

# **Migrating From TMS320F2802x/2803x to TMS320F2806x**

---



---



---

Katie Enderle

## **ABSTRACT**

This application report describes differences between Texas Instruments' TMS320F2802x/2803x and TMS320F2806x microcontrollers to assist in application migration. While the main focus of this document is migration from 2802x/2803x to 2806x, this document is also useful if you are considering migrating in the reverse direction. Functions that are identical in both devices are not necessarily included. All efforts have been made to provide a comprehensive list of the differences between the two device groups in the C28x generation.

---

## **Contents**

1	Introduction .....	2
2	Central Processing Unit (CPU).....	2
3	Development Tools .....	3
4	Package and Pinout .....	4
5	Operating Frequency .....	5
6	Supply Voltage and Power Sequencing .....	5
7	Memory Map .....	5
8	Clocks and System Control.....	7
9	Peripherals.....	9
10	Interrupts .....	12
11	Errata Fixes .....	12
12	References .....	13

## **List of Tables**

1	SARAM Addresses .....	6
2	Sector Configuration Per Device .....	6
3	New, Updated, and Removed Registers .....	8
4	New, Updated, and Removed Registers .....	12

## **Trademarks**

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

## 1 Introduction

The TMS320F2802x, TMS320F2803x, and TMS320F2806x devices are members of the C2000™ Piccolo MCU platform for use within embedded control applications. The TMS320F2806x features the same enhanced control peripherals available on the TMS230F2802x and 2803x devices. In addition, the 2806x features:

- Direct memory access (DMA)
- Viterbi Complex Math and CRC Unit (VCU)
- Multichannel Buffered Serial Port (McBSP)
- High-Resolution Capture (HRCAP)
- USB controller + PHY
- Floating-Point Unit (FPU)
- Additional enhanced quadrature encoder pulse (eQEP) and enhanced capture (eCAP) peripherals

These new peripherals enable the firmware engineer to solve challenging control problems effectively.

For purposes of migration, these devices can be classified into two groups:

- TMS320F2802x and TMS320F2803x. This group will be referenced as 2802x/2803x.
- TMS320F2806x. This group will be referenced as 2806x.

For a full list of devices currently available within the 2802x, 2803x, and 2806x family, see the TI website at <http://www.ti.com/>.

As the focus of this document is to describe the differences between the two device groups, the descriptions are explained only to the extent of highlighting areas that require attention when moving an application from one device to the other. For a detailed description of features specific to each device, see the device-specific data manuals and user's guides available on the TI website at <http://www.ti.com/>. This application report does not cover the silicon exceptions or advisories that may be present on each device. For specific advisories and workarounds, see the following silicon erratas:

- *TMS320F28069, TMS320F28068, TMS320F28067, TMS320F28066, TMS320F28065, TMS320F28064, TMS320F28063, TMS320F28062 Piccolo MCU Silicon Errata (SPRZ342)*
- *TMS320F28030, TMS320F28031, TMS320F28032, TMS320F28033, TMS320F28034, TMS320F28035 Piccolo MCU Silicon Errata (SPRZ295)*
- *TMS320F28027, TMS320F28026, TMS320F28023, TMS320F28022, TMS320F28021, TMS320F28020, TMS320F280200 Piccolo MCU Silicon Errata (SPRZ292)*

---

**NOTE:** Always refer to the TMS data manual for information regarding any electrical specifications.

---

## 2 Central Processing Unit (CPU)

The 2806x devices are the first Piccolo devices to include the C28x+ floating-point unit and Viterbi complex math and CRC unit (VCU) CPU (C28x+FPU+VCU). C28x+FPU+VCU-based controllers have the same 32-bit fixed-point architecture as TI's existing C28x MCUs, but also include a single-precision (32-bit) IEEE 754 floating-point unit (FPU) similar to the FPU on TMS320x2832x/2833x devices. It is a very efficient C/C++ engine, enabling you to develop system control software and math algorithms using C/C++. Some 2806x devices also include a control law accelerator (CLA) independent of the main CPU; this is the same CLA used by some 2803x devices. Finally, 2806x devices include the new Viterbi complex math and CRC unit (VCU), a fully programmable block that accelerates the performance of communications-based algorithms by up to a factor of 8X over C28x devices without a VCU.

No changes have been made to the existing:

- C28x instructions
- C28x pipeline
- C28x emulation
- Memory bus architecture

New instructions to support floating point operations have been added as an extension to the standard C28x instruction set. This means code written for the C28x fixed-point CPU is 100% compatible with the C28x+FPU. The C28x+FPU latched overflow and underflow (LVF, LUF) flags are connected to the peripheral interrupt expansion (PIE) block. This makes debugging overflow and underflow issues much easier. For an introduction to the C28x+FPU, see the *TMS320C28x FPU Primer* ([SPRAAN9](#)) and the *TMS320C28x Digital Signal Controller Plus Floating Point Unit* online training that is located at the TI website at <http://focus.ti.com/docs/training/catalog/events/event.jhtml?sku=OLT107003>.

The C28x+FPU+VCU architecture and instruction set are documented in the following reference guides:

- *TMS320C28x DSP CPU and Instruction Set Reference Guide* ([SPRU430](#)). This document also applies to the C28x+FPU.
- *TMS320C28x Floating Point Unit and Instruction Set Reference Guide* ([SPRUEO2](#)). This is a supplement to [SPRU430](#).
- *TMS320x2803x Piccolo Control Law Accelerator (CLA) Reference Guide* ([SPRUGE6](#))
- *TMS320x2806x Piccolo Technical Reference Manual* ([SPRUH18](#))

For more information on the Viterbi complex math and CRC (VCU) unit, see [Section 9.1.2](#).

### 3 Development Tools

A new set of header files and peripheral examples are available for the 2806x with the same structure as the 2802x/2803x header files. The F2806x Header Files and Peripheral Examples are included in the *controlSUITE zip package* ([SPRCA85](#)) under the `device_support` folder.

The C28x+FPU+VCU on the 2806x is supported with Code Composer Studio™ software with CodeGen Tools v6.0 or later. Check the Code Composer Studio update advisor for future updates. When building for native floating-point, you must use the correct run-time support library. For example, *rts2800\_fpu32.lib* for C or *rts2800\_fpu32\_eh.lib* for C++. These libraries are supplied with the compiler.

---

**NOTE:** To get the best native floating-point performance on 2806x for math routines, consider using the *C28x FPU Fast RTS Library* ([SPRC664](#)). The C28x Fast RTS is a collection of optimized floating-point math functions for C programmers of the C28x with floating-point unit. Designers of computationally intensive real-time applications can achieve execution speeds considerably faster than what are currently available without having to rewrite existing code. The functions listed in the features section are specifically optimized for the C28x + FPU controllers.

---

**NOTE:** To enable the compiler to generate native FPU instructions, you should indicate in Code Composer Studio that you have a C28x device with a floating-point unit. To do this, use the following compiler switches:

```
-v28 --float_support=fpu32
```

In Code Composer Studio, the `fpu32` switch is under the runtime model compiler options. You cannot mix code built without the `--float_support=fpu32` switch with code built with it. This is because the compiler calling conventions changed for floating-point numbers. If you try to mix the two, the linker issues an error indicating the object files are not compatible. If you receive this error, check that all libraries have been built with the same switch. In particular the runtime support library that comes with the compiler must be correct. When compiling with `--float_support=fpu32` use the *rts2800\_fpu32.lib* or *rts2800\_fpu32\_eh.lib*.

---

**NOTE:** To enable the compiler to generate native VCU instructions, you should indicate that you have a C28x device with a Viterbi, complex math, and CRC unit. To do this, use the following compiler switches:

```
-v28 --vcu_support=vcu32
```

In Code Composer Studio, the VCU switch is under the runtime model compiler options for Code Composer Studio v4.2.2+. For versions prior to v4.2.2, VCU support can be added via the C2000™ command line pattern by inserting the `--vcu_support=vcu0` switch after the `$(command)` tag and prior to the `$(flags)` tag.

### 3.1 Migrating Between IQ\_Math and Native Floating-Point

The following steps must be taken to convert a project written in IQmath format to native floating point.

1. Select `FLOAT_MATH` in the IQmath header file. The header file converts all IQmath function calls to their floating-point equivalent.
2. Convert the floating-point number to an integer when writing a floating-point number into a device register. Likewise, when reading a value from a register, it needs to be converted to float. In both cases, this is done by multiplying the number by a conversion factor. For example, to convert a floating-point number to IQ15, multiply by 32768.0 as shown below.

```
#if MATH_TYPE == IQ_MATH
    PwmReg = (int16)_IQtoIQ15(Var1);
#else // MATH_TYPE is FLOAT_MATH
    PwmReg = (int16)(32768.0L*Var1);
#endif
```

To convert from an IQ15 value to a floating-point value, multiply by 1/32768.0 or 0.00030518.0.

3. Do the following to take advantage of the on-chip floating-point unit:
  - a. Use Code Composer Studio with the C28x codegen tools version 6.0 or later.
  - b. Indicate to the compiler that it can generate native C28x floating-point code. To do this, use the `-v28`

```
                --float_support=fpu32
```

 compiler switches. In Code Composer Studio, the `float_support` switch is on the Advanced tab of the compiler options.
  - c. Use the correct run-time support library for native 32-bit floating-point. For C code, this is `rts2800_fpu32.lib`. For C++ code with exception handling, use `rts2800_fpu32_eh.lib`.
  - d. Consider using the *C28x FPU Fast RTS Library* ([SPRC664](#)) to get a performance boost from math functions such as `sin`, `cos`, `div`, `sqrt`, and `atan`. The Fast RTS Library should be linked in before the normal run-time support library.

## 4 Package and Pinout

The two device groups are not pin-compatible. Any application being moved from one to the other requires a new board layout to accommodate the changes in pinout.

## 5 Operating Frequency

The 2802x/2803x devices are available as 60 MHz or 40 MHz devices. The 2806x devices are available as 90 MHz devices. Both device groups have the same power supply and core voltage requirements.

---

**NOTE:** Always refer to the TMS data manual for information regarding any electrical specifications.

---

See the following data manuals for the most recent detailed electrical specifications:

- *TMS320F28027, TMS320F28026, TMS320F28023, TMS320F28022, TMS320F28021, TMS320F28020, TMS320F280200* *Piccolo Microcontrollers Data Manual* ([SPRS523](#))
- *TMS320F28030, TMS320F28031, TMS320F28032, TMS320F28033, TMS320F28034, TMS320F28035* *Piccolo Microcontrollers Data Manual* ([SPRS584](#))
- *TMS320F28069, TMS320F28068, TMS320F28067, TMS320F28066, TMS320F28065, TMS320F28064, TMS320F28063, and TMS320F28062* *DSPs Data Manual* ([SPRS698](#))

## 6 Supply Voltage and Power Sequencing

Supply voltage and power sequencing requirements are identical for both device groups. That is, the  $V_{DDIO}$  and  $V_{DD}$  rail can ramp together.

For details related to power sequencing, see the device-specific data manual:

- *TMS320F28069, TMS320F28068, TMS320F28067, TMS320F28066, TMS320F28065, TMS320F28064, TMS320F28063, and TMS320F28062* *DSPs Data Manual* ([SPRS698](#))

## 7 Memory Map

The memory maps are similar except for the changes described in this section.

### 7.1 Sequential Access Random Access Memory (SARAM)

This section highlights the major differences in the SARAM memory subsystem.

- **Increased amount of SARAM**

On the 2802x, up to 6K x 16 words of SARAM is available. On the 2803x, up to 10K x 16 is available. On the 2806x, up to 50K x 16 words of SARAM is available.

- **Maximum SARAM block size 8K x 16**

The maximum size of an SARAM block is now 8K x 16 (L4 through L8).

- **Additional SARAM blocks**

The 2806x devices have the same size and address SARAM blocks L0 through L3. In addition, the 2806x family adds five larger SARAM blocks L4 through L8.

The additional memory blocks make it easier to partition code and data. If your code uses multiply and accumulate operations (MAC), for example, you will want to partition the opcode and two operands into three different memory blocks. This allows for maximum efficiency.

- **SARAM blocks are not dual-memory mapped**

Memory block L0 is dual mapped into both high memory and low memory on 2802x/2803x devices. On the 2806x, L0 is not mirrored.

- **DMA accessible SARAM**

The L5-L8 memory blocks can be used as a source and/or destination for each of the 6 DMA channels. DMA accesses to L5-L8 are 0 wait. On the 2802x/2803x, DMA was not available.

Table 1 shows SARAM addresses.

**Table 1. SARAM Addresses**

Memory Address <sup>(1)</sup>	2802x Memory Block	2803x Memory Block	2806x Memory Block
0x00 8000 – 0x00 8FFF	L0	L0/L1/L2	L0/L1/L2
0x00 9000 – 0x00 9FFF	Reserved	L3	L3
0x00 A000 – 0x00 BFFF	Reserved	Reserved	L4
0x00 C000 – 0x00 DFFF	Reserved	Reserved	L5
0x00 C000 – 0x00 DFFF	Reserved	Reserved	L6
0x01 0000 – 0x01 1FFF	Reserved	Reserved	L7
0x01 2000 – 0x01 3FFF	Reserved	Reserved	L8
0x3F 8000 – 0x3F 8FFF	L0 Mirror	L0 Mirror/Boot ROM/Vector	Boot ROM/Vector

<sup>(1)</sup> Some SARAM blocks may not be available on some family derivatives. For more information, see the device-specific data sheet.

## 7.2 Flash and OTP

This section highlights the major differences in the Flash and OTP memory subsystem.

**7.2.1** The size and number of sectors has changed and code must be rebuilt accordingly. The exact Flash size as well as sector configuration varies from device to device as shown in Table 2.

**Table 2. Sector Configuration Per Device**

	F28026	F28027	F28027	F28033	F28065	F28069		
	F28022	F28023	F28023	F28032	F28035	F28064		
	F28026	F28020	F28021	F28030	F28031	F28034		
	F28063	F28062	F28067	F28066	F28066	F28066		
	2 Sectors	4 Sectors	4 Sectors	4 Sectors	8 Sectors	8 Sectors	8 Sectors	8 Sectors
	4K X 16	4K X 16	8K X 16	4K X 16	4K X 16	8K X 16	8K X 16	16K X 16
Total	8K X 16	16K X 16	32K X 16	16K X 16	32K x 16	64K x 16	64K x 16	128K x 16

### 7.2.2 Flash Access Time

For a given frequency, the access time of the Flash is the same for the 2802x/2803x and 2806x. Access times for clock speeds greater than 60 MHz will be published in the device-specific data manual.

**NOTE:** Always refer to the device-specific data manual timing information.

### 7.2.3 Entry Point Into Flash and CSM Password Locations

On both device groups, the boot ROM entry point and code security module password locations are located at the highest addresses of sector A.

### 7.2.4 Entry Point Into OTP

On both device groups, the boot ROM entry point into the OTP is the first address within the OTP.

### 7.2.5 Flash Programming

The method for programming the device remains the same. TI supplies a Flash application programming interface (API) per device that is used as the basis of all programming solutions. The Flash API and programming algorithms for the F2802x/F2803x devices cannot be used on the F2806x. New Flash APIs are required to program these devices; however, the Flash API function prototypes remain compatible.

---

**NOTE:** The Flash API includes timing critical delay loops. These loops should always be run from 0 wait-state memory in order to be timing accurate.

---

### 7.3 Boot ROM

The boot ROM loaders are nearly identical on both device groups. This section highlights the differences. Some of the enhancements found on the boot ROM include:

- The boot ROM has grown to 32Kw
- ROM API tables have been added to support the Flash API symbols library.
- Floating-point math tables have been added in addition to the IQmath tables.

---

**NOTE:** The memory locations of the IQmath tables has shifted. Make sure to use the new addresses in your linker command file as shown in the *C2806x C/C++ Header Files and Peripheral Examples*, which is included in the *controlSUITE zip package - extract to the C drive - v1.9.1* ([SPRCA85](#)).

---

#### 7.3.1 Boot ROM Memory Location

On 2802x/2803x devices as well as the 2806x devices, the boot ROM reserved memory is the first 80 words starting at 0x002. Take care not to allocate code or data to these memory locations until bootloading is complete.

OTP\_KEY is located at address 0x3D7BFB on 2806x devices compared to 0x3D7BFE on 2803x devices, and OTP\_BMODE is located at address 0x3D7BFE on 2806x devices compared to 0x3D7BFF on 2803x devices.

#### 7.3.2 Boot-Mode Selection

The boot mode selection for 2806x devices has some minor differences compared to the 2802x/2803x devices.

On the 2802x/2803x devices, the check value for OTP\_KEY is 0x55AA, while the check value on 2806x devices is instead 0x005A.

#### 7.3.3 Bootloaders

The bootloaders for 2806x devices operate the same as the bootloaders for the 2802x/2803x.

#### 7.3.4 Boot ROM Math Tables

The 2806x has selected IQmath functions programmed into ROM, similar to 2802x/2803x devices. To take full advantage of the 2806x FPU, the 2806x ROM also includes floating-point math tables used by the *C28x FPU Fast RTS Library* ([SPRC664](#)).

For more information on the boot ROM, see the following reference guides:

- *TMS320x2802x Piccolo Boot ROM Reference Guide* ([SPRUFN6](#))
- *TMS320x2803x Piccolo Boot ROM Reference Guide* ([SPRUGO0](#))
- *TMS320x2806x Piccolo Technical Reference Manual* ([SPRUH18](#))

## 8 Clocks and System Control

This section describes changes that affect device clocking and system control. This includes new and renamed registers, pin functionality, new logic, and other enhancements. For more information on system control, see the following reference guides:

- *TMS320F2803x Piccolo System Control and Interrupts Reference Guide* ([SPRUGL8](#))
- *TMS320F2802x/TMS320F2802xx Piccolo System Control and Interrupts Reference Guide* ([SPRUFN3](#))
- *TMS320x2806x Piccolo Technical Reference Manual* ([SPRUH18](#))

## 8.1 Register Changes

Table 3 shows a summary of registers that were added, renamed, or modified. The following sections describe changes that were made. This section does not include the new DMA registers as they can be considered as all new.

**Table 3. New, Updated, and Removed Registers**

Register	Change	Description
PCLKCR0	Updated	Added support for additional SCI and McBSP clock enables and removed LIN clock enable
PCLKCR1	Updated	Added support for additional ePWM, eCAP, and eQEP clock enables
PCLKCR2	New	Support for HRCAP clock enables
PCLKCR3	Updated	Added support for USB and DMA clock enables
PLLCR	Updated	Added bit adds support for more PLL clocking ratio settings
PLL2CTL	New	PLL2 Configuration Register
PLL2MULT	New	PLL2 Multiplier Register
PLL2STS	New	PLL2 Lock Status Register
SYSClk2CNTR	New	SYSClk2 Clock Counter Register
EPWMCFG	New	ePWM DMA/CLA Configuration Register

## 8.2 PLL Multiplier

On 2802x/2803x, the PLL multiplier can be set anywhere from 0 to 12 – this multiplier is controlled by the last 4 bits (DIV) of the PLL Control Register (PLLCR). On the 2806x, the PLL multiplier can be set anywhere from 0 to 16 – this multiplier is controlled by the last 5 bits (DIV) of the PLL Control Register (PLLCR). Along with the PLL divider controlled by bits (DIVSEL) in the PLL Status Register (PLLSTS), the PLL multiplier sets the PLL clocking ratio.

## 8.3 Peripheral Clock Enable Registers

Due to new peripherals and additional instances of old peripherals, the registers to enable and disable the clocks to individual peripherals have been updated (PCLKCR0, PCLKCR1, and PCLKCR3). There is an additional register to enable and disable the clocks to the HRCAP: Peripheral Clock Control Register 2 (PCLKCR2).

## 8.4 SYSClk2 Control

On 2806x, an additional System Clock (SYSClk2) has been added as a clock source to support the HRCAP and new USB modules. An additional PLL (PLL2) is used for SYSClk2. Several registers have been added to configure this new clock.

The PLL2 Configuration Register (PLL2CTL) enables and disables PLL2 and selects the clock source – by default, X1 is selected. The last 4 bits (PLL2MULT) of the PLL2 Multiplier Register (PLL2MULT) allow PLL2 to be set to any multiplier value from 0 to 15 – this is similar to the functionality of PLLCR in relation to the PLL. A read-only status bit (PLL2LOCKS) is also provided in the PLL2 Lock Status Register (PLL2STS) to determine whether the PLL2 has locked.

SYSClk2 also has a free running counter that can be read by software to help determine the approximate frequency SYSClk2 is running. This counter can be read from the COUNT field in the SYSClk2 Clock Counter Register (SYSClk2CNTR).

## 8.5 Enhanced Pulse Width Modulator (ePWM) DMA/CLA Configuration

To support the addition of the DMA module on the 2806x, a new ePWM DMA/CLA Configuration Register (EPWMCFG) has been added. Bit 0 (CONFIG) of this register controls whether ePWM blocks are connected to the DMA bus or to the CLA bus.

## 9 Peripherals

New peripherals have been added and others have been updated. This section briefly describes the changes. For an overview of all peripherals available, see the *TMS320x28xx, 28xxx DSP Peripheral Reference Guide* ([SPRU566](#)).

### 9.1 New Peripherals

The 2806x devices include new peripherals that are not available on the 2802x/2803x devices.

#### 9.1.1 Direct Memory Access (DMA)

The direct memory access module provides a hardware method of transferring data between peripherals and/or memory without intervention from the CPU, thereby, freeing up bandwidth for other system functions. Additionally, the DMA has the capability to orthogonally rearrange the data as it is transferred as well as ping-pong data between buffers. These features are useful for structuring data into blocks for optimal CPU processing.

The DMA module is an event-based machine requiring a peripheral interrupt trigger to start a transfer. The interrupt trigger source for each of the six DMA channels can be configured separately and each channel contains its own independent PIE interrupt to let the CPU know when a DMA transfers has either started or completed. Major features of the DMA are:

- Six channels. Each channel has its own interrupt in the PIE vector table.
- Trigger sources include:
  - Analog-to-digital converter (ADC) interrupts 1 and 2
  - McBSP transmit and receive
  - External interrupts 1-3
  - CPU timers
  - ePWM1-6 start of conversion (SOCA, SOCB)
  - USB endpoints transmit and receive
  - Software
- Data sources and destinations: L5-L8 SARAM, ADC result registers, McBSP transmit and receive registers, ePWM registers.
- Word size can be configured for x16 or x32 bits.

For more information, see the *TMS320x2806x Piccolo Technical Reference Manual* ([SPRUH18](#)).

#### 9.1.2 Viterbi, Complex Math, and CRC Unit (VCU)

The C28x+VCU enhances the processing power of C2000 devices by adding additional assembly instructions to target complex math, Viterbi decode, and CRC calculations. The VCU instructions accelerate many applications, including the following:

- Orthogonal frequency-division multiplex (OFDM) used in the PRIME and G3 standards for power line communications
- Short-range radar complex math calculations
- Power calculations
- Memory and data communication packet checks (CRC)

Major VCU features include:

- Instructions to support cyclic redundancy checks (CRCs), which is a polynomial code checksum.
- Instructions to support a flexible software implementation of a Viterbi decoder
- Complex math arithmetic unit
- Independent register space

For more information, see the *TMS320x2806x Piccolo Technical Reference Manual* ([SPRUH18](#)).

### 9.1.3 Multichannel Buffered Serial Port (McBSP)

The McBSP is highly configurable and supports a variety of application interfaces. The McBSP on the 2806x is similar to that on the TMS320x2832x/2833x devices. For more information, see:

- *TMS320F28069, TMS320F28068, TMS320F28067, TMS320F28066, TMS320F28065, TMS320F28064, TMS320F28063, and TMS320F28062 DSPs Data Manual (SPRS698)*
- *TMS320x2806x Piccolo Technical Reference Manual (SPRUH18)*

### 9.1.4 Universal Serial Bus (USB)

The 2806xU parts include a USB 2.0 compliant USB controller and PHY. The USB peripheral supports full-speed operation as a device and both low and full speed in host operating modes, but does not support high speed or on-the-go (OTG) operations. TI provides drivers for the USB controller as well as a protocol stack free of charge in our *controlSuite* software package. Both the USB controller and its corresponding software package are very similar to that of the Stellaris family of microcontrollers, so migration of USB applications between the two platforms requires minimal effort.

For more information, see:

- *TMS320x2806x Piccolo Technical Reference Manual (SPRUH18)*
- *TMS320F2806xU USB Software Guide*, which is included in the *controlSUITE* zip package - extract to the C drive ([SPRCA85](#))

## 9.2 Control Law Accelerator (CLA)

The 2806x CLA is similar to the CLA in 2803x devices. The CLA register sets are identical on the two device groups: the memory addresses of the CLA control registers and the CLA message RAM are also identical. Some enhancements of the 2806x CLA include:

- The 2806x CLA can operate at up to 80 MHz (2803x CLA can operate up to 60 MHz).
- L0 RAM is connected to the CLA as CLA data RAM2 for the 2806x. L1, L2, and L3 RAM are connected to the CLA as CLA data RAM0, CLA data RAM1, and CLA program RAM as in 2803x devices.
  - The CLA Memory and Clock Configuration Register (MMEMCFG) has been updated with an additional bit to enable access for the CLA data RAM2.
  - Additional bits have been added to the MMEMCFG register to allow CPU write access (in addition to read) to each CLA data RAM.
- The 2806x CLA core is DMA accessible.
- In addition to the ADC result registers, comparator registers, and the ePWM + HRPWM registers accessible by the 2803x CLA, the 2806x CLA has direct access to the eCAP and eQEP registers.
  - The 2806x CLA Peripheral Interrupt Source Select Register (MPISRCSEL1) bit definitions have been updated correspondingly to include interrupt sources from the eCAP and eQEP modules.

## 9.3 High-Resolution Capture (HRCAP)

The 2806x HRCAP module captures the width of pulses with a typical resolution of hundreds of picoseconds and performs both conventional and high-resolution delta time measurements.

Uses for the HRCAP include:

- Capacitive touch applications
- High-resolution period and duty cycle measurements of pulse train cycles
- Instantaneous speed measurements
- Instantaneous frequency measurements
- Reading the feedback across an isolation boundary
- Distance/sonar measurement and scanning

## 9.4 Enhanced Control Peripherals

The eCAP, ePWM, HRPWM, and eQEP modules remain functionally the same. The register sets are identical on the two device groups: the memory addresses of the registers for each instance of peripheral are also identical. For example, ePWM1 registers on 2802x/2803x are at the same location as ePWM1 registers on 2806x. In addition, the interrupt vector location in the PIE vector table is identical. New interrupt vectors have been added to the PIE vector table for the additional peripheral instances (i.e., eCAP3, eQEP2, etc.). In addition, the modules have these added features:

- 2806x devices have three eCAP modules (one more than 2802x/2803x devices)
- 2806x devices have two eQEP modules (one more than 2802x/2803x devices)
- 2806x devices have up to 8 ePWM modules and 8 HRPWM-capable channels (one more than 2802x/2803x devices)
- 2806x devices have 4 HRCAP modules (as discussed in [Section 9.3](#))
- On 2806x devices, the eCAP and eQEP modules are CLA accessible.
- On 2806x devices, the ePWM and HRPWM modules are DMA accessible.

## 9.5 Communication Peripherals

The serial communications interface (SCI), serial peripheral interface (SPI), and inter-integrated circuit (I2C) modules remain functionally the same. The register sets are identical on the two device groups. The memory addresses of the registers for each instance of peripheral are also identical. For example, the SCI-A registers on 2802x/2803x are at the same location as SCI-A registers on 2806x. In addition, the interrupt vector location in the PIE vector table is identical. New interrupt vectors have been added to the PIE vector table for the additional peripheral instances (i.e., SCI-B). Code that has been written for the I2C, SCI, SPI, on the 2802x/2803x can be directly targeted for the 2806x.

The 2806x eCAN is functionally the same and the register sets are identical to the eCAN on 2803x devices. 2802x devices do not have an eCAN module.

- In addition to I2C, SCI, SPI, and eCAN, some 2806x devices have McBSP and USB modules (as discussed in [Section 9.1](#)).
- 2806x devices have two SCI modules (one more than 2802x/2803x devices).
- 2806x devices do not have a LIN module.

## 9.6 Analog Peripherals

The analog peripherals remain functionally the same for both device groups, with only one or two minor additions on the 2806x due to the additional ePWM modules, etc. available as triggers for the ADC, as well as the addition of the DMA:

- An extra divider to the ADCCLK has been added in the ADCCTL2 register. This bit (CLKDIV4EN) adds SYSCLK/4 as an option for the ADC clock to CPU clock ratio for the 2806x.
- Additional ADC start-of-conversion (SOC) triggers have been added on the 2806x to allow for triggering from the additional ePWM8/HRPWM8 channel.
- The 2806x ADC result registers are DMA accessible as well as CLA accessible. Reads of these registers by the CLA, DMA, and CPU are independent of each other and can all occur simultaneously without issue.
- An ADCNONOVERLAP bit in the ADCCTL2 register resolves the ADC first sample issue observed on 2802x and 2803x Rev. 0 silicon.

## 9.7 General-Purpose I/O (GPIO)

The GPIO multiplexing scheme is the same as the 2802x/2803x devices. Additional registers have been added to support the additional GPIOs on the device. Register changes are shown in [Table 4](#).

**Table 4. New, Updated, and Removed Registers**

Register	Change	Description
GPACTRL2	New	GPIO A Control Register 2 (Special Functions)
GPBQSEL2	New	GPIO B Qualifier Select 2 Register (GPIO 48 to 63)
GPBMUX2	New	GPIO B Mux 2 Register (GPIO48 to 63)

### 9.7.1 GPIO Ports and MUX

2806x has additional GPIO pins, but the same ports as 2802x/2803x. To support these added GPIO pins, additional pins have been assigned to port B. On 2803x devices, Port B consists of only GPIO32-GPIO44, and on 2802x devices, Port B consists of only GPIO32-GPIO38. On 2802x, GPIO8 to GPIO11, GPIO13 to GPIO15, GPIO20 to GPIO27, GPIO30 to GPIO31, and GPIO39 to GPIO44 are not supported. On 2806x devices, Port B now consists of GPIO32-GPIO63 – because of this change, the GPBMUX2 register has been added to control the MUX-ing of GPIO48 to 63.

### 9.7.2 GPIO Qualification

On 2806x devices, the type of qualification required for input signals on GPIO0-GPIO63 can be specified just as in the 2802x/2803x. This qualification applies whether the pin is configured as a GPIO or peripheral. Because additional pins have been added to port B, the GPBQSEL2 register has been added to select qualifier values for GPIO48 to 63.

### 9.7.3 GPIO Special Functions

The 2806x has one additional register (GPACTRL2) that handles some special functions not supported on the 2802x/2803x devices. This register handles USB pins outside of normal GPIO MUX logic; it controls whether the pins will be used as GPIO or USB and enables/disables the USBPHY. USB is not present in 2802x/2803x devices.

## 10 Interrupts

Changes to interrupts include updates to the peripheral interrupt expansion (PIE) module.

### 10.1 Peripheral Interrupt Expansion (PIE) Module

The functionality of the PIE module and of the PIE configuration registers remains the same on the 2806x as on 2802x/2803x devices. The PIE vector table has been updated to accommodate the interrupts issued by the new peripheral blocks such as DMA, USB, McBSP, and the floating point unit overflow and underflow flags, and the additional eQEP, eCAP, and HRCAP peripherals on the 2806x.

## 11 Errata Fixes

The 2802x/2803x errata that have been fixed on the 2806x include:

- ADC first sample error
- Reads of addresses 0x3D7FF8-0x3D7FFF activates the CSM, locking the device
- ADC Revision Register (ADCREV) Limitation workaround

## 12 References

- TMS320F28069, TMS320F28068, TMS320F28067, TMS320F28066, TMS32028065, TMS320F28064, TMS320F28063, TMS320F28062 *Piccolo MCU Silicon Errata* ([SPRZ342](#))
- TMS320F28030, TMS320F28031, TMS320F28032, TMS320F28033, TMS320F28034, TMS320F28035 *Piccolo MCU Silicon Errata* ([SPRZ295](#))
- TMS320F28027, TMS320F28026, TMS320F28023, TMS320F28022, TMS320F28021, TMS320F28020, TMS320F280200 *Piccolo MCU Silicon Errata* ([SPRZ292](#))
- TMS320C28x FPU Primer ([SPRAAN9](#))
- TMS320C28x DSP CPU and Instruction Set Reference Guide ([SPRU430](#))
- TMS320C28x Floating Point Unit and Instruction Set Reference Guide ([SPRUEO2](#))
- TMS320x2803x *Piccolo Control Law Accelerator (CLA) Reference Guide* ([SPRUGE6](#))
- TMS320x2806x *Piccolo Technical Reference Manual* ([SPRUH18](#))
- C28x FPU Fast RTS Library ([SPRC664](#))
- TMS320F28027, TMS320F28026, TMS320F28023, TMS320F28022, TMS320F28021, TMS320F28020, TMS320F280200 *Piccolo Microcontrollers Data Manual* ([SPRS523](#))
- TMS320F28030, TMS320F28031, TMS320F28032, TMS320F28033, TMS320F28034, TMS320F28035 *Piccolo Microcontrollers Data Manual* ([SPRS584](#))
- TMS320F28069, TMS320F28068, TMS320F28067, TMS320F28066, TMS320F28065, TMS320F28064, TMS320F28063, and TMS320F28062 *DSPs Data Manual* ([SPRS698](#))
- TMS320x2802x *Piccolo Boot ROM Reference Guide* ([SPRUFN6](#))
- TMS320x2803x *Piccolo Boot ROM Reference Guide* ([SPRUGO0](#))
- TMS320F2802x/TMS320F2802xx *Piccolo System Control and Interrupts Reference Guide* ([SPRUFN3](#))
- TMS320F2803x *Piccolo System Control and Interrupts Reference Guide* ([SPRUGL8](#))
- TMS320x28xx, 28xxx *DSP Peripheral Reference Guide* ([SPRU566](#))
- TMS320F2806x *USB Software Guide*, which is included in the *controlSUITE* zip package - extract to the C drive - v1.9.1 ([SPRCA85](#))

---

## Revision History

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

Changes from Original (March 2011) to A Revision	Page
• Update was made in <a href="#">Section 5</a> .....	5

---

## 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](http://www.ti.com/legal/termsofsale.html)) or other applicable terms available either on [ti.com](http://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  
Copyright © 2020, Texas Instruments Incorporated