Application Report
SPRACK2A–September 2019–Revised March 2020

Migration Between TMS320F28004x and TMS320F28002x

Joseph Casuga, Frank Ankapong, and Jason Whiles

ABSTRACT

This migration guide describes the hardware and software differences to be aware of when moving between F28004x and F28002x C2000™ MCUs. This document shows the block diagram between the two MCUs as a visual representation on what blocks are similar or different. It also highlights the features that are unique between the two devices for all available packages in a device comparison table. A good starting point for migration would be the 64-PM package, which is pinout compatible between F28004x and F28002x, so a 64-PM PCB hardware section has been added. Section 2 discusses hardware considerations when migrating between F28004x and F28002x with the 64-PM package. The digital general-purpose input/output (GPIO) and analog multiplex comparison tables show pin functionality between the two MCUs. This is a good reference for hardware design and signal routing when considering a move between the two devices. Lastly, the F28002x software support is only in EABI format. The EABI migration is discussed in Section 4.

Contents

List of Figures

List of Tables

Trademarks

C2000, Code Composer Studio are trademarks of Texas Instruments.
All other trademarks are the property of their respective owners.

1 Feature Differences Between F28004x and F28002x

F28002x is a subset of F28004x with a few new features. They have one package in common, 64-pin PM. It is possible to migrate between F28002x and F28004x with the caveats in this document taken into account.

NOTE: This comparison guide focuses on the super-set devices: F280049 and F280025. Other part numbers in this product family have reduced feature support. For details specific to part numbers, see the device-specific data sheet.

1.1 F28004x and F28002x Feature Comparison

An overlaid block diagram of F28004x and F28002x is shown in Figure 1 while feature comparison of the superset part numbers for the F28002x and F28004x devices is shown in Table 1.

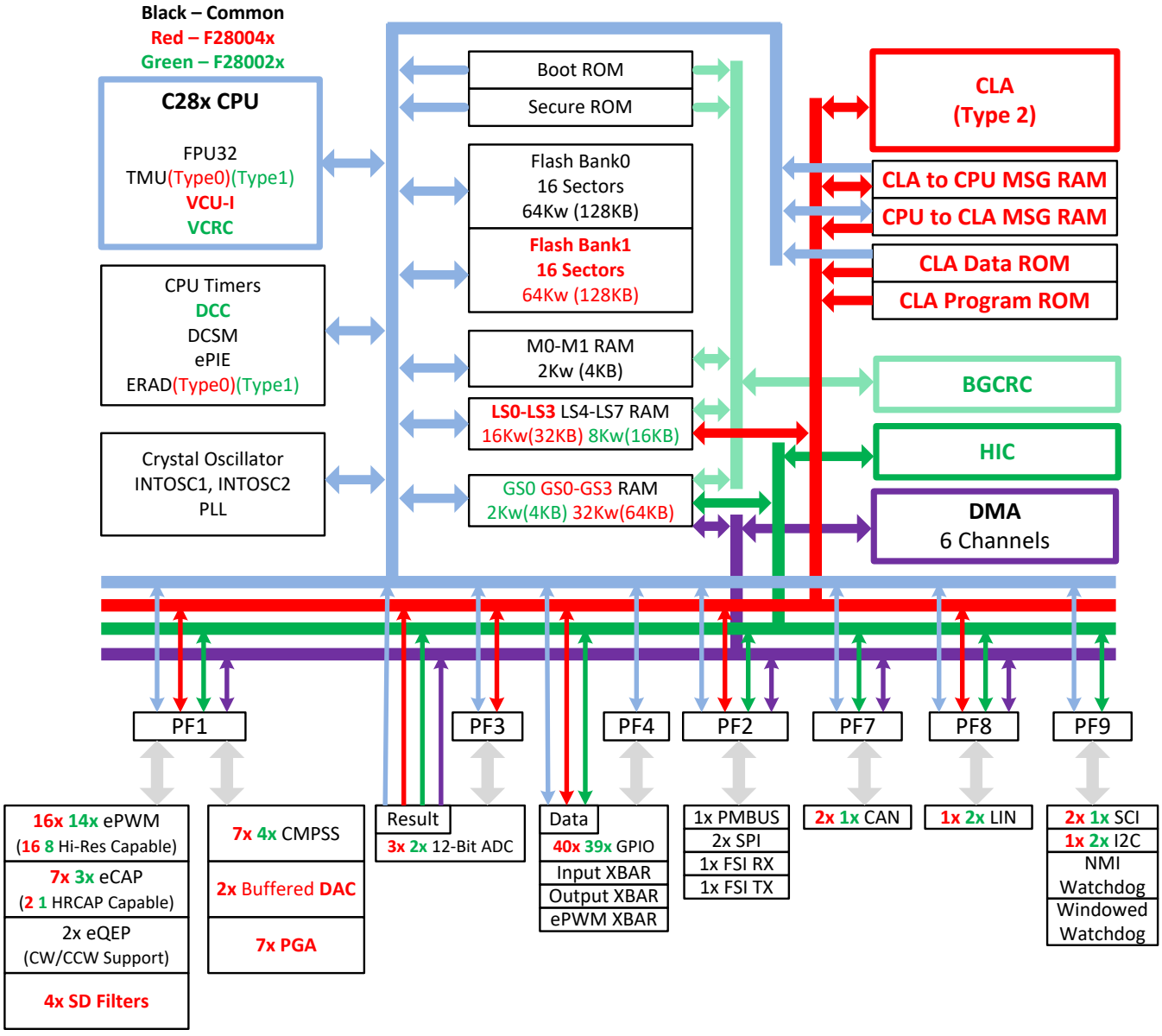

Figure 1. F28002x and F28004x Overlaid Functional Block Diagram

Table 1. F28004x and F28002x Superset Device Comparison

Feature		F280049 F280049C			F280025 F280025C		
		100-Pin PZ	64-Pin PM	56-Pin RSH	80-Pin PN	64-Pin PM	48-Pin PT
Processor and Accelerators							
C28x	Frequency (MHz)	100			100		
	FPU	Yes			Yes (with new instructions for Fast Integer Division)		
	VCU-I	Yes			No		
	VCRC	No			Yes		
	TMU	Yes - Type 0			Yes - Type 1 (with new instructions supporting NLPID)		
CLA – Type 2	Available	Yes			No		
	Frequency (MHz)	100			–		
6-Channel DMA – Type 0		Yes			Yes		
Memory							
Flash		256KB (128Kw)			128KB (64Kw)		
RAM	Dedicated and Local Shared RAM	36KB (18Kw)			20KB (10Kw)		
	Global Shared RAM	64KB (32Kw)			4KB (2Kw)		
	Total RAM	100KB (50Kw)			24KB (12Kw)		
Code security for on-chip flash and RAM		Yes			Yes		
System							
Configurable Logic Block (CLB)		F28004xC			F28002xC		
Motor Control Libraries in ROM		F28004xC			F28002xC		
32-bit CPU timers		3			3		
Watchdog timers		1			1		
Nonmaskable Interrupt Watchdog (NMIWD) timers		1			1		
Crystal oscillator/External clock input		1			1		
0-pin internal oscillator		2			2		
GPIO pins		40	26	25	39	26	14
Additional GPIO		3 (If 2-pin cJTAG is used and INTOSC used instead of X1)			4 (If 2-pin cJTAG is used and INTOSC used instead of X1/X2)		
AIO inputs		21	14	12	16		14
External interrupts		5			5		
Analog Peripherals							
ADC 12-bit	Number of ADCs	3			2		
	MSPS	3.45			3.45		
	Conversion Time (ns)	300			300		
ADC channels (single-ended)		21	14	12	16		14
Temperature sensor		1			1		
Buffered DAC		2			–		
CMPSS (each CMPSS has two comparators and two internal DACs)		7	6	5	4		
PGA (Gain Settings: 3, 6, 12, 24)		7	5	4	–		
Control Peripherals							
eCAP/HRCAP modules – F28004x: Type 1 F28002x: Type 2		7 (2 with HRCAP capability)			3 (1 with HRCAP capability)		
ePWM/HRPWM channels – Type 4		16 (16 with HRPWM capability)			14 (8 with HRPWM capability)		
eQEP modules – F28004x: Type 1 F28002x: Type 2		2	1		2		
SDFM channels – Type 1		4	3		–		

Table 1. F28004x and F28002x Superset Device Comparison (continued)

Feature	F280049 F280049C			F280025 F280025C		
	100-Pin PZ	64-Pin PM	56-Pin RSH	80-Pin PN	64-Pin PM	48-Pin PT
Communication Peripherals						
CAN – Type 0	2			1		
I2C – Type 1	1			2		
SCI – Type 0	2			1		
SPI – Type 2	2			2		
LIN – Type 1	1			2		
PMBus – Type 0	1			1		
FSI – Type 0	1 (1 RX and 1 TX)			1 (1 RX and 1 TX)		
Package Options, Temperature, and Qualification						
S: –40°C to 125°C (TJ)	Yes			Yes		
Q: –40°C to 125°C (TA) (AEC Q100 qualification)	Yes	No (use F280048 or F280040)	–	Yes	No (use F280024 or F280022)	Yes

2 PCB Hardware Changes for the 64-Pin PM Package

This section describes some considerations to note when switching boards between F28004x and F28002x devices, or designing boards for the 64-pin package option on either device.

2.1 Internal DC-DC Regulator

F28002x does not have the internal DC-DC regulator. Due to this, GPIO22 and GPIO23 do not have the alternate functions of VFBSW and VSW, respectively. Also, VDDIO_SW is just another VDDIO and VSS_SW is just another VSS.

If using a F28002x device in a F28004x board, DC-DC is not available so VFBSW (GPIO22) and VSW (GPIO23) can be used as regular GPIOs.

If using a F28004x device in a F28002x board, DC-DC is available so GPIO22 and GPIO23 can be used as the alternate functions VFBSW and VSW, respectively, for the DC-DC. However, due to how sensitive the DC-DC is to board layout, if the F28002x board was not built with a DC-DC layout in mind, it should not be used as such.

2.2 VREGENZ Pin

F28002x does not have a VREGENZ pin. The VREGENZ pin is converted to a GPIO (GPIO39) and the VREGENZ signal is internally tied low. Due to this, the internal VREG is always enabled and hence supplying the 1.2 V externally is not an option.

If using a F28002x device in a F28004x board, the VREGENZ pin can be used as GPIO39 or you can follow the unused pins practice for GPIOs found in the device-specific data sheet.

If using a F28004x device in a F28002x board, tie GPIO39 to VDDIO to supply the 1.2 V externally or tie GPIO39 to VSS to use the internal VREG.

2.3 XTAL X1 GPIO Mux Capability

On the F28002x device, the crystal pin X1 also doubles as a GPIO (GPIO19).

If using a F28002x device in a F28004x board, the X1 pin can be used as GPIO19 after making the required software changes. Remove the load cap on X1 if one exists and a crystal is not being used as the clock source.

If using a F28004x device in a F28002x board, install the load cap on X1 if using a crystal as the clock source. GPIO19 functionality is not available.

2.4 PGA Ground

The F28002x device does not have PGAs. The two PGA_GND pins on the 64-pin package are converted to ADC channels. PGA135_GND is now A15/C7/AIO233, and PGA246_GND is now A8/C11/AIO241.

If using a F28002x device in a F28004x board:

- If PGA135_GND and PGA246_GND pins are not tied to GND, they can be used as the ADC channels.
- If PGA135_GND and PGA246_GND pins are tied to GND, those two ADC channels will read a low conversion.

If using a F28004x device in a F28002x board, tie A15/C7/AIO233 and A8/C11/AIO241 to VSSA.

2.5 Analog Pin Assignment

For more information, see Section 3.10.

2.6 GPIO Pin Assignment

For more information, see Section 3.9.

3 Feature Differences for System Consideration

This section outlines the differences and similarities that exist when moving between the F28002x and F28004x devices.

3.1 New Features in F28002x

This section outlines features that only exist in the F28002x device.

3.1.1 TMU Type1

Two instructions have been added to the instruction set of the Trigonometrical Math Unit (TMU) on F28002x to support computation of the floating-point power function “powf”. These instructions calculate the inverse binary exponent in the base two logarithm and can be combined to compute the power of a floating-point number raised to the power of another floating-point number. This calculation would typically take 300 cycles using library emulation, but takes less than ten cycles using the new instructions. An example of the application of the power function is non-linear proportional integral derivative control (NLPID), which is a component of the C2000 Digital Control Library.

3.1.2 Fast Integer Division (FINTDIV)

The C28x processor Fast Integer Division (FINTDIV) unit provides an open and scalable approach to facilitate different data type sizes (16/16, 32/16, 32/32, 64/32, 64/64), signed and unsigned or mixed data type versions (ui32/ui32, i32/ui32, i32/i32) and for additional performance, the operations return both the integer and remainder portion of the calculation simultaneously. The division operations are interruptible so as to enable minimum latency for higher priority tasks, a critical requirement for high performance real-time control applications. Unique to this fast integer division unit is support for Truncated, Modulo and Euclidean division formats without any cycle penalty. Each of these formats represents the integer and remainder result in different forms. Below is a brief summary of the various division formats:

- Truncated format is the traditional division performed in C language (/ = integer, % = remainder), however, the integer value is non-linear around zero.
- Modulo division is commonly found when performing division on an Excel worksheet.
- Euclidean format is another format similar to Modulo, the difference is the sign on the remainder value.

Both the Euclidean and Modulo formats are more appropriate for precise control applications because the integer value is linear around the zero point and, hence, avoid potential calculation hysteresis. The C28x compiler supports all three division formats for all data types.

3.1.3 Host Interface Controller (HIC)

The Host Interface Controller (HIC) is a new module for the F28002x device that allows an external host controller to directly access resources of the F28002x device using the ASRAM protocol. For details on the HIC, see the *TMS320F28002x microcontrollers technical reference manual* (SPRUIN7).

3.1.4 Background CRC (BGCRC)

The Background CRC (BGCRC) is a new module for the F28002x device that can compute the CRC-32 value of a configurable block of memory. It is an upgrade on the CLAPROMCRC found in the F28004x device to test more memories than just the CLA ROM. Note that the F28002x device does not have a CLA. BGCRC was first introduced in F2838x. For details on the BGCRC, see the *TMS320F28002x microcontrollers technical reference manual* (SPRUIN7) or the *TMS320F2838x microcontrollers technical reference manual* (SPRUII0).

3.1.5 Standby Low Power Mode

In F28004x, standby low power mode was deprecated. In F28002x, standby low power mode is available for applications requiring this power saving feature.

3.1.6 X1 GPIO Functionality

In the F28002x device, the crystal pin X1 can also be used as a GPIO, GPIO19. This is a new feature for the F28002x. The X1 pin can either be used as the crystal input pin or as GPIO19, but not both simultaneously. For details on this functionality, see the *TMS320F28002x microcontrollers technical reference manual* (SPRUIN7).

3.1.7 Diagnostic Features (PBIST/HWBIST)

F28004x documents refer to PBIST as the controller that executes configurable memory tests routines as part of the boot up sequence. In F28002x documents and in future C2000 device documents, this module is referred to as MPOST (memory power on self-test). PBIST (MPOST) is enabled as part of the boot up sequence in both F28004x and F28002x devices. HWBIST is a self-test controller for the CPU for fault coverage in safety applications. HWBIST can be invoked from user application code. HWBIST is available only in the F28002x device.

3.2 Communication Module Changes

The communication module changes between the F28004x and F28002x devices only affect the number of peripherals. Module functionality is maintained for both devices. Table 2 shows the module instances differences which should be considered when migrating applications between F28004x and F28002x.

Table 2. Communication Module Instances

Module	F28004x	F28002x
LIN	1 - LINA	2 - LINA, LINB
CAN	2 - CANA, CANB	1 - CANA
SCI	2 - SCIA, SCIB	1 - SCIA
SPI	2 - SPIA, SPIB	2 - SPIA SPIB
I2C	1 - I2CA	2 - I2CA, I2CB
PMBUS	1 - PMBUSA	1 - PMBUSA
FSI	1 - FSIA	1 - FSIA

3.3 Control Module Changes

There are changes in the control modules between the F28004x and F28002x devices. The biggest change comes from the EPWM on the F28002x device which has a new generic and simple sync scheme that allows any EPWM/ECAP to be the master sync source for another EPWM/ECAP. Table 3 shows the module instances differences which should be considered when migrating applications between F28004x and F28002x.

Table 3. Control Module Differences

Module	Category	F28004x	F28002x	Note
SDFM	Number	4 - SD1_C1..C4	-	
eQEP	Number	2 - EQEP1, EQEP2	2 - EQEP1, EQEP2	
	Registers	-	QEPSRCSEL	Select source as either device pins or cmpss/epwmxbar
	Other		Support for SinCos Transducers	
eCAP	Number	7 - ECAP1..7	3 - ECAP1..3	
	Registers	-	ECAPSYNCINSEL	Select sync source for ecap
HRCAP	Number	2 - HRCAP6, HRCAP7	1 - HRCAP3	
ePWM	Number	8 - EPWM1..8	7 - EPWM1..7	
	Registers	-	TBCTL3.OSSFRCEN	F28002x can now generate an EPWMxSYNCO with GLDCTL2[OSHTLD]
		SYNCSEL	EPWMSYNCINSEL	EPWMxSYNCI to EPWMxSYNCO path removed from F28002x
		TBCTL.SYNCOSEL	EPWMSYNCOUTEN	DCAEVT1 and DCBEVT1 are new sync options for F28002x
		TBCTL2.SYNCOSELX		
HRPWM	Number	8 - HRPWM1..8	4 - HRPWM1..4	
	Clock Source	EPWM1CLK	Respective EPWM	

3.4 Analog Module Differences

This section outlines the analog differences between F28002x and F28004x. The GPDAC and PGA are not present on the F28002x and the analog mux table is re-mapped.

Table 4. Analog Module Instances

Module	F28004x	F28002x
ADC(1)	3 - ADCA, ADCB, ADCC	2 - ADCA, ADCC
GPDAC	2 - GPDACA, GPDACB	-
CMPSS(1)	7 - CMPSS1 to CMPSS7	4 - CMPSS1 to CMPSS4
PGA	7 - PGA1 to PGA7	-
Temp Sensor	1 - (in ADCB ch 14)	1 - (in ADCC ch 12)

(1) In porting software from F28004x to F28002x (or the other way around), care must be taken to ensure that the correct ADC channels are used because of a difference in channel assignment, see Section 3.10.

3.5 Other Device Changes

This section describes feature differences between F28004x and F28002x that were not covered in the previous sections, as such the changes identified below must be considered when migrating applications between devices.

3.5.1 System Control Changes

This section highlights the System Control differences between F28002x and F28004x. There are XTAL module, PLL feature and Pie channel mapping changes between the two devices which are outlined in this section.

3.5.1.1 XTAL Module

The XTAL module has a few changes between F28004x and F28002x as highlighted in Table 5.

Table 5. XTAL Module Differences

Module	Category	F28004x	F28002x	Note
XTAL	Registers	X1CNT.X1CNT[9..0]	X1CNT.X1CNT[10..0]	
		-	XTALCR2	For pre-conditioning the GPIO mode of X1/X2
	Other	X1CNT.CLR is synchronous	X1CNT.CLR is asynchronous	

3.5.1.2 PLL

The PLL blocks of F28004x and F28002x devices are different. Table 6 lists the PLL features for both devices for comparison. For more information, see the *TMS320F28002x microcontrollers technical reference manual* (SPRUIN7).

Table 6. PLL Features

Feature	F28004x	F28002x
VCO Range	120 - 400 MHz	220 - 500 MHz
PLL Raw Clock Range	15 - 200 MHz	6 - 200 MHz
X1 Input Range (PLL enabled)	2 - 20 MHz	2 - 25 MHz
REFCLK Divider	No	Yes [1..32]
PLL Slip Detect	Yes	No (use DCC)
Fractional PLLMULT	Yes	No

Due to the PLL differences between the two devices, TI recommends to use the PLL set up function, SysCtrl_setClock() in C2000Ware to ensure proper PLL setting.

3.5.1.3 Pie Channel Mapping

Pie channel mapping between F28004x and F28002x is different due to peripheral module changes between these devices. Table 8 summarizes the common and unique pie channels on these two devices.

Table 7. Pie Channel Legend

Color	Description
	Pie channel common for both devices
	Pie channel applicable only for F28004x
	Pie channel applicable only for F28002x

Table 8. Pie Table Comparison

	INTx.1	INTx.2	INTx.3	INTx.4	INTx.5	INTx.6	INTx.7	INTx.8	INTx.9	INTx.10	INTx.11	INTx.12	INTx.13	INTx.14	INTx.15	INTx.16
INT1.y	ADCA1	ADCB1	ADCC1	XINT1	XINT2	-	TIMER0	WAKE/WDOG	-	-	-	-	-	-	-	-
INT2.y	EPWM1_TZ	EPWM2_TZ	EPWM3_TZ	EPWM4_TZ	EPWM5_TZ	EPWM6_TZ	EPWM7_TZ	EPWM8_TZ	-	-	-	-	-	-	-	-
INT3.y	EPWM1	EPWM2	EPWM3	EPWM4	EPWM5	EPWM6	EPWM7	EPWM8	-	-	-	-	-	-	-	-
INT4.y	ECAP1	ECAP2	ECAP3	ECAP4	ECAP5	ECAP6	ECAP7	-	-	-	ECAP3_HRC	-	-	ECAP6_HRCAL	ECAP7_HRCAL	-
INT5.y	EQEP1	EQEP2	-	CLB1	CLB2	-	-	-	SDFM1	-	-	-	SDFM1DR1	SDFM1DR2	SDFM1DR3	SDFM1DR4
INT6.y	SPIA_RX	SPIA_TX	SPIB_RX	SPIB_TX	-	-	-	-	-	-	-	-	-	-	-	-
INT7.y	DMA_CH1	DMA_CH2	DMA_CH3	DMA_CH4	DMA_CH5	DMA_CH6	-	-	-	-	FSITX_INT1	FSITX_INT2	FSIRX_INT1	FSIRX_INT2	CLAPROMCRC	DCC0
INT8.y	I2CA	I2CA_FIFO	I2CB	I2CB_FIFO	-	-	-	-	LINA_0	LINA_1	LINB_0	LINB_1	PMBUSA	-	-	DCC1
INT9.y	SCIA_RX	SCIA_TX	SCIB_RX	SCIB_TX	CANA_0	CANA_1	CANB_0	CANB_1	-	-	-	-	BGCRC	-	-	HICA
INT10.y	ADCA_EVT	ADCA2	ADCA3	ADCA4	ADCB_EVT	ADCB2	ADCB3	ADCB4	ADCC_EVT	ADCC2	ADCC3	ADCC4	-	-	-	-
INT11.y	CLA1_1	CLA1_2	CLA1_3	CLA1_4	CLA1_5	CLA1_6	CLA1_7	CLA1_8	-	-	-	-	-	-	-	-
INT12.y	XINT3	XINT4	XINT5	PBIST (MPOST)	FMC	-	FPU_OVERFLOW	FPU_UNDERFLOW	-	RAM_CORRECTABLE_ERROR	FLASH_CORRECTABLE_ERROR	RAM_ACCESS_VIOLATION	SYS_PLL_SLIP[1]	-	CLA_OVERFLOW	CLA_UNDERFLOW

(1) Not implemented in F28002x

3.5.2 Bootrom

This section outlines bootrom similarities and differences between F28004x and F28002x devices.

3.5.2.1 Bootrom Features

For specific bootrom features on F28004x and F28002x for hardware or software consideration, see Table 9.

Table 9. Bootrom Comparison Table

	F28004x	F28002x
System Debug (ERAD)	NMI is disabled	NMI is enabled. Bootrom exception handler is updated for this NMI
HWBIST	HWBIST is not available	HWBIST is available
CPU Boot Mode GPIO Assignments	On 64-pin package, F28004x and F28002x have similar options however the BOOTDEFx values are different. For boot mode GPIO assignment on the other pin package types, see the *Bootrom* section in the device-specific data sheet.	
BMSP Restrictions - Do not use pins	GPIO20-33, GPIO36, GPIO38 and GPIO60-233	GPIO20, GPIO21,GPIO36,GPIO38,GPIO47-60 and GPIO63-223
RAM Initialization	RAM initialization occurs on POR and XRS	RAM initialization occurs only on POR
ROM Table	ROM tables for F28004x and F28002x are different. For details, see the device-specific TRM.	
PBIST(MPOST) Status Flag	Flag is reset for every reset type	Flag is reset only for POR reset type
PBIST(MPOST) Execution Speed	Will execute either at maximum SYSCLK speed or INTOSC clock	Will execute either at maximum SYSCLK or half of maximum SYSCLK speed

3.5.2.2 BOOTDEF Values Comparison

Due to GPIO and Flash memory differences between F28004x and F28002x, BOOTDEF values and options between the two devices will not be the same. Table 11 and Table 12 outlines these differences that will have to be considered when migrating an application from F28004x to F28002x.

Table 10. Boot options Legend

Color	Description
	Options common for both devices but BOOTDEFx values may differ
	Options applicable only for F28004x
	Options applicable only for F28002x

Table 11. Bootloaders and GPIO Assignment Comparison

Bootloader	Option[(1)]	BOOTDEFx	F28004x	F28002x
Parallel	0	0x00	D0-D7=0 to 7; DSP=16; Host=11	D0-D7=28,1 to 7; DSP=16; Host=29
	1	0x20	n/a	D0-D7=0 to 7; DSP=16; Host=11
SCIA	0	0x01	TX=29; RX=28	TX=29; RX=28
	1	0x21	TX=16; RX=17	TX=16; RX=17
	2	0x41	TX=8; RX=9	TX=8; RX=9
	3	0x61	TX=48; RX=49	TX=2; RX=3
	4	0x81	TX=24; RX=25	TX=16; RX=3
CANA	0	0x02	TX=32; RX=33	TX=4; RX=5
	1	0x22	TX=4; RX=5	TX=32; RX=33
	2	0x42	TX=31; RX=30	TX=2; RX=3
	3	0x62	TX=37; RX=35	n/a

(1) Consult device datasheet to ensure that the GPIOs assigned to these options are available for a specific package

Table 11. Bootloaders and GPIO Assignment Comparison (continued)

Bootloader	Option[1]	BOOTDEFx	F28004x	F28002x
SPI	0	0x06	n/a	SIMO=2; SOMI=1; CLK=3; STE=5
	1	0x26	SIMO=8; SOMI=10; CLK=9; STE=11	SIMO=16; SOMI=1; CLK=3; STE=0
	2	0x46	SIMO=54; SOMI=55; CLK=56; STE=57	SIMO=8; SOMI=10; CLK=9; STE=11
	3	0x66	SIMO=16; SOMI=17; CLK=56; STE=57	SIMO=8; SOMI=17; CLK=9; STE=11
	4	0x86	SIMO=8; SOMI=17; CLK=9; STE=11	n/a
I2C	0	0x07	SDA=32; SCL=33	SDA=32; SCL=33
	1	0x27	n/a	SDA=0; SCL=1
	2	0x47	SDA=26; SCL=27	SDA=10; SCL=8
	3	0x67	SDA=42; SCL=43	n/a

Table 12. Boot Modes Comparison

Boot Mode	Option	BOOTDEFx	F28004x	F28002x
Flash	0	0x03	Entry=0x00080000; Bank/Sector=0/0	Entry=0x00080000; Bank/Sector=0/0
	1	0x23	Entry=0x0008EFF0; Bank/Sec\tor=0/14	Entry=0x00084000; Bank/Sector=0/4
	2	0x43	Entry=0x00090000; Bank/Sector=1/0	Entry=0x00088000; Bank/Sector=0/8
	3	0x63	Entry=0x0009EFF0; Bank/Sector=1/14	Entry=0x0008EFF0; Bank/Sector=0/14
Wait	0	0x04	Watchdog enabled	Watchdog enabled
	1	0x24	Watchdog disabled	Watchdog disabled
RAM	0	0x05	Entry=0x00000000	Entry=0x00000000

3.5.3 CLA, CLB, DMA and Motor Control Libraries

There are no functional changes in CLB, DMA and Motor Control Libraries.

Table 13. CLA, CLB, DMA and Motor Control Libraries

Modules	F28004x	F28002x
CLA	Yes	No
CLB	4 tiles	2 tiles
DMA	6 channels	6 channels
Motor Control Libraries in ROM	Available in F28004xC	Available in F28002xC

3.5.4 ERAD

The ERAD module has a number of changes between F28004x and F28002x as highlighted in Table 14.

Table 14. ERAD Module Differences

Module	Category	F28004x	F28002x	Note
ERAD	Features	-	Event Masking and Exporting	EBC Unit on F28002x supports event OR/AND, masking and exporting
		-	Cumulative Mode	SEC Unit on F28002x supports a cummulative mode over several start/stop events
		-	CRC Unit	F28002x has CRC units to monitor CPU buses and compute CRC when self-test code is executed
		32 Event Selector Options	128 Event Selector Options	Connections to ADC, CMPSS, EPWM and other sources have been added to F28002x
	Registers	-	GLBL_NMI_CTL	Global Debug NMI Control
		-	GLBL_EVENT_AND_MASK	Global Bus Comparator Event AND Mask Register
		-	GLBL_EVENT_OR_MASK	Global Bus Comparator Event OR Mask Register
		-	GLBL_AND_EVENT_INT_MASK	Global AND Event Interrupt Mask Register
		-	GLBL_OR_EVENT_INT_MASK	Global OR Event Interrupt Mask Register
		-	CTM_INPUT_SEL_2	Counter Input Select Extension Register
		-	CTM_INPUT_COND	Counter Input Conditioning Register
		-	CRC_GLOBAL_CTRL	CRC Global Control Register
		-	CRC_CURRENT	Reads Current CRC Value
		-	CRC_SEED	CRC Seed Register
		-	CRC_QUALIFIER	CRC Compute Qualification Register

3.5.5 GPIO

The GPIO module in F28002x has a new registers for reading back the value written in GPyDAT. These registers is not available in F28004x. The registers are GPyDAT_R and description is as follows:

- **GPyDAT_R** are read-only registers which return the values written on GPyDAT registers instead of pin status. Writes to these registers have no effect.

3.6 Power Management

There are different options to power the F28004x and F28002x devices. F28004x has flexibility in that it supports both dual-rail (3.3 V and 1.2 V) or single-rail (3.3 V) with internal VREG or DCDC to provide the 1.2 V rail. F28002x only supports single-rail 3.3 V supply with the internal VREG providing the 1.2 V rail. This section describes the power management differences and similarities between the two devices.

3.6.1 LDO/VREG

The F28002x device supports internal VREG only. External VREG is not supported because the VREGENZ pin on the F28002x device has been converted to a GPIO, GPIO39. F28004x supports both internal VREG or external source.

3.6.2 DCDC

The F28002x device does not have a DCDC while F28004x has an internal DCDC that requires minimal external components (inductor and capacitor).

3.6.3 POR/BOR

There are no functional changes for the POR and BOR.

3.7 Power Consumption

There is not a significant difference in power consumption between the F28002x and the F28004x if the same number of peripherals are being utilized and internal VREG is being used for both. However, since the F28004x device has the option to supply VREG externally or through the internal DCDC, it has the capability to be more power efficient.

3.8 Memory Module Changes

The available FLASH and RAM memories on the F28002x family of devices are reduced compared to the F28004x devices. For specific details on available memory capacity on the F28002x family of devices, see the *TMS320F28002x microcontrollers data sheet* (SPRSP45).

3.8.1 FLASH

F28004x has a total of 256 KBytes of Flash memory in two banks while F28002x has 128KBytes in one bank.

3.8.2 RAM

F28004x has a total RAM size of 100.5KBytes while F28002x has 24KBytes.

3.9 GPIO Multiplexing Diagram

Table 16 outlines the differences and similarities that exist in the GPIO mux between F28002x and F28004x. The legend for this table is Table 15. The main changes highlighted in Table 16 are the absence of SDFM mux positions and the DCDC GPIO support pins from F28002x. The other notable change is the addition of HIC mux positions and the use of X1 as a GPIO pin in the F28002x device if external clock is not used.

NOTE: This comparison guide focuses on the super-set devices F280049 and F280025. The Q-grade Q100 Part Numbers do not have GPIO12 or GPIO13 in either device.

Table 15. Mux Legend

Color	Description
	mux function common for both devices
	mux function applicable only for F28004x
	mux function applicable only for F28002x

Table 16. GPIO Mux Table Comparison

0,4,8,12	1	2	3	5	6	7	9	10	11	13	14	15	ALT
GPIO0	EPWM1_A				I2CA_SDA	SPIA_STE	FSIRXA_CLK		CLB_OUTPUTXBAR8			HIC_BASESEL1	
GPIO1	EPWM1_B				I2CA_SCL	SPIA_SOMI			CLB_OUTPUTXBAR7	HIC_A2	FSI_TDM_TX1	HIC_D10	
GPIO2	EPWM2_A			OUTPUTXBAR1	PMBUSA_SDA	SPIA_SIMO	SCIA_TX	FSIRXA_D1	I2CB_SDA	HIC_A1	CANA_TX	HIC_D9	
GPIO3	EPWM2_B	OUTPUTXBAR2		OUTPUTXBAR2	PMBUSA_SCL	SPIA_CLK	SCIA_RX	FSIRXA_D0	I2CB_SCL	HIC_NOE	CANA_RX	HIC_D4	
GPIO4	EPWM3_A			OUTPUTXBAR3	CANA_TX	SPIB_CLK	EQEP2_STROBE	FSIRXA_CLK	CLB_OUTPUTXBAR6	HIC_BASESEL2		HIC_NWE	
GPIO5	EPWM3_B		OUTPUTXBAR3		CANA_RX	SPIA_STE	FSITXA_D1	CLB_OUTPUTXBAR5		HIC_A7	HIC_D4	HIC_D15	
GPIO6	EPWM4_A	OUTPUTXBAR4	SYNCOUT	EQEP1_A	CANB_TX	SPIB_SOMI	FSITXA_D0		FSITXA_D1	HIC_NBE1	CLB_OUTPUTXBAR8	HIC_D14	
GPIO7	EPWM4_B		OUTPUTXBAR5	EQEP1_B	CANB_RX	SPIB_SIMO	FSITXA_CLK	CLB_OUTPUTXBAR2		HIC_A6		HIC_D14	
GPIO8	EPWM5_A	CANB_TX	ADCSOCAO	EQEP1_STROBE	SCIA_TX	SPIA_SIMO	I2CA_SCL	FSITXA_D1	CLB_OUTPUTXBAR5	HIC_A0	FSI_TDM_CLK	HIC_D8	
GPIO9	EPWM5_B	SCIB_TX	OUTPUTXBAR6	EQEP1_INDEX	SCIA_RX	SPIA_CLK		FSITXA_D0	LINB_RX	HIC_BASESEL0	I2CB_SCL	HIC_NRDY	
GPIO10	EPWM6_A	CANB_RX	ADCSOCBO	EQEP1_A	SCIB_TX	SPIA_SOMI	I2CA_SDA	FSITXA_CLK	LINB_TX	HIC_NWE	FSI_TDM_TX0		
GPIO11	EPWM6_B	SCIB_RX	OUTPUTXBAR7	EQEP1_B	SCIB_RX	SPIA_STE	FSIRXA_D1	LINB_RX	EQEP2_A	SPIA_SIMO	HIC_D6	HIC_NBE0	
GPIO12	EPWM7_A	CANB_TX		EQEP1_STROBE	SCIB_TX	PMBUSA_CTL	FSIRXA_D0	LINB_TX	SPIA_CLK	CANA_RX	HIC_D13	HIC_INT	
GPIO13	EPWM7_B	CANB_RX		EQEP1_INDEX	SCIB_RX	PMBUSA_ALERT	FSIRXA_CLK	LINB_RX	SPIA_SOMI	CANA_TX	HIC_D11	HIC_D5	
GPIO14	EPWM8_A	SCIB_TX		I2CB_SDA	OUTPUTXBAR3	PMBUSA_SDA	SPIB_CLK	EQEP2_A	LINB_TX	EPWM3_A	CLB_OUTPUTXBAR7	HIC_D15	
GPIO15	EPWM8_B	SCIB_RX		I2CB_SCL	OUTPUTXBAR4	PMBUSA_SCL	SPIB_STE	EQEP2_B	LINB_RX	EPWM3_B	CLB_OUTPUTXBAR6	HIC_D12	
GPIO16	SPIA_SIMO	CANB_TX	OUTPUTXBAR7	EPWM5_A	SCIA_TX	SD1_D1	EQEP1_STROBE	PMBUSA_SCL	XCLKOUT	EQEP2_B	SPIB_SOMI	HIC_D1	
GPIO17	SPIA_SOMI	CANB_RX	OUTPUTXBAR8	EPWM5_B	SCIA_RX	SD1_C1	EQEP1_INDEX	PMBUSA_SDA	CANA_TX			HIC_D2	
GPIO18	SPIA_CLK	SCIB_TX	CANA_RX	EPWM6_A	I2CA_SCL	SD1_D2	EQEP2_A	PMBUSA_CTL	XCLKOUT	LINB_TX	FSI_TDM_CLK	HIC_INT	X2
GPIO19	SPIA_STE		CANA_TX	EPWM6_B	I2CA_SDA		EQEP2_B	PMBUSA_ALERT	CLB_OUTPUTXBAR1	LINB_RX	FSI_TDM_TX0	HIC_NBE0	X1
GPIO22	EQEP1_STROBE		SCIB_TX		SPIB_CLK	SD1_D4	LINA_TX	CLB_OUTPUTXBAR1	LINB_TX	HIC_A5	EPWM4_A	HIC_D13	VFBSW
GPIO23	EQEP1_INDEX				SPIB_STE		LINA_RX		LINB_RX	HIC_A3	EPWM4_B	HIC_D11	VSW
GPIO24	OUTPUTXBAR1	EQEP2_A		EPWM8_A	SPIB_SIMO	SD1_D1	LINB_TX	PMBUSA_SCL	SCIA_TX	ERRORSTS		HIC_D3	
GPIO25	OUTPUTXBAR2	EQEP2_B		EQEP1_A	SPIB_SOMI	SD1_C1	FSITXA_D1	PMBUSA_SDA	SCIA_RX		HIC_BASESEL0		
GPIO26	OUTPUTXBAR3	EQEP2_INDEX		OUTPUTXBAR3	SPIB_CLK	SD1_D2	FSITXA_D0	PMBUSA_CTL	I2CA_SDA		HIC_D0	HIC_A1	
GPIO27	OUTPUTXBAR4	EQEP2_STROBE		OUTPUTXBAR4	SPIB_STE	SD1_C2	FSITXA_CLK	PMBUSA_ALERT	I2CA_SCL		HIC_D1	HIC_A4	
GPIO28	SCIA_RX		EPWM7_A	OUTPUTXBAR5	EQEP1_A	SD1_D3	EQEP2_STROBE	LINA_TX	SPIB_CLK	ERRORSTS	I2CB_SDA	HIC_NOE	
GPIO29	SCIA_TX		EPWM7_B	OUTPUTXBAR6	EQEP1_B	SD1_C3	EQEP2_INDEX	LINA_RX	SPIB_STE	ERRORSTS	I2CB_SCL	HIC_NCS	
GPIO30	CANA_RX		SPIB_SIMO	OUTPUTXBAR7	EQEP1_STROBE	SD1_D4	FSIRXA_CLK		EPWM1_A		HIC_D8		
GPIO31	CANA_TX		SPIB_SOMI	OUTPUTXBAR8	EQEP1_INDEX	SD1_C4	FSIRXA_D1		EPWM1_B		HIC_D10		

Table 16. GPIO Mux Table Comparison (continued)

0,4,8,12	1	2	3	5	6	7	9	10	11	13	14	15	ALT
GPIO32	I2CA_SDA		SPIB_CLK	EPWM8_B	LINA_TX	SD1_D3	FSIRXA_D0	CANA_TX		ADCSOCBO		HIC_INT	
GPIO33	I2CA_SCL		SPIB_STE	OUTPUTXBAR4	LINA_RX	SD1_C3	FSIRXA_CLK	CANA_RX	EQEP2_B	ADCSOCAO		HIC_D0	
GPIO34	OUTPUTXBAR1				PMBUSA_SDA					HIC_NBE1	I2CB_SDA	HIC_D9	
GPIO35	SCIA_RX		I2CA_SDA	CANA_RX	PMBUSA_SCL	LINA_RX	EQEP1_A	PMBUSA_CTL			HIC_NWE	TDI	
GPIO37	OUTPUTXBAR2		I2CA_SCL	SCIA_TX	CANA_TX	LINA_TX	EQEP1_B	PMBUSA_ALERT			HIC_NRDY	TDO	
GPIO39					CANB_RX	FSIRXA_CLK	EQEP2_INDEX		CLB_OUTPUTXBAR2	SYNCOUT	EQEP1_INDEX	HIC_D7	
GPIO40	SPIB_SIMO			EPWM2_B	PMBUSA_SDA	FSIRXA_D0	SCIB_TX	EQEP1_A	LINB_TX		HIC_NBE1	HIC_D5	
GPIO41				EPWM2_A	PMBUSA_SCL	FSIRXA_D1		EQEP1_B	LINB_RX	HIC_A4	SPIB_SOMI	HIC_D12	
GPIO42		LINA_RX	OUTPUTXBAR5	PMBUSA_CTL	I2CA_SDA			EQEP1_STROBE	CLB_OUTPUTXBAR3		HIC_D2	HIC_A6	
GPIO43			OUTPUTXBAR6	PMBUSA_ALERT	I2CA_SCL			EQEP1_INDEX	CLB_OUTPUTXBAR4		HIC_D3	HIC_A7	
GPIO44			OUTPUTXBAR7	EQEP1_A		FSITXA_CLK		CLB_OUTPUTXBAR3		HIC_D7		HIC_D5	
GPIO45			OUTPUTXBAR8			FSITXA_D0		CLB_OUTPUTXBAR4				HIC_D6	
GPIO46			LINA_TX			FSITXA_D1						HIC_NWE	
GPIO56	SPIA_CLK			EQEP2_STROBE	SCIB_TX	SD1_D3	SPIB_SIMO		EQEP1_A				
GPIO57	SPIA_STE			EQEP2_INDEX	SCIB_RX	SD1_C3	SPIB_SOMI		EQEP1_B				
GPIO58				OUTPUTXBAR1	SPIB_CLK	SD1_D4	LINA_TX	CANB_TX	EQEP1_STROBE				
GPIO59				OUTPUTXBAR2	SPIB_STE	SD1_C4	LINA_RX	CANB_RX	EQEP1_INDEX				
AIO224												HIC_A3	
AIO225												HIC_NWE	
AIO226												HIC_A1	
AIO227												HIC_NBE0	
AIO228												HIC_A0	
AIO230												HIC_BASESEL2	
AIO231												HIC_BASESEL1	
AIO232												HIC_BASESEL0	
AIO233												HIC_A4	
AIO237												HIC_A6	
AIO238												HIC_NCS	
AIO239												HIC_A5	
AIO241												HIC_NBE1	
AIO242												HIC_A2	
AIO244												HIC_A7	
AIO245												HIC_NOE	

3.10 Analog Multiplexing Diagram

Table 18 outlines the differences and similarities that exist in the analog mux between F28002x and F28004x. The legend for this table is Table 17. The main changes highlighted in Table 18 are the absence of ADCB, DAC, PGA and the reduction of the number of CMPSS modules in F28002x from the seven that exist in F28004x to four. The other notable change is that in the F28004x device, the analog modules were grouped. This grouping does not exist in F28002x. This is particularly important for the CMPSS as highlighted in the table.

Table 17. Mux Legend

Color	Description
	mux function common for both devices
	mux function applicable only for F28004x
	mux function applicable only for F28002x

Table 18. F28004x and F28002x Analog Mux Table Differences

(F28004x Pin Name)	F28004x Group Name	PKG	Always Connected (NO MUX)					Comparator Subsystem (MUX)				AIO Input
F28002x Pin Name		64 PM	ADCA	ADCB	ADCC	PGA	DAC	High Positive	High Negative	Low Positive	Low Negative	
VREFHIA	-	16										
VREFHIB	-											
VREFHIC	-											
VREFLOA	-	17	A13									
VREFLOB	-			B13								
VREFLOC	-				C13							
F28004x Analog Group 1								**F28004x CMP1**				
(A3)	G1_ADCAB		A3					HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO233
(A2/B6/PGA1_OF) A2/C9	PGA1_OF	9	A2	B6	C9	PGA1_OF		HPMXSEL = 0		LPMXSEL = 0		AIO224
(C0) A11/C0	G1_ADCC	12	A11		C0			HPMXSEL = 1	HNMXSEL = 1	LPMXSEL = 1	LNMXSEL = 1	AIO237
(PGA1_IN)	PGA1_IN					PGA1_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA1_GND) A15/C7	PGA1_GND	10	A15		C7	PGA1_GND		HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO233
(-)	PGA1_OUT		A11	B7		PGA1_OUT		HPMXSEL = 4		LPMXSEL = 4		
F28004x Analog Group 2								**F28004x CMP2**				
(A5)	G2_ADCAB		A5					HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO234
(A4/B8/PGA2_OF) A4/C14	PGA2_OF	23	A4	B8	C14	PGA2_OF		HPMXSEL = 0 ***CMP4_HPMXSEL = 3***	***CMP4_HNMXSEL = 0***	LPMXSEL = 0 ***CMP4_LPMXSEL = 3***	***CMP4_LNMXSEL = 0***	AIO225
(C1) A12/C1	G2_ADCC	18	A12		C1			HPMXSEL = 1 ***CMP4_HPMXSEL = 2***	HNMXSEL = 1	LPMXSEL = 1 ***CMP4_LPMXSEL = 2***	LNMXSEL = 1	AIO238
(PGA2_IN)	PGA2_IN					PGA2_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA2_GND) A8/C11	PGA2_GND	20	A8		C11	PGA2_GND		HPMXSEL = 4 ***CMP4_HPMXSEL = 4***		LPMXSEL = 4 ***CMP4_LPMXSEL = 4***		AIO241
(-)	PGA2_OUT		A12	B9		PGA2_OUT		HPMXSEL = 4		LPMXSEL = 4		
F28004x Analog Group 3								**F28004x CMP3**				
(B3/VDAC) A3/C5/VDAC	G3_ADCAB	8	A3	B3	C5		VDAC	HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO242
(B2/C6/PGA3_OF) C6	PGA3_OF	7		B2	C6	PGA3_OF		HPMXSEL = 0		LPMXSEL = 0		AIO226
(C2) A5/C2	G3_ADCC	13	A5		C2			HPMXSEL = 1	HNMXSEL = 1	LPMXSEL = 1	LNMXSEL = 1	AIO244
(PGA3_IN)	PGA3_IN					PGA3_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA3_GND)	PGA3_GND	10				PGA3_GND						
(-)	PGA3_OUT			B10		PGA3_OUT		HPMXSEL = 4		LPMXSEL = 4		

Table 18. F28004x and F28002x Analog Mux Table Differences (continued)

(F28004x Pin Name) / F28002x Pin Name	F28004x Group Name	PKG 64 PM	Always Connected (NO MUX) ADCA	ADCB	ADCC	PGA	DAC	Comparator Subsystem (MUX) High Positive	High Negative	Low Positive	Low Negative	AIO Input
F28004x Analog Group 4								**F28004x CMP4**				
(B5)	G4_ADCAB			B5				HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO243
(B4/C8/PGA4_OF) A9/C8	PGA4_OF	24	A9	B4	C8	PGA4_OF		HPMXSEL = 0 ***CMP2_HPMXSEL = 2***		LPMXSEL = 0 ***CMP2_LPMXSEL = 2***		AIO227
(C3) A7/C3	G4_ADCC	19	A7		C3			HPMXSEL = 1	HNMXSEL = 1	LPMXSEL = 1	LNMXSEL = 1	AIO245
(PGA4_IN)	PGA4_IN					PGA4_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA4_GND)	PGA4_GND	20				PGA4_GND						
(-)	PGA4_OUT			B11	C9	PGA4_OUT		HPMXSEL = 4		LPMXSEL = 4		
F28004x Analog Group 5								**F28004x CMP5**				
(A7)	G5_ADCAB		A7					HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO235
(A6/PGA5_OF) A6	PGA5_OF	6	A6			PGA5_OF		HPMXSEL = 0 ***CMP1_HPMXSEL = 2***		LPMXSEL = 0 ***CMP1_LPMXSEL = 2***		AIO228
(C4) A14/C4	G5_ADCC	11	A14		C4			HPMXSEL = 1 ***CMP3_HPMXSEL = 4***	HNMXSEL = 1	LPMXSEL = 1 ***CMP3_LPMXSEL = 4***	LNMXSEL = 1	AIO239
(PGA5_IN)	PGA5_IN					PGA5_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA5_GND)	PGA5_GND	10				PGA5_GND						
(-)	PGA5_OUT		A14			PGA5_OUT		HPMXSEL = 4		LPMXSEL = 4		
F28004x Analog Group 6								**F28004x CMP6**				
(A9)	G6_ADCAB		A9					HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO236
(A8/PGA6_OF)	PGA6_OF		A8			PGA6_OF		HPMXSEL = 0		LPMXSEL = 0		AIO229
(C5)	G6_ADCC				C5			HPMXSEL = 1	HNMXSEL = 1	LPMXSEL = 1	LNMXSEL = 1	AIO240
(PGA6_IN)	PGA6_IN					PGA6_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA6_GND)	PGA6_GND	20				PGA6_GND						
(-)	PGA6_OUT		A15			PGA6_OUT		HPMXSEL = 4		LPMXSEL = 4		
F28004x Analog Group 7								**F28004x CMP7**				
(B0)	G7_ADCAB			B0				HPMXSEL = 3	HNMXSEL = 0	LPMXSEL = 3	LNMXSEL = 0	AIO241
(A10/B1/C10/PGA7_OF) A10/C10	PGA7_OF	25	A10	B1	C10	PGA7_OF		HPMXSEL = 0 ***CMP2_HPMXSEL = 3***	***CMP2_HNMXSEL = 0***	LPMXSEL = 0 ***CMP2_LPMXSEL = 3***	***CMP2_LNMXSEL = 0***	AIO230
(C14)	G7_ADCC				C14			HPMXSEL = 1	HNMXSEL = 1	LPMXSEL = 1	LNMXSEL = 1	AIO246
(PGA7_IN)	PGA7_IN					PGA7_IN		HPMXSEL = 2		LPMXSEL = 2		
(PGA7_GND)	PGA7_GND					PGA7_GND						
(-)	PGA7_OUT			B12	C11	PGA7_OUT		HPMXSEL = 4		LPMXSEL = 4		

Table 18. F28004x and F28002x Analog Mux Table Differences (continued)

(F28004x Pin Name)	F28004x Group Name	PKG	Always Connected (NO MUX)					Comparator Subsystem (MUX)				AIO Input
F28002x Pin Name		64 PM	ADCA	ADCB	ADCC	PGA	DAC	High Positive	High Negative	Low Positive	Low Negative	
Other F28004x Analog												
(A0/B15/C15/DACA_OUT) A0/C15		15	A0	B15	C15		DACA_OUT	***CMP3_HPMXSEL = 2***		***CMP3_LPMXSEL = 2***		AIO231
(A1/DACB_OUT) A1		14	A1				DACB_OUT	***CMP1_HPMXSEL = 4***		***CMP1_LPMXSEL = 4***		AIO232
(C12)					C12							AIO247
-	TempSensor			B14	C12							

4 Application Code Migration From F28004x to F28002x

The following section describes code changes when migrating from F28004x to F28002x. Software examples for the new features in F28002x are also discussed in this section.

4.1 C2000Ware Header Files

Header files for both F28002x and F28004x devices are available in C2000Ware under the device_support sub directory.

4.2 Linker command Files

Linker command files for both F28002x and F28004x devices are available in C2000Ware under the device_support sub directory. Specific to F28002x, which have to be compiled to the Embedded Application Binary Interface (EABI) format, the section names would also need to conform to the EABI standard. For more details, see Table 19.

4.3 Minimum Compiler Version Requirement for TMU Type 1

Code Composer Studio™ (CCS) compiler version 18.12.0.LTS supports the new instruction sets for TMU Type1 new instruction sets.

4.4 C2000Ware Examples

C2000Ware has examples specific for both F28002x and F28004x devices.

4.5 Specific Use Cases related to F28002x new features

This section outlines the new examples in C2000Ware for the F28002x device to support the new features such as HIC and FID/NLPID.

4.5.1 HIC

C2000Ware has examples hic_1 and hic_2 that demonstrate the functionality of the new HIC module on the F28002x device.

4.5.2 FINTDIV

C2000Ware has examples that demonstrate the functionality of the new fast integer division instructions for the F28002x device.

4.5.3 TMU Type1

C2000Ware DCL under control libraries has examples that demonstrate the two new instructions (IEXP2F32 and LOG2F32) to support NLPID in the F28002x device.

4.6 EABI Support

In the past, F28004x applications have always supported the Common Object File Format (COFF) binary executable output. COFF has several limitations. One of which is that the symbolic debugging information is not capable of supporting C/C++. There is also a limit on the maximum number of sections and length of section names and source files, among other things. COFF is also not an industry standard. For these reasons, C2000 is now migrating to Embedded Application Binary Interface (EABI) format and F28002x is one of the first devices to support it. EABI and COFF are incompatible and conversion between the two formats is not possible. This section provides summary of COFF and EABI differences and useful links that provide more guidelines in migrating applications from COFF to EABI.

- EABI key differences with COFF:
  - Direct initialization
    - Uninitialized data is zero by default in EABI.
    - Initialization of RW data is accomplished via linker-generated compressed copy tables in EABI.
  - C++ language support
    - C++ inline function semantics: In COFF, inline functions are treated as static inline and this causes issues for functions that cannot be inlined or have static data. In EABI, inline functions without the ‘static’ qualifier have external linkage.
    - Better template instantiation: COFF uses a method called late template instantiation and EABI uses early template instantiation. Late template instantiation can run into issues with library code and can result in long link times. Early instantiation uses ELF COMDAT to guarantee templates are always instantiated properly and at most one version of each instantiation is present in the final executable.
    - Table-Driven Exception Handling (TDEH): Almost zero impact on code performance as opposed to COFF which uses setjmp/longjmp to implement C++ exceptions Features enabled by EABI.
  - Features enabled by EABI
    - Location attribute: Specify the run-time address of a symbol in C-source code.
    - Noinit/persistent attribute: Specify if a symbol should not be initialized during C auto initialization.
    - Weak attribute: Weak symbol definitions are pre-empted by strong definitions. Weak symbol references are not required to be resolved at link time. Unresolved weak symbols resolve to 0.
    - External aliases: In COFF, the compiler will make A an alias to B if all calls to A can be replaced with B. A and B must be defined in the same file. In EABI, the compiler will make A an alias to B even if B is external.
  - Calling convention
    - Scalar calling convention is identical between COFF and EABI
    - Struct calling convention (EABI)
      - Single field structs are passed/returned by value corresponding to the underlying scalar types.
      - For FPU32, homogenous float structs with size less than 128 bits will be passed by value.
      - Passed in R0H-R3H, then by value on the stack.
      - Structs that are passed by value are also candidates for register allocation.
      - For FPU64, the same applies for 64-bit doubles(R0-R3).
  - Double memory size
    - In EABI, double is 64-bit size while in COFF, double is still represented as 32-bit size.
    - C/C++ requires that double be able to represent integer types with at least 10 decimal digits, which effectively requires 64-bit double precision.

- Sections overview:
  Table 19 summarizes the section names for COFF and EABI. These are compiler-generated sections.

Table 19. Section Names

Description	COFF	EABI
Read-Only Sections		
Const data	.econst	.const
Const data above 22-bits	.farconst	.farconst
Code	.text	.text
Pre-main constructors	.pinit	.init_array
Exception handling	N/A	.c28xabi.exidx/.c28xabi.extab
Read-Write Sections		
Uninitialized data	.ebss	.bss
Initialized data	N/A	.data
Uninitialized data above 22-bits	.farbss	.farbss
Initialized data above 22-bits	N/A	.fardata
Heap	.esysmem	.sysmem
Stack	.stack	.stack
CIO Buffer	.cio	.bss:cio

- Resources:
  For more information regarding EABI and the migration process, see the resources on the links below:
  - Wiki: http://processors.wiki.ti.com/index.php/EABI
  - Wiki: http://processors.wiki.ti.com/index.php/C2000_EABI_Migration
  - C28 EABI Specifications: *C28x embedded application binary interface* (SPRAC71)

4.6.1 Flash API

F28004x has two Flash banks. F28002x has only one Flash bank. Hence, the F28002x Flash API library (FlashAPI_F28002x_FPU32.lib) supports erase, program and verify operations only for the Flash Bank0 address range. Compared to the F28004x Flash API library (F021_API_F28004x_FPU32.lib), the F28002x Flash API is enhanced to return an error when an invalid address is provided for erase, blank-check, program and verify functions. Also, the F28002x Flash API is enhanced to return an error when an invalid programming mode is provided for program operation. Fapi_getLibraryInfo() in FlashAPI_F28002x_FPU32.lib returns the Flash API minor version as 57 (F28004x Flash API returns 56 as the API minor version). The F28002x Flash API library is compiled for EABI format, whereas, the F28004x Flash API library is compiled for legacy COFF. F28002x Flash API size is approximately 5.5KB. Note that F28004x and F28002x have the same Bank0 memory map and sector sizes. Also, the Flash wait-state configuration requirement is the same between the two devices. Hence, the F28002x Flash API version is chosen as 1.57.00.00, an update of F28004x Flash API V1.56.01.00. These features are summarized in Table 20

Table 20. Flash API Differences

Feature	F28004x	F28002x
Library Name	F021_API_F28004x_FPU32.lib	FlashAPI_F28002x_FPU32.lib
Library Executable Output	COFF (with future EABI support)	EABI
Erase, Blank-check, Program and Verify	Operation on two banks	Operation on one bank
Flash Wait States	Same wait states on both devices	
FlashAPI Minor Version	56	57

4.6.2 NoINIT Struct Fix (linker command)

With EABI, the SECTIONS area of a linker command file has to be modified as shown in the example below in order for the registers or memory areas to not be initialized to a zero value. This is important as failure to make this modification can result to unintended behavior when register bits are forced to zero during start up. By default, EABI initializes registers or memory areas defined in the SECTIONS part of the linker to zero.

Linker modification example:

```
SECTIONS
{
:
Regs1File :> REG1_ADDR, type=NOINIT
Regs2File :> REG2_ADDR, type=NOINIT
:
}
```

4.6.3 Pre-Compiled Libraries

All F28002x libraries supplied by TI will be released as EABI. Future F28002x libraries created by customers should be generated and compiled as EABI as well.

5 References

- Texas Instruments: C28 EABI Specifications: *C28x Embedded Application Binary Interface*
- EABI wiki
- C2000 EABI Migration wiki
- Texas Insruments: *TMS320F28002x Microcontrollers Technical Reference Manual* (SPRUIN7)
- Texas Insruments: *TMS320F2838x Microcontrollers Technical Reference Manual*
- Texas Insruments: *TMS320F28002x Microcontrollers Data Sheet* (SPRSP45)

Revision History

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

Changes from Original (September 2019) to A Revision	Page
• Change references of InstaSPIN to Motor Control Libraries in ROM.	1
• Update was made in Section 1.1.	3
• Updates made in Section 2.3.	5
• Update was made in Section 3.9.	14

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

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