Technical Brief
SPRT731A–June 2017–Revised January 2018

The TMS320F28004x Microcontroller: A Comparison to the TMS320F2806x and TMS320F2803x Microcontrollers

Kenneth W. Schachter *C2000 Technical Staff*

ABSTRACT

The Texas Instruments Piccolo™ TMS320F28004x microcontroller (MCU) device series offers an optimal balance between performance and cost for many real-time control systems. This document discusses the technical details of the TMS320F28004x architecture and highlights the new improvements to various key peripherals, such as an enhanced Type-2 CLA capable of running a background task, and the inclusion of a set of high-speed programmable gain amplifiers. Also, a completely new boot mode flow enables expanded booting options. A comparison to the Piccolo TMS320F2806x and TMS320F2803x MCU device series is used to assist with the understanding of the new features. This document may also be useful when considering migration from the TMS320F2806x or TMS320F2803x MCU device series to the TMS320F28004 MCU. Some knowledge about these previous device architectures is helpful in understanding the topics discussed in this document.

Contents

List of Figures

List of Tables

Trademarks

1 Introduction

Many real-time control systems require a high-performance microcontroller (MCU) for cost-sensitive applications. However, it can be a challenge to optimize these two parameters: performance and cost. To meet this challenge, the Texas Instruments Piccolo™ TMS320F28004x MCU device series, referred to as the F28004x in this document, was developed. This MCU incorporates critical control peripherals, analog components, communications peripherals, and nonvolatile memory in a single device. This level of integration provides a cohesive real-time control solution; increasing system performance and reliability while simplifying the board layout which reduces the overall system cost. The purpose of this document is to highlight the various key enhancements to the architecture and peripherals which are new to the F28004x. To assist in understanding these enhancements, a comparison to the Piccolo TMS320F2806x and TMS320F2803x MCU device series is used. In this document the TMS320F2806x and the TMS320F2803x is referred to as the F2806x and F2803x, respectively. Also, F2806x/03x is used to refer to both MCU device series, collectively. Therefore, some working knowledge about these previous C2000™ MCU device architectures is helpful in understanding the topics presented in this document.

2 Processor and Accelerators

The F28004x CPU is based on the TI 32-bit C28x fixed-point accumulator-based architecture and it is capable of operating at a clock frequency of up to 100 MHz. While the F2806x/03x utilizes the same CPU architecture, the F2806x and F2803x operate at a maximum clock frequency of 90 MHz and 60 MHz, respectively. The CPU in the F28004x and F2806x is tightly coupled with a Floating-Point Unit (FPU) which enables support for hardware IEEE-754 single-precision floating-point format operations, whereas the F2803x does not have a FPU. Also, the F28004x has a tightly coupled Trigonometric Math Unit (TMU) which extends the capability of the CPU to efficiently execute trigonometric and arithmetic operations commonly found in control system applications. Similar to the FPU, the TMU provides hardware support for IEEE-754 single-precision floating-point operations which accelerate trigonometric math functions. A Viterbi, Complex Math, and CRC Unit (VCU) further extends the capabilities of the CPU for supporting various communication-based algorithms and it is very useful for general-purpose signal processing applications, such as filtering and spectral analysis. The F28004x and F2806x utilize the same VCU-I (Type 1). The VCU is not available on the F2803x. A “Type” change represents a major functional feature difference in a peripheral module.

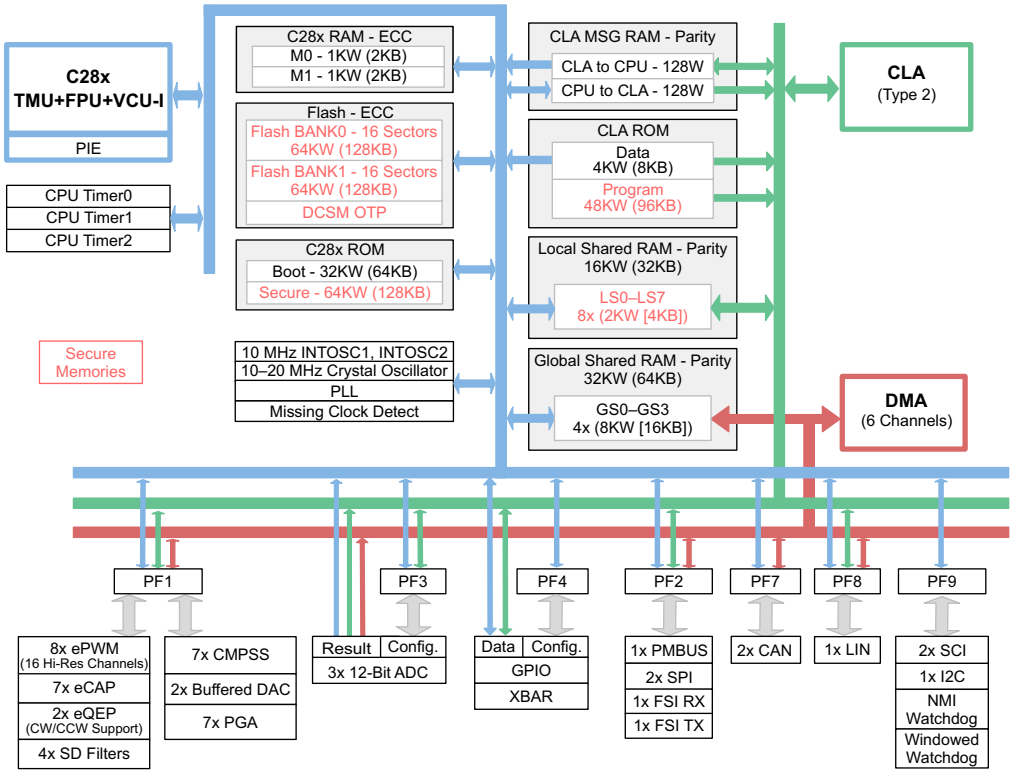

Figure 1. F28004x Functional Block Diagram

The Control Law Accelerator (CLA) is an independent 32-bit floating-point math hardware accelerator which executes real-time control algorithms in parallel with the main C28x CPU, effectively doubling the computational performance. With direct access to the various control peripherals (and communication peripherals on the F28004x), the CLA minimizes latency, enables a fast trigger response, and avoids CPU overhead. Also, with direct access to the ADC results registers, the CLA is able to read the result on the same cycle that the ADC sample conversion is completed, providing “just-in-time” reading, which reduces the sample to output delay.

The F2806x/03x utilizes a Type 0 CLA which has access to a fixed program memory RAM block (L3 DPSARAM) and two or three data memory RAM blocks (L1 and L2 DPSARAM on the F2803x, and the addition of L0 DPSARAM on the F2806x). Furthermore, the F2806x includes an option to enable CPU access to the data memory RAM blocks when mapped to CLA data space. The Type 0 CLA has a 12-bit program counter providing a 4K address memory reach, and each of the eight CLA Tasks can be triggered by up to three (for the F2803x) or eight (for the F2806x) peripheral interrupt sources.

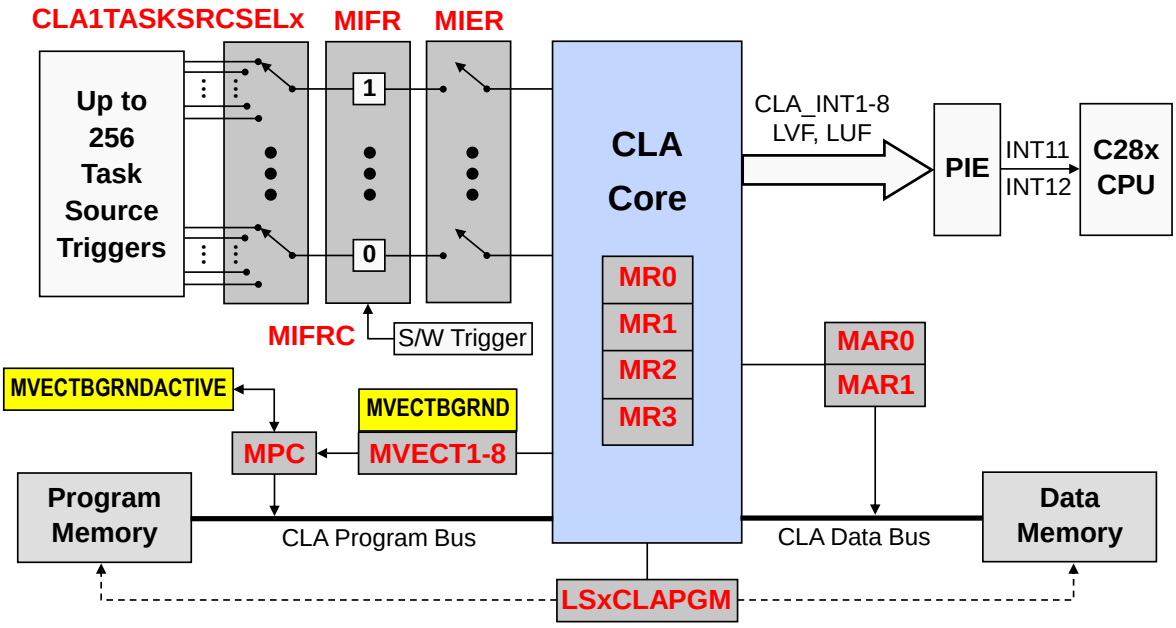

Figure 2. Type 2 Control Law Accelerator

In contrast, the F28004x utilizes a new Type 2 CLA which has the flexibility of accessing eight RAM blocks (LS0 – LS7 RAM). Each RAM block can be configured as either CLA program memory or CLA data memory, and each RAM block can be configured to be shared between the CPU and CLA. The Type 2 CLA has a 16-bit program counter providing a 64K address memory reach, and each of the eight CLA Tasks can be triggered by up to 256 possible peripheral interrupt sources.

The F28004x new Type 2 CLA features the option of running the lowest priority task (Task 8) as a background task. Once triggered, it runs continuously until it is terminated or reset by the CLA or MCU. The remaining tasks (Task 1 through Task 7, in priority order) can interrupt the background task when they are triggered. If needed, portions of the background task can be made uninterruptible. Some uses of the background task include running continuous functions, such as communications and clean-up routines. The background task is enabled by setting the BGEN bit in the MCTRLBGRND register. The Task 8 interrupt vector is then loaded from the MVECTBGRND register rather than the MVECT8 register. When the background task is interrupted, the branch return address is saved to the MVECTBGRNDACTIVE register, and the address is then loaded back to the MPC when execution continues.

3 Memory

As with the F2806x/03x, the F28004x utilizes a memory map where the unified memory blocks can be accessed in either program space or data space. This type of memory map lends itself well for supporting high-level programming languages. The memory map structure consists of RAM blocks dedicated to the CPU, RAM blocks accessible by the CPU and CLA, RAM blocks accessible by the DMA module, message RAM blocks between the CPU and CLA, flash, and one-time programmable (OTP) memory. The memory configuration of the F28004x differs compared to the F2806x/03x in some of the RAM block naming, location, sizes, and the number of blocks, in addition to the total available device memory as described below.

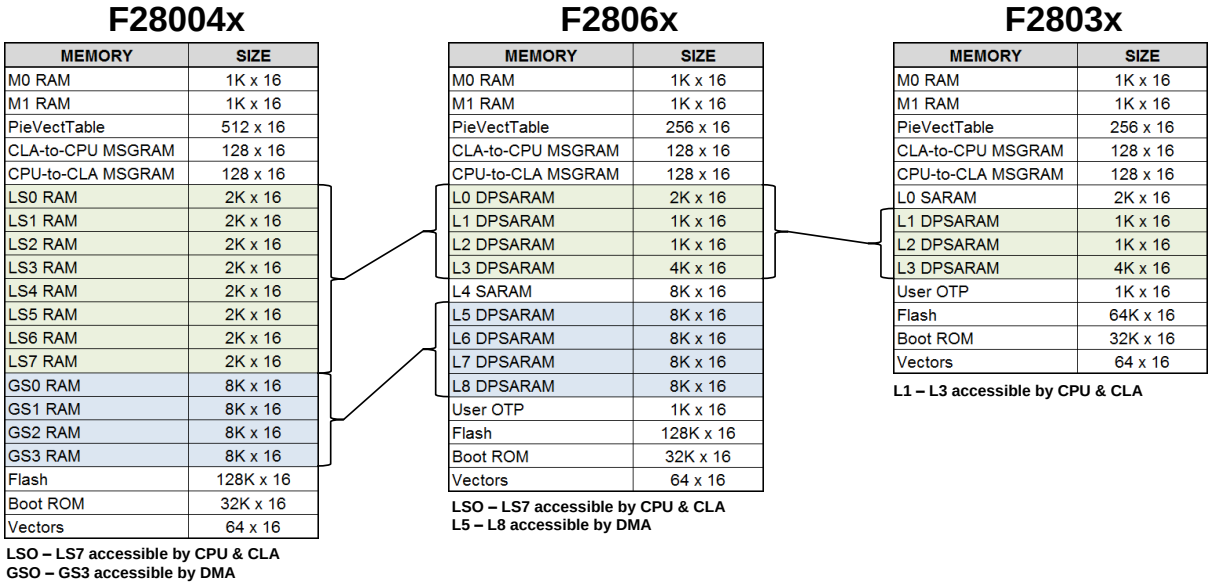

Figure 3. F28004x, F2806x, and F2803x Memory Comparison

Memory blocks M0 and M1 exists on the F2806x/03x and the F28004x at the same location with the same size of 2KB (1Kx16) each. However, the F28004x includes Error-Correcting Code (ECC) protection. These memory blocks are tightly coupled with the CPU, where only the CPU has access to them.

Memory blocks DPSARAM L0 through L3 on the F2806x and L1 through L3 on the F2803x can be mapped to either the CPU or CLA by setting the appropriate bits in the CLA Memory Configuration register. When mapped to the CLA, L3 DPSARAM is configured as program memory with a size of 8KB (4Kx16), and L0 through L2 on the F2806x or L1 and L2 on the F2803x can be individually configured as data memory with a size of 4KB (2Kx16) for L0 and 2KB (1Kx16) for L1 and L2. The F2806x also has an option to allow the CPU to have access to any of the L0 through L2 data memory blocks when mapped to the CLA. The above mentioned DPSARAM blocks can be secured.

Compared to the F2806x/03x, the F28004x has eight local shared RAM blocks named LS0 through LS7 with a size of 4KB (2Kx16) each, and are accessible by the CPU and CLA. The LSx RAM blocks have parity and access protection, and can be secured. By default, the LSx RAM blocks are assigned to the CPU only, however each block can be allocated to the CPU and CLA by configuring the appropriate bit in the LSx Memory Selection register. When allocated to the CLA, each LSx RAM block can then be configured as either CLA program memory or CLA data memory. This provides additional flexibility for allocating program and data memory to the CLA.

The F2806x memory blocks DPSARAM L5 through L8, with a size of 16KB (8Kx16) each, are accessible by the DMA module. Likewise, the F28004x four global shared RAM blocks named GS0 through GS3 have a size of 16KB (8Kx16) each, and are accessible by the DMA module. However, the GSx RAM blocks have parity and access protection.

For the F2806x/03x and F28004x, the CLA message RAM blocks are used to share data between the CPU and CLA. There is a dedicated CLA message RAM block for “CPU to CLA” and another dedicated CLA message RAM block for “CLA to CPU”. By using one-way message RAMs the CLA can read the “CPU to CLA” message RAM while the CPU is writing to it with no arbitration or stalls, and vice versa. The CLA message RAM blocks have a size of 256B (128x16) each, and the F28004x has parity and access protection. The CLA has write access to the “CLA to CPU” message RAM block, and the CPU has write access to the “CPU to CLA” message RAM block. The CPU and CLA have read access to both CLA message RAM blocks.

The F2806x and F2803x contains up to 256KB (128Kx16) and up to 128KB (64Kx16) of flash, respectively. The flash is primarily used to store program code, but can also be used to store static data. Optimal flash performance of code execution is achieved using a 64-bit 2-level deep fetch buffer, which is known as flash pipelining. A code security module (CSM) is used to protect access to the flash, OTP, and selected RAM locations, with the intent to prevent duplication or reverse engineering of proprietary code. The CSM is secured using a 128-bit password which is stored at the end of the flash in the last eight 16-bit word locations. Also, a 2KB (1Kx16) User OTP is available for storing program data or code, and another 2KB (1Kx16) TI OTP is used for calibration data. The flash configuration, method of securing the device memory resources, and locations for the passwords differ for the F28004x, as discussed below.

The F28004x contains 256KB (128Kx16) of flash divided equally into two banks (bank 0 and bank 1). By using two banks, the user can program one bank while operating out of the other bank. Then as needed, the application can switch to the newly programmed bank. The flash sectors have ECC protection providing Single-Error Correction and Double-Error Detection (SECDED), and each sector can be secured. By utilizing a 128-bit code pre-fetch mechanism and data cache, optimal system performance is be achieved. The F28004x flash is optimized for lower cost and lower power consumption.

The F28004x contains two types of OTP memory blocks: TI-OTP and USER-OTP. The TI-OTP contains device specific calibration data for the ADCs, internal oscillators, programmable gain amplifiers, and buffered DACs, in addition to settings used by the flash state machine for erase and program operations. The USER-OTP contains locations for programming security settings, such as passwords for selectively securing memory blocks, configuring the standalone boot process, as well as selecting the boot-mode pins. This information is programmed into the dual code security module (DCSM).

The DCSM offers protection for two zones (zone-1 and zone-2), and like previously discussed, is intended to block access and visibility to the various on-chip memory resources with the purpose of preventing duplication and reverse engineering of proprietary code. The options for both zones are identical, and each memory resource can be assigned to either zone. Either zone can protect each sector of flash individually, and each LSx memory block individually.

Each zone is secured by its own 128-bit (four 32-bit words) user defined CSM password, which is stored in its dedicated OTP location based on a zone-specific link pointer. The user accessible CSMKEY registers are used to secure and unsecure the device, and a new or un-programmed device will be unlocked by default. Each zone’s dedicated OTP block contains its security configuration settings, such as the CSM passwords, and the allocations for securing the LSx RAM blocks and flash sectors. Since the OTP cannot be erased, flexibility is provided by using a link pointer to select the location of the active zone region within the OTP block, allowing the user to make multiple modifications to the configuration up to thirty times. This is accomplished by exploiting the fact that each bit in the OTP can be programmed one bit at a time, and a “1” can be programmed to a “0”, but not erased back to a “1”. The most significant Each zone is secured by its own 128-bit (four 32-bit words) user defined CSM password, which is stored in its dedicated OTP location based on a zone-specific link pointer. The user accessible CSMKEY registers are used to secure and unsecure the device, and a new or un-programmed device will be unlocked by default. Each zone’s dedicated OTP block contains its security configuration settings, such as the CSM passwords, and the allocations for securing the LSx RAM blocks and flash sectors. Since the OTP cannot be erased, flexibility is provided by using a link pointer to select the location of the active zone region within the OTP block, allowing the user to make multiple modifications to the configuration up to thirty times. This is accomplished by exploiting the fact that each bit in the OTP can be programmed one bit at a time, and a “1” can be programmed to a “0”, but not erased back to a “1”.

4 Boot Modes

When the MCU is powered-on, and each time the MCU is reset, the internal bootloader software located in the boot ROM is executed. The boot ROM contains bootloading routines and execution entry points into specific on-chip memory blocks. This initial software program is used to load an application to the device RAM through the various bootable peripherals, or it can be configured to start an application located in flash.

The F2806x/03x boot mode is determined by two RAM locations (EMU_KEY and EMU_BMODE) in the peripheral interrupt expansion block (also known as the PIE block) when emulator is connected, or by two GPIO pins and two OTP memory locations (OTP_KEY and OTP_BMODE) when the emulator is not connected. The F28004x boot mode flow differs and is extremely flexible in its ability to use alternate, reduce, or completely eliminate boot mode selection pins by programming a BOOTPIN_CONFIG register, as discussed below.

After the MCU is powered-up or reset, the PIE block and the master interrupt switch INTM are disabled. This prevents any interrupts during the boot process. The program counter is set to 0x3FFFC0, where the reset vector is fetched. Execution then continues in the boot ROM at the code section named InitBoot. If the emulator is connected, then the boot process follows the Emulation Boot mode flow. In Emulation Boot mode, the boot is determined by the EMU-BOOTPIN-CONFIG and EMU-BOOTDEF-LOW/HIGH registers located in the PIE RAM. If the emulator is not connected, the boot process follows the Stand-alone Boot mode flow. In Stand-alone Boot mode, the boot is determined by two GPIO pins or the Z1-OTP-BOOTPIN-CONFIG and Z1-OTP-BOOTDEF-HIGH/LOW registers located in the DCSM OTP.

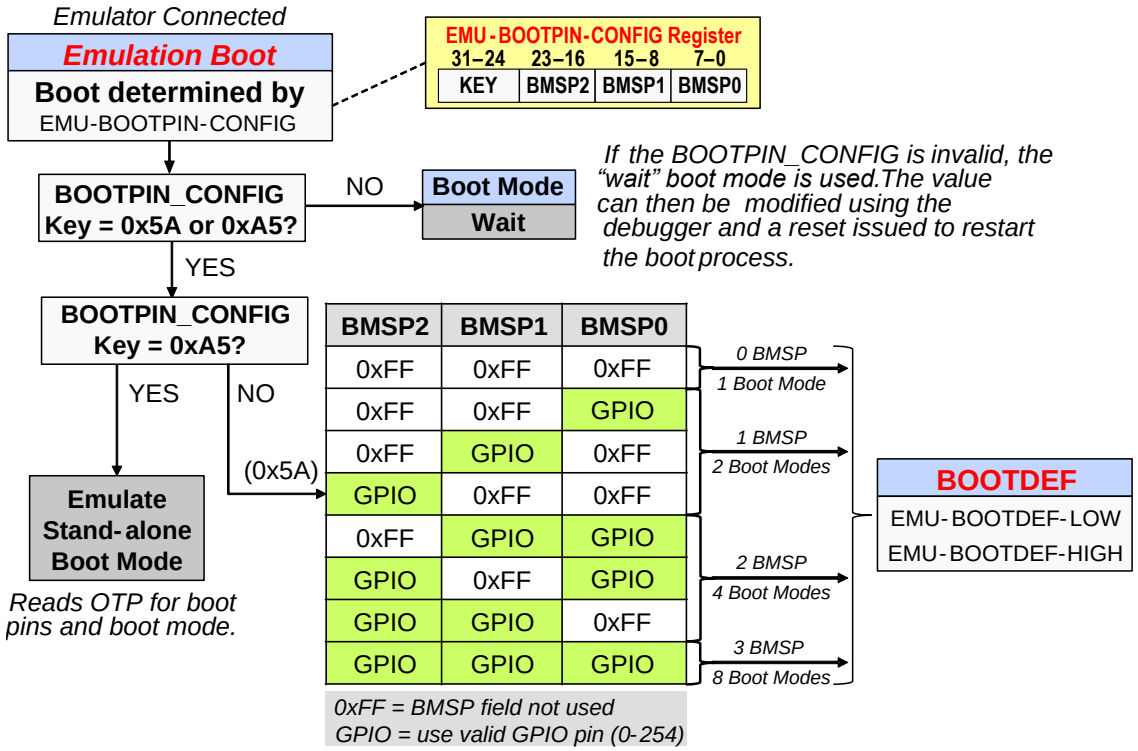

Figure 4. Emulation Boot Mode

In Emulation Boot mode, first the KEY value located in the EMU-BOOTPIN-CONFIG register (bit fields 31-24) is checked for a value of 0x5A or 0xA5. If the KEY value is not 0x5A or 0xA5, the "wait" boot mode is entered. The KEY value and the Boot Mode Selection Pin values (BMSP2-0, bit fields 23-0) can then be modified using the debugger and a reset is issued to restart the boot process. This is the typical sequence followed during device power-up with the emulator connected, allowing the user to control the boot process using the debugger.

Once the EMU-BOOTPIN-CONFIG register is configured and a reset is issued, the KEY value is checked again. If the KEY value is set to 0xA5 the Stand-alone Boot mode is emulated and the Z1-OTP-BOOTPIN-CONFIG register is read for the boot pins and boot mode. Otherwise, the KEY value is set to 0x5A and the boot mode is determined by the BMSP bit field values in the EMU-BOOTPIN-CONFIG register and the EMU-BOOTDEF-LOW/HIGH registers. The EMU-BOOTPIN-CONFIG register contains three BMSP bit fields. If the BMSP bit field is set to 0xFF, then the bit field is not used. Therefore, the boot modes can be set by zero, one, two, or three BMSP bit fields. This provides one, two, four, or eight boot mode options, respectively. Details about the BOOTDEF options will be discussed after the Stand-alone Boot mode is covered.

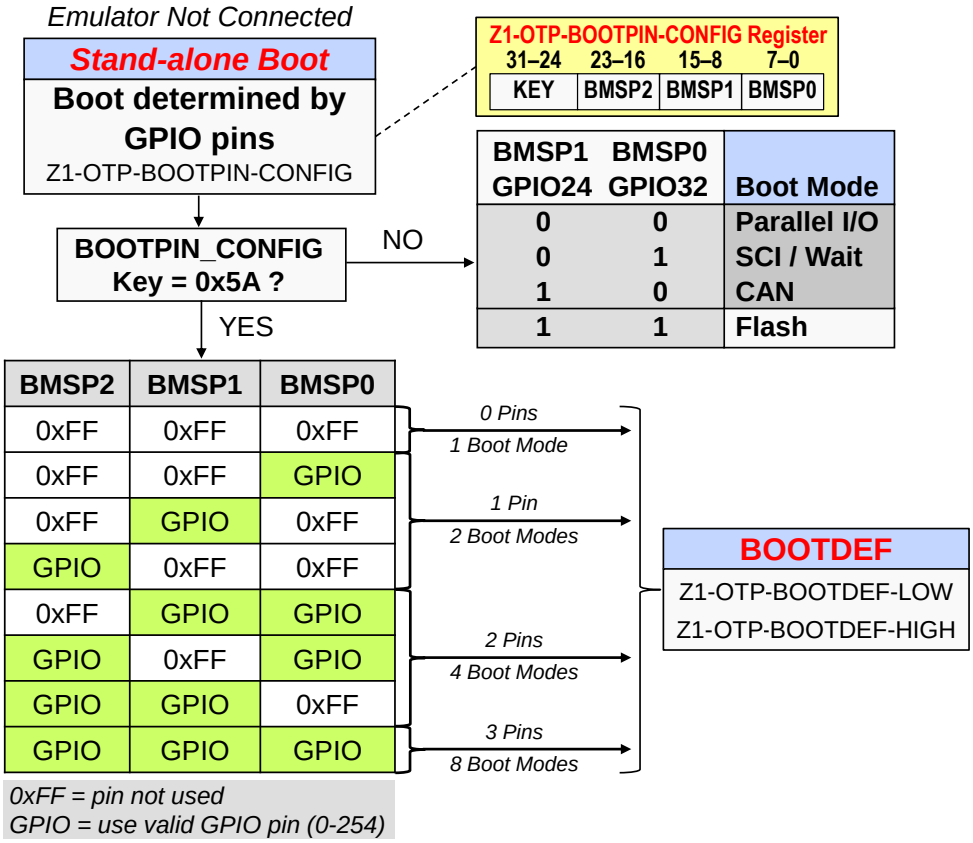

Figure 5. Stand-Alone Boot Mode

In Stand-alone Boot mode, if the KEY value located in the Z1-OTP-BOOTPIN-CONFIG register (bit fields 31-24) is not 0x5A, the boot mode is determined by the default GPIO24 and GPIO32 pins. These two pins provide four boot options: Parallel I/O, SCI/Wait, CAN or flash. If the KEY value is 0x5A the boot mode is determined by the BMSP bit field values in the Z1-OTP-BOOTPIN-CONFIG and the OTP-BOOTDEF-LOW/HIGH registers. The Z1-OTP-BOOTPIN-CONFIG register contains three BMSP bit fields. If the BMSP bit field is set to 0xFF, then the GPIO pin is not used. Therefore, the boot modes can be set by zero, one, two, or three GPIO pins. This provides one, two, four, or eight boot mode options, respectively.

The BOOTDEF options described here applies to both the EMU-BOOTDEF-LOW/HIGH registers used in Emulation Boot mode and the Z1-OTP-BOOTDEF-LOW/HIGH registers used in Stand-alone Boot mode. The BOOTDEF-LOW/HIGH registers consist of eight separate bit fields named BOOT_DEF0 through BOOT-DEF7. These bit fields correspond to the one, two, four, or eight boot mode options that are selected by the zero, one, two, or three BMSP bit fields/GPIO pins, respectively in the BOOTPIN_CONFIG register. Therefore, if zero BMSP bit fields/GPIO pins are selected, then only the BOOT_DEF0 bit field in the BOOTDEF-LOW/HIGH registers is used. Likewise, if three BMSP bit fields/GPIO pins are selected, then BOOT_DEF0 through BOOT_DEF7 in the BOOTDEF-LOW/HIGH registers is used.

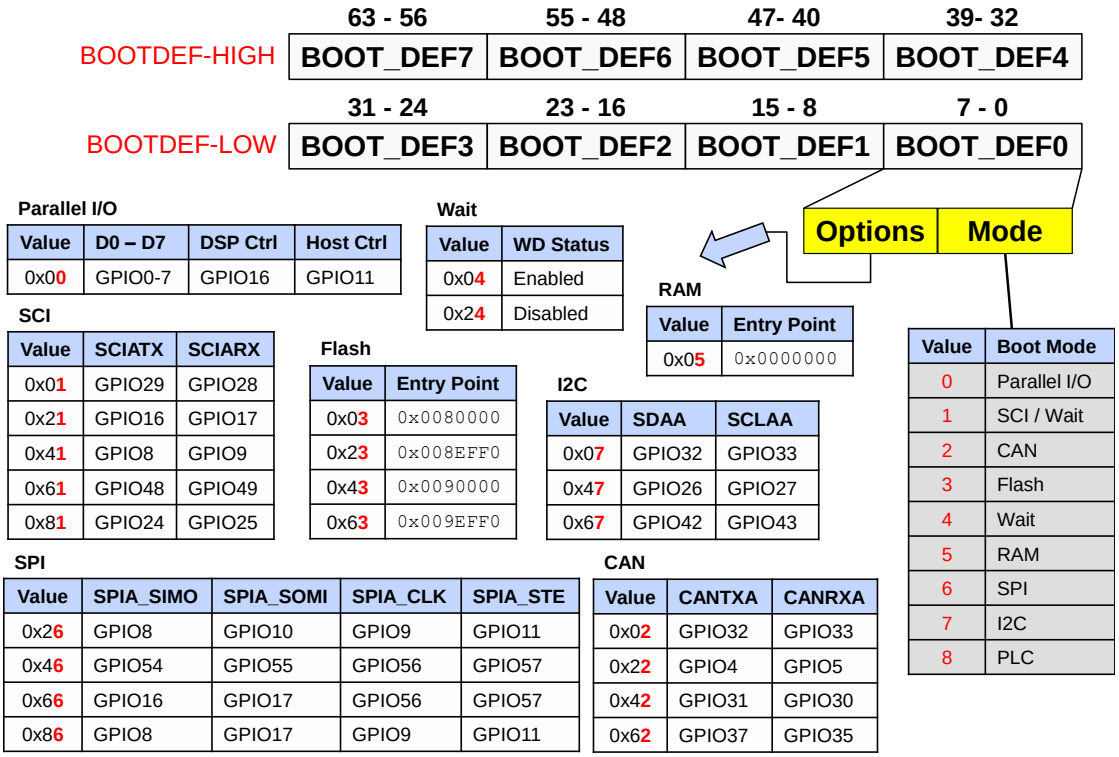

Figure 6. Boot Mode Definitions

The value in the BOOT_DEF bit fields determines which peripheral is used for bootloading or the entry point that is used for code execution. In the BOOT_DEF bit field the lower bits define the boot mode used and the upper bits define the options for that bit mode. Utilizing this type of booting technique provides flexibility for selecting multiple boot modes, as well as reducing the number of boot mode pins.

5 System

The watchdog timer module on the F28004x now includes a WDCLK divider in addition to the watchdog prescaler. Using the WDCLK divider, the WDCLK derived from INTOSC1 can be divided by 2 to 4096 in powers of 2. This, along with the watchdog prescaler, provides a very wide range of timeout values for safety-critical applications. The WDCLK divider defaults to divide by 512 for backwards compatibility. Previously on the F2806x/03x and earlier devices the WDCLK divider was fixed at divide by 512.

The F28004x has three options for supplying the required 1.2 V to the core (VDD): an external supply, the internal 1.2 V low dropout (LDO) voltage regulator (VREG), or the internal 1.2 V switching regulator (DC-DC). The internal DC-DC regulator provides increased efficiency over the LDO for converting 3.3 V to 1.2 V, however using the internal DC-DC regulator requires additional external components. By default the DC-DC regulator is disabled. To use this supply, the MCU must power up initially with the internal LDO VREG and then transition to the internal DC-DC regulator using the application software.

The F28004x interrupt structure PIE module multiplexes up to sixteen peripheral interrupts into each of the twelve CPU interrupt lines, referred to as groups, further expanding support for up to 192 peripheral interrupt signals. This also expands the interrupt vector table, allowing each unique interrupt signal to have its own interrupt service routine (ISR), permitting the CPU to support a large number of peripherals. By comparison, the F2806x/03x multiplexes up to eight peripheral interrupts into each of the twelve groups for up to 96 peripheral interrupt signals.

A series of “lock” registers are incorporated into the F28004x architecture for protecting several system configuration settings from spurious CPU writes. After the lock registers bits are set, the respective locked registers can no longer be modified by software. Once locked, only a module or device reset can restore write capability.

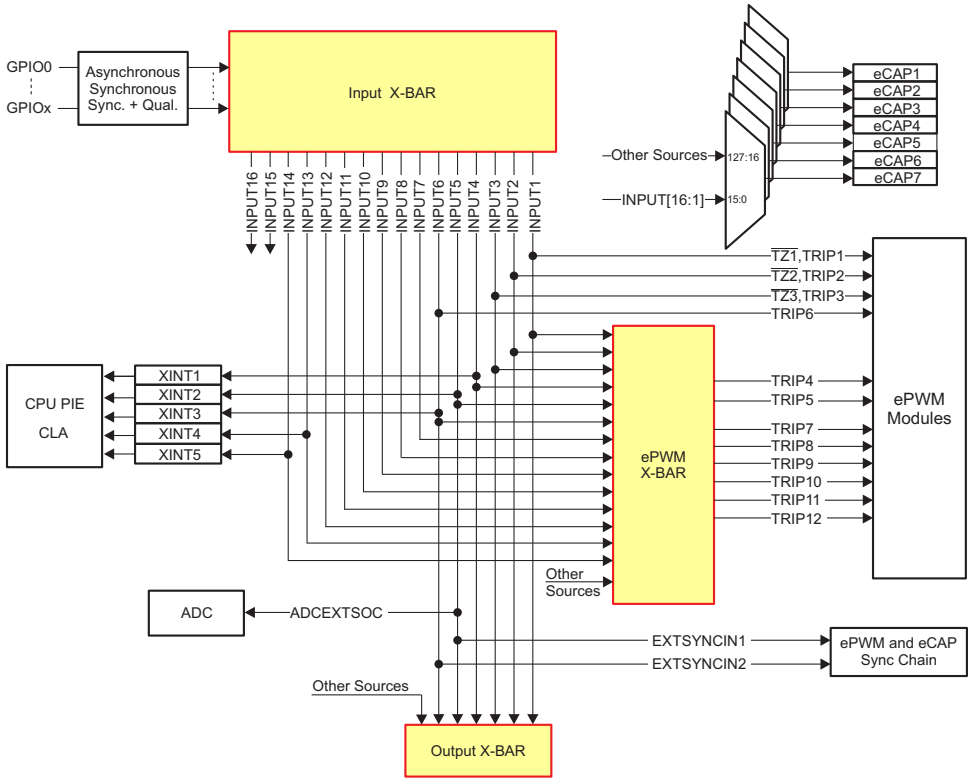

Figure 7. Input, Output, and ePWM Crossbars

The F28004x architecture includes three crossbars (X-BARs): the Input X-BAR, the Output X-BAR, and the ePWM X-BAR. X-BARs provide a flexible means for interconnecting multiple inputs, outputs, and internal resources in various configurations.

The Input X-BAR is used to route external GPIO signals into the device. With access to every GPIO pin, each signal can be routed to any or multiple destinations which include the ADCs, eCAPs, ePWMs, Output X-BAR, and external interrupts. This provides additional flexibility above the multiplexing scheme used by the GPIO structure. Since the GPIO does not affect the Input X-BAR, it is possible to route the output of one peripheral to another, such as measuring the output of an ePWM with an eCAP for frequency testing. The F28004x Input X-BAR has sixteen inputs (INPUT1 through INPUT16) and any of the sixteen inputs can be selected as an input to each of the seven eCAP modules. Whereas, the F2806x/03x architecture uses the GPIO multiplexer to select a dedicated input pin to access the eCAP module.

The Output X-BAR is used to route various internal signals out of the device. It contains eight outputs that are routed to the GPIO structure, where each output has one or multiple assigned pin positions, which are labeled as OUTPUTXBARx. Additionally, the Output X-BAR can select a single signal or logically OR up to 32 signals.

The ePWM X-BAR is used to route signals to the ePWM Digital Compare submodules of each ePWM module for actions such as trip zones and synchronizing. It contains eight outputs that are routed as TRIPx signals to each ePWM module. Likewise, the ePWM X-Bar can select a single signal or logically OR up to 32 signals.

6 Analog Peripherals

Analog peripherals are a critical component in many real-time control systems. They are used to sense signals for the feedback control loops, as well as for hardware protection. By integrating the analog peripherals on-chip, cost and design efforts can be reduced. For this reason, the F28004x incorporates on-chip analog-to-digital converters (ADCs), buffered digital-to-analog converters (DACs), programmable gain amplifiers (PGAs), and comparator subsystems (CMPSS).

Newly introduced on the F28004x is an analog subsystem interconnect which enables a very flexible pin usage, allowing for smaller device packages. The DAC outputs, CMPSS inputs, PGA functions, and digital inputs are multiplexed with the ADC inputs. This type of interconnect permits a single pin to route a signal to multiple analog modules. The analog pins are organized into analog groups around a PGA and CMPSS module, and the routing is defined in an analog pin and internal connections table.

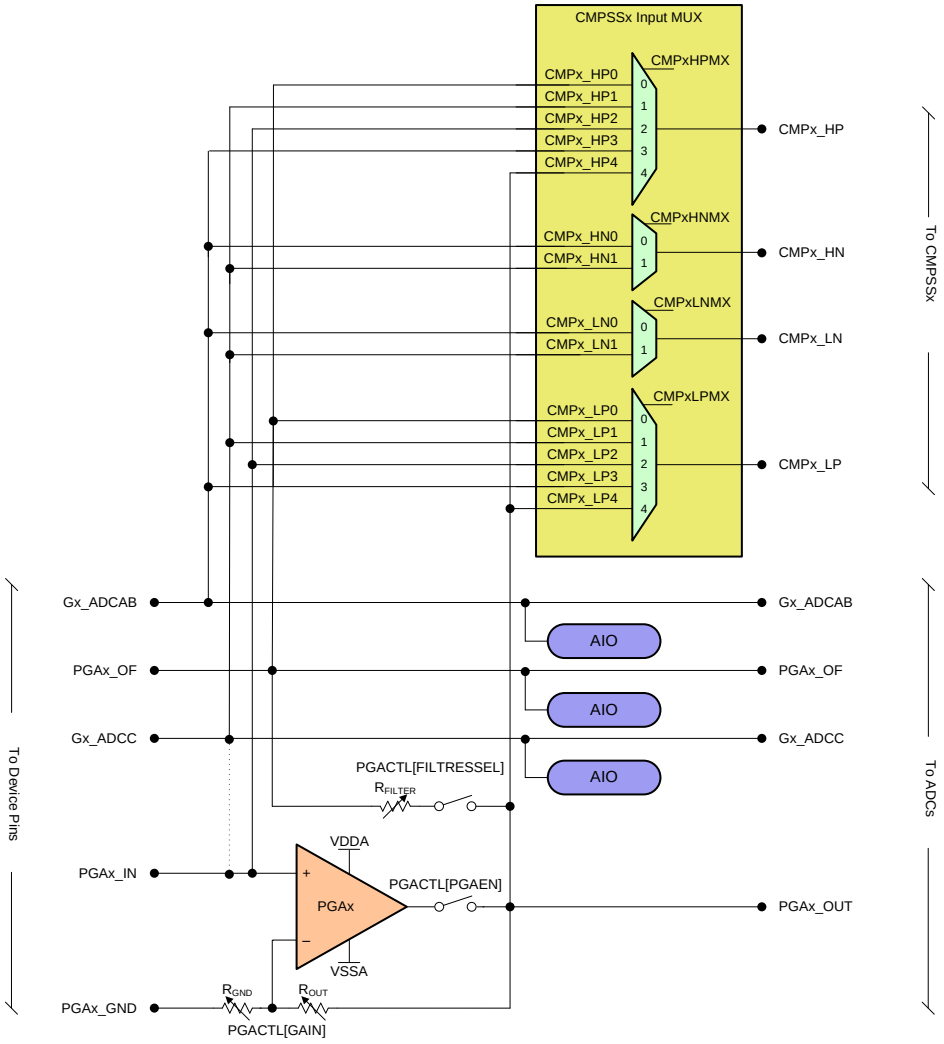

Figure 8. Analog Group Connections

With up to seven PGA modules available on the F28004x, the PGA amplifies small input signals to increase the dynamic range for the downstream ADC and CMPSS modules. The PGA has four programmable gain modes (3x, 6x, 12x, and 24x) and support for low-pass filtering using an external capacitor, making it adaptable to various performance needs. The filtered signal is available for sampling and monitoring by the ADC and CMPSS modules.

The F28004x CMPSS provides substantial enhancements over the analog comparator circuits found on the F2806x/03x. The CMPSS modules are useful for implementing peak current mode control and voltage trip monitoring, which are used in applications such as switched-mode power control and power factor correction. The module is designed around a pair of analog comparators which generates a digital output indicating if the voltage on the positive input is greater than the voltage on the negative input. The positive input to the comparator is always driven from an external pin. The negative input can be driven by either an external pin or an internal programmable 12-bit digital-to-analog (DAC) as a reference voltage. Values written to the DAC can take effect immediately or be synchronized with ePWM events. A falling-ramp generator is available to control the internal DAC reference value for one comparator in the module, which enables peak current mode control in digital power applications. Each comparator output is fed through a programmable digital filter to prevent electrical switching noise from causing spurious trip signals. The output of the CMPSS generates trip signals to the ePWM event trigger submodule and GPIO structure.

Up to seven Type 1 CMPSS modules are available on the F28004x. The Type 1 CMPSS features PWM blanking capability to clear-and-reset existing or imminent trip conditions near the ePWM cycle boundaries. Also, by using the analog subsystem interconnect scheme the CMPSS comparator positive and negative input signals are independently selectable.

Unlike the ADC found on the F2806x/03x where a single ADC has two sample-and-hold (S/H) circuits, the F28004x utilizes three independent ADCs and each has a single S/H circuit. This allows the F28004x to efficiently manage multiple analog signals for enhanced overall system throughput. By using multiple ADC modules, simultaneous sampling or independent operation can be achieved. The ADC module is implemented using a successive approximation type ADC with a resolution of 12-bits and it provides performance of 3.45 MSPS, yielding 10.35 MSPS for the device.

Like the F2806x/03x, ADC triggering and conversion sequencing is managed by a series of start-of-conversion (SOCx) configuration registers. Also, the F28004x adds burst priority mode, in addition to the round robin and high priority modes. Burst mode allows a single trigger to convert one or more than one SOCx sequentially at a time. This mode uses a separate Burst Control register to select the burst size and trigger source.

To further enhance the capabilities of the F28004x ADC, each ADC module incorporates four post-processing blocks (PPB), and each PPB can be linked to any of the ADC result registers. The PPBs can be used for offset correction, calculating an error from a set-point, detecting a limit and zero-crossing, and capturing a trigger-to-sample delay. Offset correction can simultaneously remove an offset associated with an ADCIN channel that was possibly caused by external sensors or signal sources with zero-overhead, thereby saving processor cycles. Error calculation can automatically subtract out a computed error from a set-point or expected result register value, reducing the sample to output latency and software overhead. Limit and zero-crossing detection automatically performs a check against a high/low limit or zero-crossing and can generate a trip to the ePWM and/or generate an interrupt. This lowers the sample to ePWM latency and reduces software overhead. Also, it can trip the ePWM based on an out-of-range ADC conversion without any CPU intervention which is useful for safety conscious applications. Sample delay capture records the delay between when the SOCx is triggered and when it begins to be sampled. This can enable software techniques to be used for reducing the delay error.

The buffered 12-bit DAC module can be used to provide a programmable reference output voltage and it includes an analog output buffer that is capable of driving an external load. The F28004x has two Type 1 DAC modules which include 1x and 2x gain mode options. Values written to the DAC can take effect immediately or be synchronized with ePWM events. Also, the DAC internal reference supports both 2.5 volt and 3.3 volt options.

7 Control Peripherals

The enhanced Pulse Width Modulator (ePWM) module is a fundamental building block for most digital control systems. Power switching devices can be difficult to control when operating in the proportional region, but are easy to control in the saturation and cutoff regions. Since PWM is a digital signal by nature and easy for an MCU to generate, it is ideal for use with power switching devices. Essentially, PWM performs a DAC function, where the duty cycle is equivalent to the DAC analog amplitude value. The F28004x uses Type 4 ePWM modules, which provides enhanced capabilities over the Type 1 ePWM modules found on the F2806x/03x. Though the ePWM module remains functionally the same, the enhanced capabilities resulted in additional registers and the ePWM address space has been remapped for better alignment and usage.

Some key enhancements to the ePWM module include the addition of two more compare registers (CMPC and CMPD) to the Counter Compare sub-module. This allows additional interrupts and ADC SOC events to be generated, as well as other PWM SYNC related functions, such as PWM SYNCOUT generation and PWMSYNC pulse selection. The additional compare registers are shadowed to allow for dynamic configuration changes. The Action Qualifier sub-module includes two additional trigger events (T1 and T2) to support delayed trip functionality for peak current mode control type applications. Registers AQCTLA and AQCTLB are shadowed to enable changes that must occur at the end of a period even when the phase changes. The Dead-Band sub-module counters are increased to 14-bits, the RED and FED can be enabled on either PWM outputs, and the DBCTL register is shadowed. The Digital Compare sub-module supports up to twelve external trip sources which are selected by the Input X-BAR, in addition to logically OR all of them with up to 14 external and internal sources to create the respective digital compare event trip. The ePWM module allows for one shot and global reload capability from shadow to active registers to avoid partial reload conditions.

Eight ePWM modules are available on the F28004x. The ePWM modules are highly programmable, extremely flexible, and easy to use, while being capable of generating complex pulse width waveforms with minimal CPU overhead or intervention.

The enhanced Capture (eCAP) module is used to accurately time external events by timestamping transitions on the capture input pin. It can be used to measure the speed of a rotating machine, determine the elapsed time between pulses, calculate the period and duty cycle of a pulse train signal, and decode current/voltage measurements derived from duty cycle encoded current/voltage sensors. The F28004x has seven Type 1 eCAP modules, which provides enhanced capabilities over the Type 0 eCAP modules found on the F2806x/03x. This new eCAP module features high-resolution capture capability which is now an extension to the module, rather than being a separate module. With continuous hardware calibration, complex software intensive based calibration routines are not needed. Each eCAP module has a 128-to-1 multiplexer which is used to select a variety of input signals. External signals are routed first through the Input X-BAR and any GPIO pin can be used as an input. An event filter reset bit is now available which clears the event filter, modulo counter, and any pending interrupt flags (useful for initialization and debug), in addition to a modulo counter status bit for reading the current state of the modulo counter. Also, any one of the four available capture event interrupt triggers can be used as a trigger source for the DMA, and EALLOW protection has been added to eCAP critical registers.

The enhanced Quadrature Encoder Pulse (eQEP) module interfaces with a linear or rotary incremental encoder for determining position, direction, and speed information from a rotating machine which is typically found in high-performance motion and position-control systems. The F28004x has a new Type 1 eQEP that includes a QEP Mode Adapter (QMA). The QMA evaluates transitions on the external EQEPA and EQEPB signal lines and generates direction and clock signals for supporting industrial drive applications. To use the QMA the eQEP module needs to be configured in the Direction-Count mode. The F28004x has up to two eQEP modules. The F2806x/03x uses a Type 0 eQEP.

The sigma delta filter module (SDFM) is a four-channel digital filter designed to sense analog signals, such as shunt currents and voltages, for digital power and motor control applications. Each channel can receive an independent delta-sigma modulator bit stream which is processed individually by programmable digital decimation filters. The filters include fast comparators for immediate digital threshold comparisons for over-current and under-current monitoring. A key benefit of the SDFM is that its use in a system enables a simple and cost-effective high-voltage isolation boundary. The F28004x has a new Type 1 SDFM which adds 16-deep by 32-bit FIFOs to all data filters, and each data filter has its own data ready peripheral interrupt.

8 Communication Peripherals

The F28004x adds a new Type 0 Power Management Bus (PMBus) and a new Type 0 Fast Serial Interface (FSI). The PMBus module provides an interface for communicating between the MCU and other devices that are compliant with the System Management Interface (SMI) specification. PMBus is an open-standard digital power management protocol that enables communication between components of a power system. The FSI module is a highly reliable high-speed serial communication peripheral capable of operating at dual data rate providing 100 Mbps transfer using a 50 MHz clock. The FSI consists of independent transmitter and receiver cores that are configured and operated independently. FSI is a point-to-point communication protocol operating in a single-master/single-slave configuration. With this high-speed data rate and low channel count, the FSI can be used to increase the amount of information transmitted and reduce the costs to communicate over an isolation barrier.

9 Embedded Real-Time Analysis and Diagnostic

The embedded real-time analysis and diagnostic (ERAD) module extends the debug and system analysis capabilities of the F28004x. A standard C28x CPU includes two analysis units, where the first analysis unit counts events or monitors address buses, and the second analysis unit monitors address and data buses. The two analysis units can be configured for hardware breakpoints or hardware watch points, and additionally the first analysis unit can be configured as a benchmark counter or event counter. The ERAD module is external to the C28x CPU and further expands this capability to provide additional hardware breakpoints, hardware watch points, and counters for profiling, as well as other advanced features.

The F28004x ERAD module contains eight enhanced bus comparator units and four benchmark system event counter units. The bus comparator units are used to generate hardware breakpoints, hardware watch points, and other output events. The event counter units are used to analyze and profile the system.

Each bus comparator unit is connected to the CPU and information from the address buses, data buses, and program counter is used as inputs for comparison. The resulting events generated by the bus comparator unit can be used as debug triggers to the CPU, and may be fed to the event counter unit for additional system profiling. The event counter unit can use the output of the bus comparator unit and external system events, such as PIE interrupts, timer interrupts, and CLA task interrupts, for controlling the counter to provide benchmark and profiling analysis. The event counter unit can also generate debug triggers to the CPU.

Typically, the ERAD module is used by the debug software. However, the user application software can also be configured to use the ERAD module. This is especially useful for real-time systems where it is not possible to connect a debug probe for intrusive debug. In this case, the user software can control the ERAD module for non-intrusive debug and profiling of the system.

10 Summary

The F28004x MCU device family offers a balance of high-performance and cost efficiency. Based on the high-performance TI C28x 32-bit CPU along with its tightly coupled accelerators, advanced control peripherals, and integrated analog functions, the F28004x is extremely capable of solving today's demanding complex real-time control systems and signal processing applications. The enhanced boot modes offer the ability to use alternate, reduce, or completely eliminate boot mode selection pins, while the new analog subsystem interconnect enables a very flexible pin usage for smaller device packages. The other F28004x architectural enhancements discussed in this document were carefully implemented to achieve the goal of optimizing the two key parameters of performance and cost. Table 1 provides a simplified general feature comparison between the F28004x, F2806x, and F2803x MCU device families.

Table 1. Device Matrix

	F28004x	F2806x	F2803x
Clock	100 MHz	90 MHz	60 MHz
FPU	√ (1)	√	—
TMU	√	—	—
VCU-I	√	√	—
CLA	√ (Type 2)	√ (Type 0)	√ (Type 0)
6-Channel DMA	√	√	—
Flash / RAM	128Kx16 / 50Kx16	128Kx16 / 50Kx16	64Kx16 / 10Kx16
32-bit CPU Timers	√	√	√
Watchdog Timer	√	√	√
On-chip Oscillators	√	√	√
ADC	Three 12-bit	One 12-bit	One 12-bit
CMP w/DAC	√ (CMPSS)	√	√
Buffered DAC	√	—	—
PGA	√	—	—
ePWM / HRPWM	√ / √ (Type 4)	√ / √ (Type 1)	√ / √ (Type 1)
eCAP / HRCAP	√ / √ (Type 1)	√ / √ (Type 0)	√ / √ (Type 0)
eQEP	√ (Type 1)	√ (Type 0)	√ (Type 0)
SDFM	√ (Type 1)	—	—
CAN (2)	√ (DCAN)	√ (eCAN)	√ (eCAN)
I2C	√ (Type 1)	√ (Type 0)	√ (Type 0)
McBSP	—	√	—
SCI	√	√	√
SPI	√ (Type 2)	√ (Type 1)	√ (Type 1)
LIN	√ (Type 1)	—	√ (Type 0)
USB	—	√	—
PMBus	√	—	—
FSI	√	—	—
ERAD	√	—	—

(1) Indicates available – check specific device for details.

(2) DCAN and eCAN are not software compatible.

11 References

For more details, see the F28004x documentation listed below:

- *TMS320F28004x Piccolo Microcontrollers Technical Reference Manual*
- *TMS320F28004x Piccolo™ Microcontrollers Data Manual*

Also, additional information about the C2000 accelerators can be found in the following document:

- *Accelerators: Enhancing the Capabilities of the C2000™ MCU Family*

Revision History

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

Changes from Original (June 2017) to A Revision	Page
• Added new Section 9.	14
• Update was made toTable 1.	15

IMPORTANT NOTICE FOR TI DESIGN INFORMATION AND RESOURCES

Texas Instruments Incorporated ('TI") technical, application or other design advice, services or information, including, but not limited to, reference designs and materials relating to evaluation modules, (collectively, "TI Resources") are intended to assist designers who are developing applications that incorporate TI products; by downloading, accessing or using any particular TI Resource in any way, you (individually or, if you are acting on behalf of a company, your company) agree to use it solely for this purpose and subject to the terms of this Notice.

TI's provision of TI Resources does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such TI Resources. TI reserves the right to make corrections, enhancements, improvements and other changes to its TI Resources.

You understand and agree that you remain responsible for using your independent analysis, evaluation and judgment in designing your applications and that you have full and exclusive responsibility to assure the safety of your applications and compliance of your applications (and of all TI products used in or for your applications) with all applicable regulations, laws and other applicable requirements. You represent that, with respect to your applications, you have all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. You agree that prior to using or distributing any applications that include TI products, you will thoroughly test such applications and the functionality of such TI products as used in such applications. TI has not conducted any testing other than that specifically described in the published documentation for a particular TI Resource.

You are authorized to use, copy and modify any individual TI Resource only in connection with the development of applications that include the TI product(s) identified in such TI Resource. NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information regarding or referencing third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of TI Resources may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI RESOURCES ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING TI RESOURCES OR USE THEREOF, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY YOU AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS EVEN IF DESCRIBED IN TI RESOURCES OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF TI RESOURCES OR USE THEREOF, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You agree to fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of your non-compliance with the terms and provisions of this Notice.

This Notice applies to TI Resources. Additional terms apply to the use and purchase of certain types of materials, TI products and services. These include; without limitation, TI's standard terms for semiconductor products http://www.ti.com/sc/docs/stdterms.htm), evaluation modules, and samples (http://www.ti.com/sc/docs/sampterms.htm).

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265