table includes the signal name for every pin and (where applicable) the GPIO multiplexed pin function for every pin.
- Description: The Description column in the table provides a verbose (descriptive) name for the signal.
- General-Purpose Port: The Port column in the table shows whether or not the signal is multiplexed with other signals on a general-purpose I/O port pin.
- Pin Name: The Pin Name column in the table identifies the name of the package pin (at power on reset) on which the signal is located (if a single function pin) or is multiplexed (if a general-purpose I/O pin).

Table 7. ADSP-CM402F/ADSP-CM403F 120-Lead LQFP Signal Descriptions

Signal Name	Description	Port	Pin Name
ADC0_VIN00	Channel 0 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN00
ADC0_VIN01	Channel 1 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN01
ADC0_VIN02	Channel 2 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN02
ADC0_VIN03	Channel 3 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN03
ADC0_VIN04	Channel 4 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN04
ADC0_VIN05	Channel 5 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN05
ADC0_VIN06	Channel 6 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN06
ADC0_VIN07	Channel 7 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN07
ADC0_VIN08	Channel 8 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN08
ADC0_VIN09	Channel 9 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN09
ADC0_VIN10	Channel 10 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN10
ADC0_VIN11	Channel 11 Single-Ended Analog Input for ADC0	Not Muxed	ADC0_VIN11
ADC1_VIN00	Channel 0 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN00
ADC1_VIN01	Channel 1 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN01
ADC1_VIN02	Channel 2 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN02
ADC1_VIN03	Channel 3 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN03
ADC1_VIN04	Channel 4 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN04
ADC1_VIN05	Channel 5 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN05
ADC1_VIN06	Channel 6 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN06
ADC1_VIN07	Channel 7 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN07
ADC1_VIN08	Channel 8 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN08
ADC1_VIN09	Channel 9 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN09
ADC1_VIN10	Channel 10 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN10
ADC1_VIN11	Channel 11 Single-Ended Analog Input for ADC1	Not Muxed	ADC1_VIN11
BYP_A0	On-chip Analog Power Regulation Bypass Filter Node for ADC0 (see recommended bypass - Figure 4 on Page 6)	Not Muxed	BYP_A0
BYP_A1	On-chip Analog Power Regulation Bypass Filter Node for ADC1 (see recommended bypass - Figure 4 on Page 6)	Not Muxed	BYP_A1
BYP_D0	On-chip Digital Power Regulation Bypass Filter Node for Analog Subsystem (see recommended bypass - Figure 4 on Page 6)	Not Muxed	BYP_D0
CAN0_RX	CAN0 Receive	B	PB_15
CAN0_TX	CAN0 Transmit	C	PC_00
CAN1_RX	CAN1 Receive	B	PB_10
CAN1_TX	CAN1 Transmit	B	PB_11
CNT0_DG	CNT0 Count Down and Gate	B	PB_02
CNT0_OUTA	CNT0 Output Divider A	B	PB_13

Table 7. ADSP-CM402F/ADSP-CM403F 120-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
CNT0_OUTB	CNT0 Output Divider B	B	PB_14
CNT0_UD	CNT0 Count Up and Direction	B	PB_01
CNT0_ZM	CNT0 Count Zero Marker	B	PB_00
CNT1_DG	CNT1 Count Down and Gate	B	PB_05
CNT1_UD	CNT1 Count Up and Direction	B	PB_04
CNT1_ZM	CNT1 Count Zero Marker	B	PB_03
CPTMR0_IN0	CPTMR0 Capture Timer0 Input 0	B	PB_07
CPTMR0_IN1	CPTMR0 Capture Timer0 Input 1	B	PB_08
CPTMR0_IN2	CPTMR0 Capture Timer0 Input 2	B	PB_09
DAC0_VOUT	Analog Voltage Output 0	Not Muxed	DAC0_VOUT
DAC1_VOUT	Analog Voltage Output 1	Not Muxed	DAC1_VOUT
GND	Digital Ground	Not Muxed	GND
GND_ANA0	Analog Ground return for VDD_ANA0 (see recommended bypass - Figure 4 on Page 6)	Not Muxed	GND_ANA0
GND_ANA1	Analog Ground return for VDD_ANA1 (see recommended bypass - Figure 4 on Page 6)	Not Muxed	GND_ANA1
GND_ANA2	Analog Ground (see recommended bypass - Figure 4 on Page 6)	Not Muxed	GND_ANA2
GND_ANA3	Analog Ground (see recommended bypass - Figure 4 on Page 6)	Not Muxed	GND_ANA3
GND_VREF0	Ground return for VREF0 (see recommended bypass filter - Figure 4 on Page 6)	Not Muxed	GND_VREF0
GND_VREF1	Ground return for VREF1 (see recommended bypass filter - Figure 4 on Page 6)	Not Muxed	GND_VREF1
JTG_TCK/SWCLK	JTAG Clock/Serial Wire Clock	Not Muxed	JTG_TCK/SWCLK
JTG_TDI	JTAG Serial Data In	Not Muxed	JTG_TDI
JTG_TDO/SWO	JTAG Serial Data Out/Serial Wire Trace Output	Not Muxed	JTG_TDO/SWO
JTG_TMS/SWDIO	JTAG Mode Select/Serial Wire Debug Data I/O	Not Muxed	JTG_TMS/SWDIO
$\overline{\text{JTG\_TRST}}$	JTAG Reset	Not Muxed	$\overline{\text{JTG\_TRST}}$
PA_00-PA_15	Port A Positions 0 – 15	A	PA_00 – PA_15
PB_00-PB_15	Port B Positions 0 – 15	B	PB_00 – PB_15
PC_00-PC_07	Port C Positions 0 – 7	C	PC_00 – PC_07
PWM0_AH	PWM0 Channel A High Side	A	PA_02
PWM0_AL	PWM0 Channel A Low Side	A	PA_03
PWM0_BH	PWM0 Channel B High Side	A	PA_04
PWM0_BL	PWM0 Channel B Low Side	A	PA_05
PWM0_CH	PWM0 Channel C High Side	A	PA_06
PWM0_CL	PWM0 Channel C Low Side	A	PA_07
PWM0_DH	PWM0 Channel D High Side	B	PB_00
PWM0_DL	PWM0 Channel D Low Side	B	PB_01
PWM0_SYNC	PWM0 Sync	A	PA_00
$\overline{\text{PWM0\_TRIP0}}$	PWM0 Trip Input 0	A	PA_01
PWM1_AH	PWM1 Channel A High Side	A	PA_12
PWM1_AL	PWM1 Channel A Low Side	A	PA_13
PWM1_BH	PWM1 Channel B High Side	A	PA_14
PWM1_BL	PWM1 Channel B Low Side	A	PA_15
PWM1_CH	PWM1 Channel C High Side	A	PA_08
PWM1_CL	PWM1 Channel C Low Side	A	PA_09
PWM1_DH	PWM1 Channel D High Side	B	PB_02

Table 7. ADSP-CM402F/ADSP-CM403F 120-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
PWM1_DL	PWM1 Channel D Low Side	B	PB_03
PWM1_SYNC	PWM1 Sync	A	PA_10
PWM1_TRIP0	PWM1 Trip Input 0	A	PA_11
PWM2_AH	PWM2 Channel A High Side	B	PB_06
PWM2_AL	PWM2 Channel A Low Side	B	PB_07
PWM2_BH	PWM2 Channel B High Side	B	PB_08
PWM2_BL	PWM2 Channel B Low Side	B	PB_09
PWM2_CH	PWM2 Channel C High Side	C	PC_03
PWM2_CL	PWM2 Channel C Low Side	C	PC_04
PWM2_DH	PWM2 Channel D High Side	C	PC_05
PWM2_DL	PWM2 Channel D Low Side	C	PC_06
PWM2_SYNC	PWM2 Sync	B	PB_04
PWM2_TRIP0	PWM2 Trip Input 0	B	PB_05
REFCAP	Output of BandGap Generator Filter Node (see recommended bypass filter - Figure 4 on Page 6)	Not Muxed	REFCAP
SINC0_CLK0	SINC0 Clock 0	B	PB_10
SINC0_CLK1	SINC0 Clock 1	C	PC_07
SINC0_D0	SINC0 Data 0	B	PB_11
SINC0_D1	SINC0 Data 1	B	PB_12
SINC0_D2	SINC0 Data 2	B	PB_13
SINC0_D3	SINC0 Data 3	B	PB_14
SMC0_A01	SMC0 Address 1	B	PB_13
SMC0_A02	SMC0 Address 2	B	PB_14
SMC0_A03	SMC0 Address 3	B	PB_15
SMC0_A04	SMC0 Address 4	C	PC_00
SMC0_A05	SMC0 Address 5	C	PC_01
SMC0_AMS0	SMC0 Memory Select 0	B	PB_11
SMC0_AMS2	SMC0 Memory Select 2	A	PA_07
SMC0_AOE	SMC0 Output Enable	B	PB_12
SMC0_ARDY	SMC0 Asynchronous Ready	B	PB_08
SMC0_ARE	SMC0 Read Enable	B	PB_09
SMC0_AWE	SMC0 Write Enable	B	PB_10
SMC0_D00	SMC0 Data 0	A	PA_08
SMC0_D01	SMC0 Data 1	A	PA_09
SMC0_D02	SMC0 Data 2	A	PA_10
SMC0_D03	SMC0 Data 3	A	PA_11
SMC0_D04	SMC0 Data 4	A	PA_12
SMC0_D05	SMC0 Data 5	A	PA_13
SMC0_D06	SMC0 Data 6	A	PA_14
SMC0_D07	SMC0 Data 7	A	PA_15
SMC0_D08	SMC0 Data 8	B	PB_00
SMC0_D09	SMC0 Data 9	B	PB_01
SMC0_D10	SMC0 Data 10	B	PB_02
SMC0_D11	SMC0 Data 11	B	PB_03
SMC0_D12	SMC0 Data 12	B	PB_04
SMC0_D13	SMC0 Data 13	B	PB_05

Table 7. ADSP-CM402F/ADSP-CM403F 120-Lead LQFP Signal Descriptions (Continued)

Signal Name	Description	Port	Pin Name
SMC0_D14	SMC0 Data 14	B	PB_06
SMC0_D15	SMC0 Data 15	B	PB_07
SPI0_CLK	SPI0 Clock	C	PC_03
SPI0_D2	SPI0 Data 2	B	PB_10
SPI0_D3	SPI0 Data 3	B	PB_11
SPI0_MISO	SPI0 Master In, Slave Out	C	PC_04
SPI0_MOSI	SPI0 Master Out, Slave In	C	PC_05
SPI0_RDY	SPI0 Ready	C	PC_02
SPI0_SEL1	SPI0 Slave Select Output 1	C	PC_06
SPI0_SEL2	SPI0 Slave Select Output 2	B	PB_13
SPI0_SEL3	SPI0 Slave Select Output 3	B	PB_14
SPI0_SS	SPI0 Slave Select Input	B	PB_14
SPT0_ACLK	SPORT0 Channel A Clock	B	PB_00
SPT0_AD0	SPORT0 Channel A Data 0	B	PB_02
SPT0_AD1	SPORT0 Channel A Data 1	B	PB_03
SPT0_AFS	SPORT0 Channel A Frame Sync	B	PB_01
SPT0_ATDV	SPORT0 Channel A Transmit Data Valid	B	PB_04
SPT1_ACLK	SPORT1 Channel A Clock	A	PA_00
SPT1_AD0	SPORT1 Channel A Data 0	A	PA_02
SPT1_AD1	SPORT1 Channel A Data 1	A	PA_03
SPT1_AFS	SPORT1 Channel A Frame Sync	A	PA_01
SPT1_ATDV	SPORT1 Channel A Transmit Data Valid	B	PB_15
SPT1_BCLK	SPORT1 Channel B Clock	A	PA_04
SPT1_BD0	SPORT1 Channel B Data 0	A	PA_06
SPT1_BD1	SPORT1 Channel B Data 1	A	PA_07
SPT1_BFS	SPORT1 Channel B Frame Sync	A	PA_05
SPT1_BTDV	SPORT1 Channel B Transmit Data Valid	C	PC_00
SYS_BMODE0	Boot Mode Control 0	Not Muxed	SYS_BMODE0
SYS_BMODE1	Boot Mode Control 1	Not Muxed	SYS_BMODE1
SYS_CLKIN	Clock/Crystal Input	Not Muxed	SYS_CLKIN
SYS_CLKOUT	Processor Clock Output	Not Muxed	SYS_CLKOUT
SYS_DSWAKE0	Deep Sleep Wake-up 0	C	PC_06
SYS_DSWAKE1	Deep Sleep Wake-up 1	C	PC_07
SYS_DSWAKE2	Deep Sleep Wake-up 2	B	PB_14
SYS_DSWAKE3	Deep Sleep Wake-up 3	B	PB_13
SYS_FAULT	System Fault Output	Not Muxed	SYS_FAULT
SYS_HWRST	Processor Hardware Reset Control	Not Muxed	SYS_HWRST
SYS_NMI	Nonmaskable Interrupt	Not Muxed	SYS_NMI
SYS_RESOUT	Reset Output	Not Muxed	SYS_RESOUT
SYS_XTAL	Crystal Output	Not Muxed	SYS_XTAL
TM0_ACI1	TIMER0 Alternate Capture Input 1	B	PB_10
TM0_ACI2	TIMER0 Alternate Capture Input 2	B	PB_08
TM0_ACI3	TIMER0 Alternate Capture Input 3	B	PB_12
TM0_ACI4	TIMER0 Alternate Capture Input 4	B	PB_15
TM0_ACI5	TIMER0 Alternate Capture Input 5	C	PC_01
TM0_ACLK0	TIMER0 Alternate Clock 0	B	PB_13