Application Report
SPRAAR8–November 2007

Migrating from TMS320DM6446 to TMS320DM6437

Jon Bradley

ABSTRACT

This application report describes device considerations for migrating a design based on a TI TMS320DM6446 Digital Media System-on-Chip (SoC) to one based on a TI TMS320DM6437 Digital Media Processor (DMP). These two devices have many similarities; they both contain the TMS320C64x+™ DSP CPU core, feature video front- and back-end processing capability, and a similar mixture of memory and other peripherals useful in a system environment. The TMS320DM6446 also contains an ARM CPU core, however, the TMS320DM6437 does not. This document describes the details of the considerations of concern for performing this migration.

Note that since this document describes migration from a TMS320DM6446 device to a TMS320DM6437, familiarity with the TMS320DM6446 device and its documentation is assumed.

You can find the documentation for the TMS320DM6446 and the TMS320DM6437 referenced in this migration guide on the TI website located in the device-specific product folders at:

http://focus.ti.com/docs/prod/folders/print/tms320dm6437.html

http://focus.ti.com/docs/prod/folders/print/tms320dm6446.html

Contents

List of Figures

List of Tables

Trademarks

TMS320C64x+, C64x+ are trademarks of Texas Instruments.
All other trademarks are the property of their respective owners.

1 Basic Feature Comparison

Table 1 shows a comparison of the basic features of the TMS320DM6446 and the TMS320DM6437. The remainder of this document presents a comparison of these features in greater detail, and also provides references to the appropriate documentation for further information.

Table 1. Basic DM6446/DM6437 Feature Comparison

Feature			DM6446	DM6437
CPU	DSP		C64x+	C64x+
	ARM		ARM926EJ-S	
Speeds	DSP		600 MHz	600/500/400 MHz
	ARM		300 MHz	
Endianness			Little	Little
Memory	DSP			
		Cache: L1P	32K bytes	32K bytes
		Cache: L1D	80K bytes	80K bytes
		Cache: L2	64K bytes	128K bytes
		ROM		64K bytes
	ARM			
		Data/Program RAM	16K bytes	
		Data/Program ROM	8K bytes	
		Program Cache	16K bytes	
		Data Cache	8K bytes	
Peripherals	Video Ports		VPFE, VPBE	VPFE, VPBE
	Video/Imaging Coprocessor		1	
	DDR2 EMIF		1	1
	Asynchronous EMIF		1	1
	EDMA		EDMA 3.0 - 64 ch.	EDMA 3.0 - 64 ch.
	PCI			32 bits, v2.3
	Serial Ports		1 x ASP	
				1 x McASP
				2 x McBSP
	SPI		1	
	PLL		3	2
	UARTs		3	2
	Timers		2 × 64 bits	2 × 64 bits
	Watchdog Timer		1 x 64 bits	1 x 64 bits
	EMAC		1	1
	USB		1	
	ATA/ATAPI		1	
	MMC/SD		1	
	I2C		1	1
	HECC			1
	VLYNQ		1	1
	HPI		16 bit	16 bit
	PWM		3	3
	GPIO		71	111
Power Supplies			3	3
Packages			361-pin BGA	361/376-pin BGA

2 CPU Core Considerations

The DM6446 contains an ARM CPU core as well as a DSP CPU core. The DM6437 contains only a DSP CPU core. The following sections discuss considerations of the different CPU cores provided on these two devices.

2.1 DSP CPU Core Considerations

The TMS320DM6446 and the TMS320DM6437 utilize the same TMS320C64x+ core DSP CPU; therefore, code written for the DSP CPU on the DM6446 functions in the same fashion on the DM6437. Note, however, that there are differences in other aspects of these devices that affect how code operates, such as functionality of peripheral modules and differences in memory map, and these differences must be accounted for when migrating applications from the DM6446 to the DM6437. These additional differences are discussed in detail in the remainder of this document.

2.2 ARM CPU Core Considerations

Since the DM6446 contains an ARM CPU and a DSP CPU, and the DM6437 contains only the DSP CPU, all of the processing in the DM6437 is performed in the DSP CPU. Note that basic control code for the ARM CPU on the DM6446, if written in C, can often be recompiled on the DSP CPU on the DM6437, and used without major modifications.

2.3 CPU Clock Speeds

The DM6446 and DM6437 can be operated at a range of clock speeds to accommodate a variety of different performance requirements. In addition, the DM6437 is also available in three different speed versions in order to address different system speed and cost tradeoffs.

Table 2 shows a summary of the different speed versions of the DM6446 and DM6437.

Table 2. Available Performance Versions of the DM6446 and DM6437

DM6446		DM6437
ARM CPU	DSP CPU	DSP CPU
300 MHz (3.37 ns cycle time)	600 MHz (1.67 ns cycle time)	600 MHz (1.67 ns cycle time) 500 MHz (2.00 ns cycle time) 400 MHz (2.50 ns cycle time)

Note that the CPU speeds listed in Table 2 are the maximum operating speeds for the CPUs in question. Actual operating speeds are typically chosen to match clocking requirements of the system as a whole. For example, the 600 MHz DSP CPUs are commonly operated at a frequency of 594 MHz, with a device input clock of 27 MHz. This results in clocks for other parts of the devices to match with requirements for their applications, such as video processing.

Also note that the power supply voltage requirements are different on these devices for some of the different speed versions. For detailed information regarding power supply voltage requirements for the DM6446 and DM6437, see Section 10 of this document . In addition, see Section 4.7 of this document for information regarding operation of the PLL/clock generators on these two devices, and Section 8 for information regarding initialization of the PLL/clock generators on these two devices.

For additional detailed information regarding performance, timing requirements, and characteristics for the DM6437, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345).

2.4 Endianness Considerations

The DM6446 and the DM6437 function in little-endian operating mode; therefore, there are no endianness considerations when migrating an application from the DM6446 to the DM6437.

3 Internal Memory Comparisons

The TMS320DM6446 and the TMS320DM6437 feature on-chip internal memories, allowing efficient handling of varied partitions of internal program and data information. Both devices feature several different types of cache memory, allowing significant flexibility in using this memory to enhance algorithm performance. These devices also provide an on-chip ROM, which contains the bootloader program. Since there are some differences between the memory architectures on the two devices, some software modifications are required when migrating applications from the DM6446 to the DM6437. Note that since the DM6446 contains both ARM and DSP CPU cores, memory usage is partitioned between the two CPUs; therefore, certain areas of internal memory, although accessible to both CPUs, are typically used more by one CPU than the other.

Table 3 shows a comparison of the DM6446 and DM6437 DSP internal memory.

Table 3. DSP Internal Memory Comparison

Memory Type	DM6446	DM6437
L1P Program Memory	32K-bytes RAM/Cache (direct mapped), flexible allocation	32K-bytes RAM/Cache (direct mapped), flexible allocation
L1D Data Memory	80K-bytes RAM/Cache (2-way set associative), flexible allocation	80K-bytes RAM/Cache (2-way set associative), flexible allocation
L2 RAM Memory	64K-bytes Unified Mapped RAM/Cache (4-way set associative), flexible allocation	80K-bytes RAM/Cache (2-way set associative), flexible allocation
ROM		64K-bytes

Again, note that although the data and program RAM and ROM memory described in Table 3, can also be accessed by the DSP CPU on the DM6446, it is mainly for use by the ARM CPU. The cache memory is for exclusive use of the ARM CPU.

The DM6446 ARM CPU also has its own associated memory on board the device. This memory consists of 16K bytes of RAM which can be used for instructions or data, and 8K bytes of ROM which can also be used for instructions or data. The DM6446 ARM memory also includes 16K bytes of instruction cache and 8K bytes of data cache. These cache memories are used by the CPU to enhance instruction and data handling, but they are not directly accessible within the memory space of the device.

Table 4 presents a description of the ARM CPU memory on the DM6446 device.

Table 4. DM6446 ARM CPU Internal Memory

Memory Type	Size
Data/Program RAM	16K bytes
Data/Program ROM	8K bytes
Program Cache	16K bytes
Data Cache	8K bytes

Again, note that although the data and program RAM and ROM memory described in Table 4 can also be accessed by the DSP CPU on the DM6446, it is mainly for use by the ARM CPU. The cache memory is for exclusive use of the ARM CPU.

For additional detailed information regarding use of the DM6437's internal memory, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345) and the *TMS320C64x+ DSP Cache User's Guide* (SPRU862). For additional information regarding the bootloader on the DM6437, see Section 6 of this document.

4 Peripherals

The TMS320DM6446 and TMS320DM6437 feature a wide variety of peripheral modules which are useful in many different system environments. This section presents a comparison of the peripheral offerings on these two devices.

4.1 Video Ports

The DM6446 and DM6437 feature video processing peripherals which can accept video input, and can generate video output. The video processing capabilities on these two devices is implemented in the video processing subsystem (VPSS), which is partitioned into two subsections - an input video processing front end (VPFE), and an output video processing back end (VPBE).

The VPSS on these two devices is based on the same peripheral modules; therefore, migration of an application utilizing the VPSS from the DM6446 to the DM6437 requires little, if any, modification of hardware or software.

There are, however, a few differences between these peripheral modules on the two devices. Some of the main differences in usage between these peripheral modules are signal locations within the device pinouts, and the pin multiplexing configuration settings.

In addition to these basic differences, there are also three register bits in the VPBE whose location has been moved. These three bits were moved from three different registers into the MISCCTL register for convenience of access. Table 5 describes the details of the changes in the locations of these three bits.

Table 5. VPBE Bit Location Changes

DM6446 Register Name [bit number]	Bit Name	DM6437 Bit Function	DM6437 Location of DM6446 Bit
VIDWINMD[15]	VFINV	Reserved	MISCCTL[11]
OSDWIN0MD[14]	ATN0E	Reserved	MISCCTL[9]
OSDWIN1MD[14]	ATN1E	Reserved	MISCCTL[8]

In addition to the items discussed above, there are also four DM6446 device errata VPSS advisories which are applicable to the DM6437. Two of these advisories (1.3.8 and 1.3.10) have been fixed, and the other two (1.3.9 and 1.3.12) have workarounds on the DM6437.

Table 6 summarizes these advisories on the DM6446 and the DM6437.

Table 6. DM6446 and DM6437 VPSS Advisory Summary

DM6446 Advisory	DM6437 Status	DM6437 Advisory	Comments
1.3.8	Fixed	X	
1.3.9	Workarounds	1.2.9	Set MISCCTL register VIDOVRLMODE bit [1]
1.3.10	Fixed	X	
1.3.12	Workarounds	1.2.1	Set MISCCTL register VIDOVRLMODE bit for x1 horizontal zoom [1]

[1] See DM6437 Errata for additional information regarding workarounds.

For additional detailed information regarding the DM6437 VPSS advisories, see the *TMS320DM6437/35/33/31 Digital Media Processor (DMP) Silicon Revisions 1.2, 1.1, and 1.0 Silicon Errata* (SPRZ250).

For additional detailed information regarding use of the video ports on the DM6437, see the *TMS320DM643x DMP Video Processing Front End (VPFE) User's Guide* (SPRU977) and the *TMS320DM643x DMP Video Processing Back End (VPBE) User's Guide* (SPRU952).

4.2 Video/Imaging Coprocessor (VICP)

The DM6446 contains a specialized coprocessor, the video/imaging coprocessor (VICP), which can be used to perform operations that offload many video and imaging processing tasks from the DSP core, making more DSP MIPS available for common video and imaging algorithms. The VICP, however, is not documented in detail other than for providers of algorithms specifically using this peripheral.

The DM6437 does not contain the VICP, therefore, algorithms written for the DM6446 using the VICP need to be rewritten for the DM6437. For these algorithms to be rewritten, you should contact the provider of the algorithms in question.

4.3 External Memory Interfaces (EMIF)

The DM6446 and the DM6437 feature flexible external interfaces which support accessing various types of memories. Both devices support two independent memory interfaces, each architected for specific memory types. These two memory interfaces are the DDR2 memory interface, specifically designed to gluelessly interface to industry-standard DDR2 memories, and the asynchronous memory interface, designed to efficiently interface to a variety of asynchronous memory types. They are similar on both devices making migration of applications from the DM6446 to the DM6437 straightforward. There are, however, some differences in using these peripherals on the two devices. The following paragraphs describe migration of applications using these two interfaces from the DM6446 to the DM6437 device.

4.3.1 DDR2 EMIF

On DM6446 and DM6437, the DDR2 interface uses the same peripheral module; therefore, migration of an application utilizing the DDR2 EMIF between these two devices requires little, if any, modification of hardware or software.

The DDR2 interface uses a 32-bit data bus, and is optimized for use with high-speed, high-density DDR2 memory for storage of programs and large blocks of data. Additionally, both devices also have the capability to use the interface data bus in 16-bit mode, instead of 32-bit mode, if desired, to save pinout connections.

Since the DM6446 and the DM6437 utilize the same DDR2 EMIF peripheral module, the predominant differences in usage between this peripheral module on these two devices are signal locations within the device pinouts, and the pin multiplexing configuration settings. The base address and locations where the memory is accessed, and the location of the memory mapped control registers within each of the device address spaces is the same for both devices.

In addition to the minor differences mentioned above, there are two other significant differences between the DDR2 EMIF on these two devices. First, the DDR2 EMIF peripheral module is clocked by a different clock on the DM6437 than on the DM6446. On the DM6446, the DDR2 EMIF is clocked by the PLL2 SYSCLK2 clock, while on the DM6437, the DDR2 EMIF is clocked by the PLL2 SYSCLK1 clock. This does not affect performance, it only affects which registers must be written to during initialization. Note that these two clocks do not default to the same frequencies following reset; therefore, for identical performance, the control registers for these clocks must be modified on the DM6437 to achieve the same performance as with the DM6437.

The second major difference between the two devices is that the default values after reset of several configuration registers are different between the two devices. Specifically, the SDRCR, SDTIMR, SDTIMR2, and VTPIOCR register default initialization values are different between the DM6446 and the DM6437 (the exact differences are presented in Table 7).

A comparison between the DDR2 EMIF on the DM6446 and the DM6437 is shown in Table 7.

Table 7. Comparison of DDR2 EMIF Features

Feature		DM6446	DM6437
Data Width		32 or 16 bits	32 or 16 bits
Base address in device address space		80000000h	80000000h
Address Range		256 Mbytes	256 Mbytes
Input Clock		PLL2 SYSCLK2	PLL2 SYSCLK1
Register Default Initialization Values	SDRCR	00000300h	00000884h
	SDTIMR	00000000h	356B4B9Bh
	SDTIMR2	00000000h	001DF145h
	VTPIOCR	00000000h	0000001Fh

For additional detailed information regarding use of the DDR2 EMIF on the DM6437, see the *TMS320DM643x DMP DDR2 Memory Controller User's Guide* (SPRU986).

4.3.2 Asynchronous EMIF

On DM6446 and DM6437, the same peripheral module is also used for the asynchronous EMIF; therefore, migration of an application utilizing this interface between these two devices also typically requires only minor modification of hardware or software.

The asynchronous memory interface on the DM6446 and the DM6437 is designed for slower special-purpose memory such as SRAM, NOR, NAND flash memory, ROM, and other asynchronous memory types. Since the DM6446 and the DM6437 utilize the same asynchronous EMIF peripheral module, some of the main differences in usage between this peripheral module on these two devices are signal locations within the device pinouts, and the pin multiplexing configuration settings.

Another important difference in the use of this peripheral module between these two devices is that on the DM6446, an 8- or 16-bit data bus width is supported, while on the DM6437, only an 8-bit data bus width is supported. Accordingly, the memory system configuration may need to be reorganized when migrating an application from the DM6446 to the DM6437.

Additionally, there are differences in the EMIF address space range and partitioning on these two devices. The total accessible asynchronous EMIF memory range on both these devices is divided into four chip select (CS) spaces. Although the starting address of the asynchronous EMIF is 42000000h on both devices, 128 Mbytes of memory can be accessed contiguously on the DM6446. Whereas on the DM6437, only 64 Mbytes can be accessed in the four CS spaces, and the four CS spaces are not contiguous on this device.

In addition, on the DM6446, the EMIF address space is shadowed at location 02000000h in addition to being accessible starting at location 42000000h within the device address space; however, accesses to EMIF memory starting at address 02000000h are for data information only, and not for program execution. This address range is reserved on the DM6437; therefore, data accesses to this range made in a DM6446 application need to be remapped on the DM6437.

There are also some slight differences in the memory mapped asynchronous EMIF control registers between the two devices. On the DM6446, the Revision Code and Status Register (RCSR) is not available, whereas on the DM6437, it is available. Also, the default initialization value of the Asynchronous Wait Cycle Configuration Register (AWCCR) is 10000080h on the DM6446 and F0000080h on the DM6437. Note that this difference in register default initialization values is only in the reserved bits in these registers; therefore, this does not affect device operation in either case.

One final difference in the asynchronous EMIF between the two devices is that the $\overline{\text{CS}}$ outputs do not require pullup resistors on the DM6446, while on the DM6437, they are required in most cases. Specifically, if the EM_CS3, EM_CS4, and EM_CS5 pins are to be connected and used as EMIF chip select signals, external pullup resistors should be applied to these pins to ensure the EMIF chip select signals default to an inactive (high) state immediately following reset. If the EM_CS2 pin is to be connected and used as an EMIF chip select signal, and the AEM[2:0] pins on the device are set to a value other than zero at reset, an external pullup resistor should also be applied to EM_CS2 to ensure this EMIF chip select signal defaults to an inactive (high) state immediately following reset.

Table 8 presents a comparison of the differences between the asynchronous EMIF on the DM6446 and the DM6437.

Table 8. DM6446 and DM6437 Asynchronous EMIF Feature Comparison

Feature	DM6446	DM6437
Data Width	16 or 8 bits [1]	8 bits
Memory Types	Async [2]	Async [2]
Starting address within device address space	42000000h	42000000h[3]
Total Memory Address Space	128 Mbytes	64 Mbytes [3]
Revision Code/Status Register	Not available	Available
AWCCR Register Default Initialization Values	10000080h [4]	F0000080h [4]
Pullups required on $\overline{\text{CS}}$ pins	No	In most cases

[1] Default width is 8 bits

[2] Memory types include, SRAM, flash, etc.

[3] Total memory space is not contiguous

[4] Differences are in reserved bits only

For detailed information regarding use of the asynchronous EMIF on the DM6437, see the *TMS320DM643x DMP Asynchronous External Memory Interface (EMIF) User's Guide* (SPRU984).

4.4 EDMA Controllers

The DM6446 and the DM6437 feature enhanced DMA (EDMA) controllers which can be used to transfer data to and from numerous locations, both on- and off-chip. Both devices support 64 independent channels of EDMA transfers.

The EDMA controllers used on these two devices are based on the EDMA 3.0 peripheral module; however, there are some differences between the peripheral modules on the two devices. Specifically, the DM6437 EDMA implements several advanced features beyond those available on the DM6446. For example, the DM6446 provides two EDMA queues and two transfer controllers, while the DM6437 provides three of each. Also, the default DMA burst size for each transfer controller is fixed on the DM6446, while on the DM6437, the default burst size is programmable. Finally, the DM6437 provides one more transfer completion and error interrupt than the DM6446.

Table 9 presents a comparison of the features of the EDMA controllers on the DM6446 and the DM6437.

Table 9. DM6446 to DM6437 EDMA Comparison

Feature	TC	DM6446	DM6437
Number of Queues		2	3
Number of Transfer Controllers		2	3
Default Burst Size	TC0	Fixed - 16 bytes	Programmable - 16 bytes default
	TC1	Fixed - 32 bytes	Programmable - 32 bytes default
	TC2	TC2 not implemented	Programmable - 64 bytes default
Transfer Completion Interrupts		2 (Shadow Region 0, 1)	3 (Global + Shadow Region 0, 1)
Error Interrupts		3 (Global + TC0 + TC1)	4 (Global + TC0 + TC1 + TC2)

For the DM6446 and the DM6437, the 64 possible EDMA channel synchronization events are predefined to various sources on the device, and the actual synchronization events used to trigger specific EDMA operations are selected from these possible synchronization events. Table 10 presents a comparison of the 64 possible EDMA channel synchronization events available on the DM6446 and the DM6437.

Table 10. EDMA Channel Synchronization Events Comparison

	DM6446		DM6437	
Channel	Event Name	Event Description	Event Name	Event Description
0-1	—	Reserved	—	Reserved
2	XEVT	ASP Transmit Event	XEVT0	McBSP0 Transmit Event
3	REVT	ASP Receive Event	REVT0	McBSP0 Receive Event
4	HISTEVT	VPSS Histogram Event	XEVT1	McBSP1 Transmit Event
5	H3AEVT	VPSS H3A Event	REVT1	McBSP1 Receive Event
6	PRVUEVT	VPSS Previewer Event	HISTEVT	VPSS Histogram Event
7	RSZEVT	VPSS Resizer Event	H3AEVT	VPSS H3A Event
8	IMXINT	VICP Interrupt	PRVUEVT	VPSS Previewer Event
9	VLCDINT	VICP VLCD Interrupt	RSZEVT	VPSS Resizer Event
10	ASQINT	VICP ASQ Interrupt	AXEVTE0	McASP0 Transmit Event Even
11	DSQINT	VICP DSQ Interrupt	AXEVTO0	McASP0 Transmit Event Odd
12	—	Reserved	AXEVT0	McASP0 Transmit Event
13	—	Reserved	AREVTE0	McASP0 Receive Event Even
14	—	Reserved	AREVTO0	McASP0 Receive Event Odd
15	—	Reserved	AREVT0	McASP0 Receive Event
16	SPIXEVT	SPI Transmit Event	—	Reserved
17	SPIREVT	SPI Receive Event	—	Reserved
18	URXEVT0	UART 0 Receive Event	—	Reserved
19	UTXEVT0	UART 0 Transmit Event	—	Reserved
20	URXEVT1	UART 1 Receive Event	—	Reserved
21	UTXEVT1	UART 1 Transmit Event	—	Reserved
22	URXEVT2	UART 2 Receive Event	URXEVT0	UART 0 Receive Event
23	UTXEVT2	UART 2 Transmit Event	UTXEVT0	UART 0 Transmit Event
24	—	Reserved	URXEVT1	UART 1 Receive Event

Table 10. EDMA Channel Synchronization Events Comparison (continued)

	DM6446		DM6437	
Channel	**Event Name**	**Event Description**	**Event Name**	**Event Description**
25	—	Reserved	UTXEVT1	UART 1 Transmit Event
26	MMCRXEVT	MMC Receive Event	—	Reserved
27	MMCTXEVT	MMC Transmit Event	—	Reserved
28	I2CREVT	I2C Receive Event	ICREVT	I2C Receive Event
29	I2CXEVT	I2C Transmit Event	ICXEVT	I2C Transmit Event
30-31	—	Reserved	—	Reserved
32	GPINT0	GPIO 0 Interrupt	GPINT0	GPIO 0 Interrupt
33	GPINT1	GPIO 1 Interrupt	GPINT1	GPIO 1 Interrupt
34	GPINT2	GPIO 2 Interrupt	GPINT2	GPIO 2 Interrupt
35	GPINT3	GPIO 3 Interrupt	GPINT3	GPIO 3 Interrupt
36	GPINT4	GPIO 4 Interrupt	GPINT4	GPIO 4 Interrupt
37	GPINT5	GPIO 5 Interrupt	GPINT5	GPIO 5 Interrupt
38	GPINT6	GPIO 6 Interrupt	GPINT6	GPIO 6 Interrupt
39	GPINT7	GPIO 7 Interrupt	GPINT7	GPIO 7 Interrupt
40	GPBNKINT0	GPIO Bank 0 Interrupt	GPBNKINT0	GPIO Bank 0 Interrupt
41	GPBNKINT1	GPIO Bank 1 Interrupt	GPBNKINT1	GPIO Bank 1 Interrupt
42	GPBNKINT2	GPIO Bank 2 Interrupt	GPBNKINT2	GPIO Bank 2 Interrupt
43	GPBNKINT3	GPIO Bank 3 Interrupt	GPBNKINT3	GPIO Bank 3 Interrupt
44	GPBNKINT4	GPIO Bank 4 Interrupt	GPBNKINT5	GPIO Bank 4 Interrupt
45	—	Reserved	GPBNKINT4	GPIO Bank 5 Interrupt
46	—	Reserved	GPBNKINT6	GPIO Bank 6 Interrupt
47	—	Reserved	—	Reserved
48	TINT0	Timer 0 Interrupt	TEVTL0	Timer 0 Event Low Interrupt
49	TINT1	Timer 1 Interrupt	TEVTH0	Timer 0 Event High Interrupt
50	TINT2	Timer 2 Interrupt	TEVTL1	Timer 1 Event Low Interrupt
51	TINT3	Timer 3 Interrupt	TEVTH1	Timer 1 Event High Interrupt
52	PWM0	PWM 0 Event	PWM0	PWM 0 Event
53	PWM1	PWM 1 Event	PWM1	PWM 1 Event
54	PWM2	PWM 2 Event	PWM2	PWM 2 Event
55-63	—	Reserved	—	Reserved

For detailed information regarding the use of EDMA on the DM6437, see the *TMS320DM643x DMP Enhanced Direct Memory Access (EDMA3) Controller User's Guide* (SPRU987).

4.5 PCI Interface

While the DM6446 does not feature a PCI interface, the DM6437 does; therefore, this added capability can be utilized in DM6437 systems.

The DM6437 PCI interface is compatible with the industry-standard peripheral component interconnect (PCI) interface version 2.3. This interface allows communication with devices complaint to the PCI Local Bus Specification, revision 2.3, via a 32-bit address/data bus operating at speeds up to 33 MHZ. The PCI interface can operate in either master or slave mode.

For detailed information regarding the use of the PCI interface on the DM6437, see the *TMS320DM643x DMP Peripheral Component Interconnect (PCI) User's Guide* (SPRU985).

4.6 Serial Ports

The DM6446 and DM6437 feature serial port interfaces to provide connectivity to a wide variety of external devices including codecs, communications peripherals, and other processors. The DM6437 features an audio serial port (ASP) interface, while the DM6437 features two multi-channel buffered serial ports (McBSPs) and one multi-channel audio serial port (McASP). The following paragraphs describe migration between these peripheral modules in detail.

4.6.1 ASP Serial Port

The DM6446 features an ASP interface, while the DM6437 does not; therefore DM6446 applications utilizing the ASP requires some modifications when migrating between these two devices. The DM6437 does, however, feature two McBSPs, which provide a superset of the DM6446 ASP functionality. Essentially all of the ASP functionality, and more, is included in the DM6437 McBSP (see Section 4.6.3 of this document for additional information regarding the DM6437 McBSP).

Since the ASP is a peripheral derived from the McBSP, the architecture and programming of the McBSP is quite similar to the ASP, which greatly simplifies migration of applications from the DM6446 to the DM6437.

The DM6446 ASP and the DM6437 McBSP are configured by similar sets of control registers, however, these control registers are located at different base addresses. The base address of the DM6446 ASP control registers is 01E02000h, while the DM6437 McBSP control registers are located at 01D00000h and 01D00800h.

For detailed information regarding the use of the McBSP on the DM6437, see the *TMS320DM643x DMP Multichannel Buffered Serial Port (McBSP) User's Guide* (SPRU943).

4.6.2 McASP Serial Port

While the DM6446 does not feature the McASP peripheral, the DM6437 does; therefore this added capability can be utilized in DM6437 systems.

The DM6437 McASP functions as a general-purpose audio serial port optimized for the needs of multichannel audio applications. The McASP is useful for time-division multiplexed (TDM) stream, inter-integrated sound (I2S) protocols, and intercomponent digital audio interface transmission (DIT). The McASP consists of transmit and receive sections that may operate synchronized, or completely independently with separate master clocks, bit clocks, and frame syncs, and using different transmit modes with different bit-stream formats. The McASP module also includes up to four serializers that can be individually enabled to either transmit or receive.

For detailed information regarding the use of the McASP on the DM6437, see the *TMS320DM643x DMP Multichannel Audio Serial Port (McASP) User's Guide* (SPRU980).

4.6.3 McBSP Serial Ports

While the DM6446 does not feature the McBSP peripheral, the DM6437 has two; therefore, this added capability can be utilized in DM6437 systems. Additionally, the DM6437 McBSP can be used, if necessary, to support the functionality of the SPI serial port featured on the DM6446 (see Section 4.7).

The primary use for the DM6437 McBSP is for audio interface purposes. This McBSP is a specialized version of the McBSP peripheral used on other TI DSPs. The primary audio modes that are supported by the McBSP are the AC97 and IIS modes. In addition to the primary audio modes, the McBSP can be programmed to support other serial formats but is not intended to be used as a high-speed interface.

The McBSP consists of a data path and a control path that connect to external devices. Separate pins for transmission and reception communicate data to these external devices. The DM643x CPU communicates to the McBSP using 32-bit-wide control registers accessible via the internal peripheral bus.

For detailed information regarding the use of the McBSP on the DM6437, see the *TMS320DM643x DMP Multichannel Buffered Serial Port (McBSP) User's Guide* (SPRU943).

4.7 SPI Port

The DM6446 features a dedicated SPI (serial peripheral interface) port, while the DM6437 does not; therefore, DM6446 applications utilizing the SPI require some modifications when migrating between these two devices. However, since the DM6437 contains two McBSP serial ports, a DM6446 application using the SPI can migrate this functionality to the DM6437 by using one of the McBSPs configured in SPI mode. Additionally, while the DM6446 SPI port can only function in master mode, the DM6437 McBSP, when operating as an SPI port, can function in either master or slave mode, providing added capability in a DM6437-based system.

Note that the architecture and programming of the SPI and the McBSP are different; therefore, when the McBSP is used to implement SPI functionality, some software modifications are required. Specifically, the control register set and their functionality are different between the SPI and the McBSP (see also Section 4.6.3 of this document for additional information regarding the DM6437 McBSP).

For detailed information regarding the use of the McBSP on the DM6437, see the *TMS320DM643x DMP Multichannel Buffered Serial Port (McBSP) User's Guide* (SPRU943).

4.8 PLL/Clock Generators

The DM6446 and the DM6437 feature clock generators with PLLs that are used to provide clocks for these devices. While the DM6446 and DM6437 PLLs are slightly different in architecture, their overall features allow them to provide comparable system clocking, which allows easy migration between the two devices.

There are three PLLs on the DM6446 which are used to generate a variety of different clocks for the device from the input clocks. PLL1 can multiply its input clock by a programmable value, and the resultant clock can then be post-divided by 1, 2 or 3. Five system clocks are then generated for the device using fixed dividers of 1, 2, 3, 4, and 6 (note that the divide-by-four clock is reserved). These system clocks are then used to satisfy the clocking requirements for most of the different parts of the DM6446 device. PLL1 also provides the additional feature of being able to use the input clock directly, bypassing the PLL, and divide this by a programmable value between 1 and 32, to produce an additional clock which drives the CLKOUT0 output.

PLL2 can also multiply its input clock by a programmable value, and the resultant clock is then divided by programmable values between 1 and 16 to produce two output clocks. These two clocks are then used to satisfy the clocking requirements of the VPSS and the DDR2 interface on the device. PLL2 also provides the additional feature of being able to use the input clock directly, bypassing the PLL, and divide this by a programmable value between 1 and 32, to produce an additional clock which drives the DDR2 VTP.

Figure 1 and Figure 2 show the architecture of PLL1 and PLL2 on the DM6446.

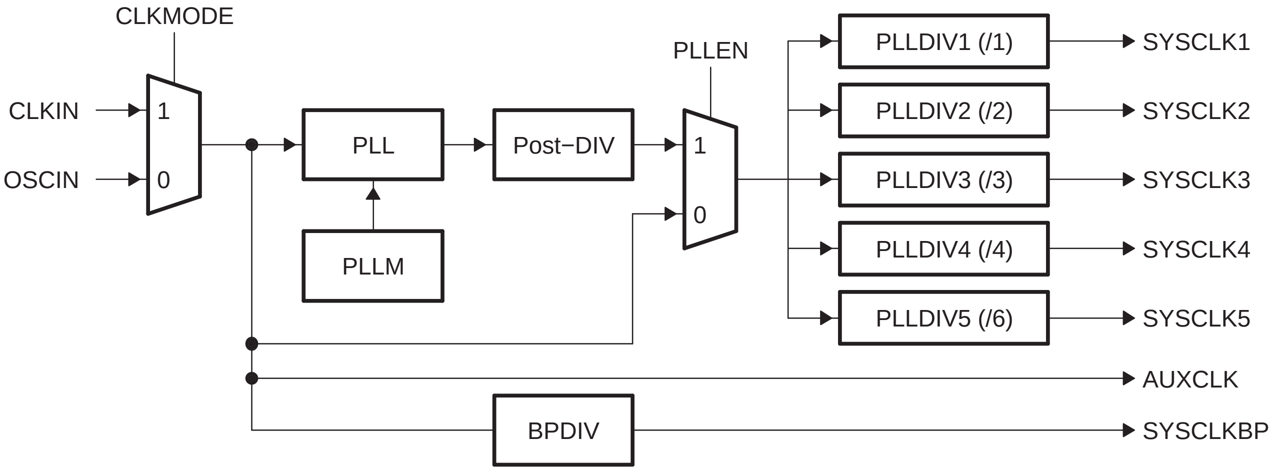

Figure 1. PLL1 Structure in the TMS320DM6446 DMSoC

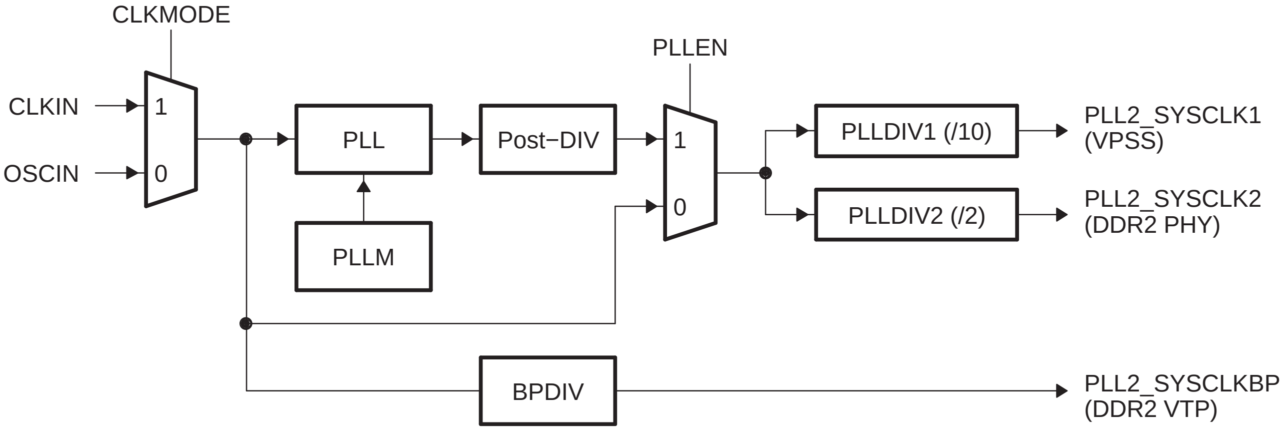

Figure 2. PLL2 Structure in the TMS320DM6446 DMSoC

PLL3 is used to provide clock for the USB interface. Its 24 MHz input clock is multiplied by 2.5 to generate a 60 MHz clock for the USB peripheral module.

On the DM6437, the clock generator has two PLLs — PLL1 and PLL2. Each of these PLLs also supports a number of multiply and divide configurations, allowing the device to generate a number of different clock frequencies to satisfy a wide variety of system requirements.

With PLL1, the input clock may be multiplied by a software programmable value, and the resultant clock is then divided by programmable values between 1 and 32 to produce three system clocks. The three system clocks produced by PLL1 are then used to clock the DSP, the DMA, and the VPFE. Note that when choosing divider values for the three system clocks, a ratio of 1:3:6 must be maintained with their frequencies for proper system operation.

Within PLL1, the input clock is also used directly, bypassing the PLL, and divided by a programmable value between 1 and 32 to produce an additional clock which drives the VPBE. In the same fashion, the input clock is also used directly, bypassing the PLL, and divided by a programmable value between 1 and 32 to generate the observed output clock for the device, CLKOUT0.

PLL2 can also multiply the input clock by a programmable value, and the resultant clock is then divided by programmable values between 1 and 32 to produce two output clocks. These two clocks are then used to satisfy the clocking requirements of the DDR2 interface and the VPBE on the device. PLL2 also provides the additional feature of being able to use the input clock directly, bypassing the PLL, and divide this by a programmable value between 1 and 32, to produce an additional clock which drives the DDR2 VTP.

Figure 3 and Figure 4 show the architecture of PLL1 and PLL2 on the DM6437.

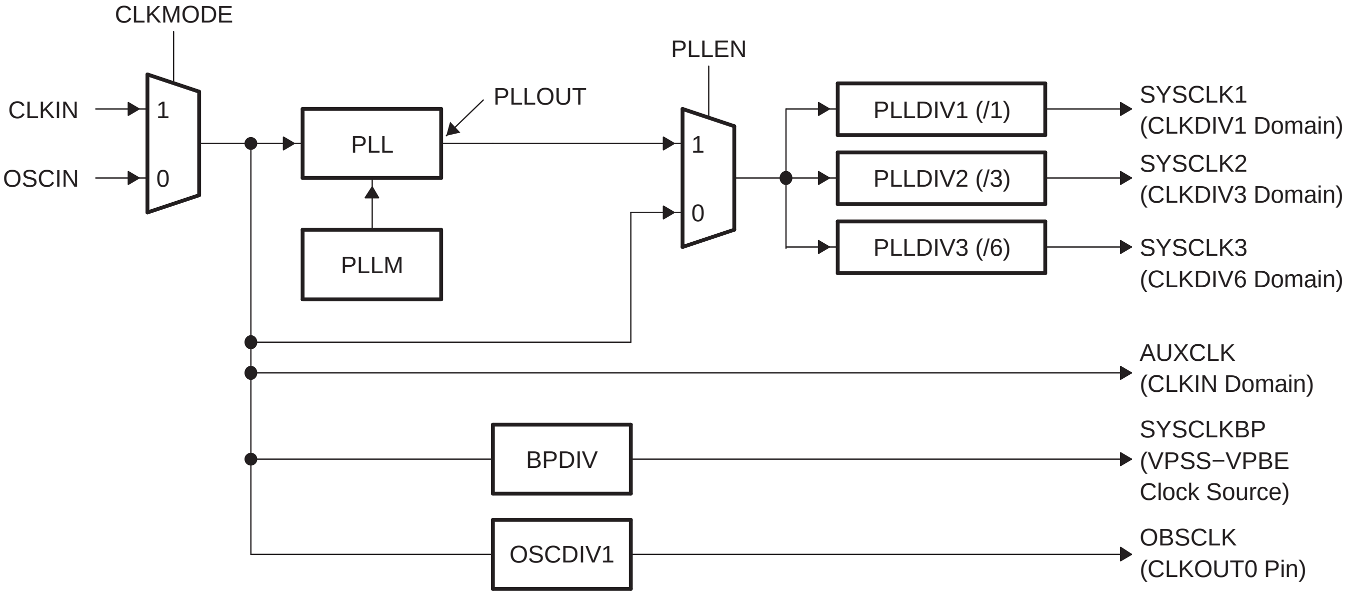

Figure 3. PLL1 Structure in the TMS320DM643x DMP

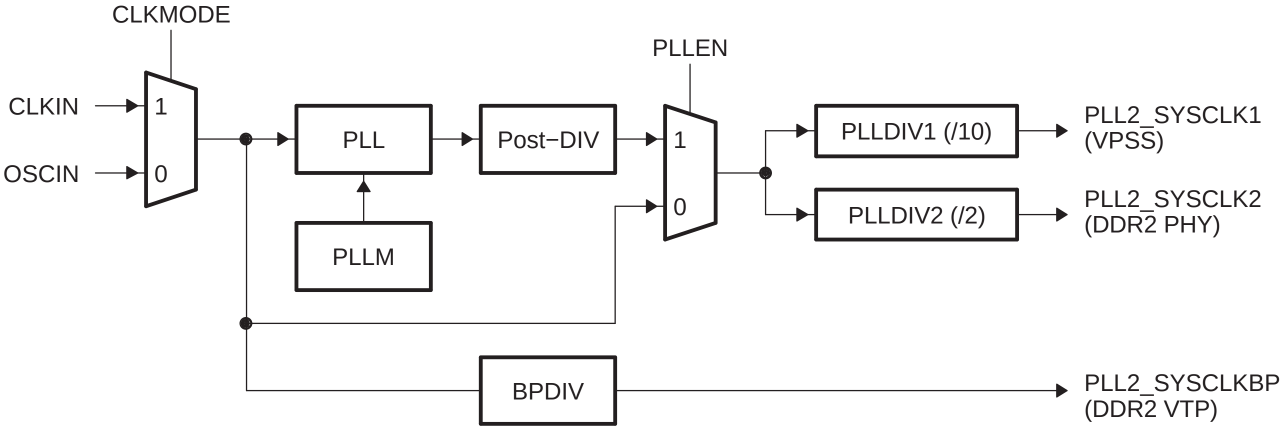

Figure 4. PLL2 Structure in the TMS320DM643x DMP

Table 11 summarizes the PLL/clock generator features on the DM6446 and the DM6437.

Table 11. PLL/Clock Generator Comparison

PLL	Feature	DM6446	DM6437
PLL1	PLL output frequencies	400-600 MHz	400-600 MHz [1]
	Post-divider ratios	1, 2, or 3	Not implemented
	Output divider ratios	Fixed 1,2,3,4,6	1 → 32; default 1,3,6 [2]
	Destinations	All but VPSS, DDR2	All but VPBE, DDR2
	Bypass divider ratios	1 → 32; default 1	1 → 32; default 1
	Bypass divider destination	CLKOUT0	VPBE
	Observed clock divider ratios	Not implemented	1 → 32; default 1
	Observed clock destination	Not implemented	CLKOUT0
PLL2	PLL output frequencies	400-900 MHz	400-900 MHz [1]
	Post-divider ratios	Fixed divide by 1	Not implemented
	Output divider ratios	1 → 16; default 10,2	1 → 32; default 2,10
	Destinations	VPSS, DDR2	DDR2, VPSS
	Bypass divider ratios	1 → 32; default 1	1 → 32; default 2
	Bypass divider destination	DDR2 VTP	DDR2 VTP
PLL3	PLL output frequencies	60 MHz	Not implemented
	Destinations	USB	Not implemented

[1] Range is smaller at reduced V_{DD} and in lower speed versions. See Section 10 for information on the various device speed versions power supply requirements.

[2] Programmed values must maintain 1:3:6 frequency ratio.

The peripheral mix and allocation of peripheral clocking is similar on the DM6446 and the DM6437, however, there are several differences in clock allocation between the two devices. Table 11 presents a comparison of peripheral/module clock domain assignments for the DM6446 and the DM6437. Note that CLKDIV2 runs at a frequency of CLKDIV1/2, CLKDIV3 runs at a frequency of CLKDIV1/3 and CLKDIV6 runs at a frequency of CLKDIV1/6.

Table 12. Peripheral/Module Clock Domain Assignments on the DM6446 and the DM6437

Clock Domain	Peripheral/Module	Peripheral/Module
CLKIN	UART0	UART0
CLKIN	UART1	UART1
CLKIN	UART2	
CLKIN		HECC
CLKIN	I2C	I2C
CLKIN	Timer0	Timer0
CLKIN	Timer1	Timer1
CLKIN	Timer2	Timer2
CLKIN	PWM0	PWM0
CLKIN	PWM1	PWM1
CLKIN	PWM2	PWM2
CLKDIV2	ARM Subsystem	
CLKDIV3	DDR2	DDR2
CLKDIV3	VPSS	VPSS
CLKDIV3	EDMA	EDMA
CLKDIV3	SCR	SCR
CLKDIV3		PCI
CLKDIV6	GPSC	GPSC
CLKDIV6	LPSCs	LPSCs
CLKDIV6	Ice Pick	Ice Pick
CLKDIV6		PLLC1
CLKDIV6		PLLC2
CLKDIV6	EMIFA	EMIFA
CLKDIV6	USB	
CLKDIV6	HPI	HPI
CLKDIV6	VLYNQ	VLYNQ
CLKDIV6	EMAC	EMAC
CLKDIV6	ATA/CF	
CLKDIV6	MMC/SD/SDIO	
CLKDIV6	SPI	
CLKDIV6	ASP	
CLKDIV6		McASP0
CLKDIV6		McBSP0
CLKDIV6		McBSP1
CLKDIV6	GPIO	GPIO
CLKDIV1	C64x+ CPU	C64x+ CPU

For information regarding initialization of the PLL/clock generators, see Section 8 of this document. For additional detailed information regarding use of the PLL/clock generator on the DM6437, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345).

4.9 UARTs

The DM6446 and the DM6437 feature the UART peripheral, and use the same peripheral module on both devices; therefore, migration of an application utilizing the UART from the DM6446 to the DM6437 requires little, if any, modification of hardware or software. The DM6446 contains three UARTs, while the DM6437 contains two.

The UART peripherals can be used for serial asynchronous communication between the DM6437 and other devices in the system. On the DM6446, only the third UART, UART2, features flow control capability using the RTS and CTS signals, whereas UART0 and UART1, do not implement this capability. On the DM6437, only the first UART, UART0, features flow control capability, while UART1 does not. Table 13 summarizes flow control support for the UART peripheral modules on the DM6446 and the DM6437.

Table 13. Flow Control Support on the DM6446 and DM6437 UARTs

	Flow Control Supported?	
Timer Feature	**DM6446**	**DM6437**
UART0	No	Yes
UART1	No	No
UART2	Yes	

Besides support for flow control, the only differences in usage between the UART peripheral modules on these two devices are signal locations within the device pinouts, and the pin multiplexing configuration settings.

For detailed information regarding use of the UART peripherals on the DM6437, see the *TMS320DM643x DMP Universal Asynchronous Receiver/Transmitter (UART) User's Guide* (SPRU997).

4.10 Timers

The DM6446 and DM6437 feature two 64-bit timers, each of which can also be operated as two 32-bit timers, and one 64-bit watchdog timer. The same timer peripheral modules are used on both devices; however, there are some minor differences in using these peripherals on the two devices. The following paragraphs describe in detail migration of applications using these two peripherals from the DM6446 to the DM6437 device.

4.10.1 General-Purpose (GP) Timers

The DM6446 and the DM6437 utilize the same timer peripheral modules; therefore, migration of an application utilizing the general-purpose timers from the DM6446 to the DM6437 requires little, if any, modification of hardware or software.

The DM6446 and DM6437 general-purpose timers actually support three modes of operation: as a 64-bit general-purpose timer, as dual unchained 32-bit GP timers, or as dual chained 32-bit timers. These timers can be used to generate periodic interrupts, or EDMA synchronization events. Also, on the DM6437, the GP timers can be used to generate an external clock output.

The predominant differences in usage between these peripheral modules on the DM6446 and the DM6437 are signal locations within the device pinouts, and the pin multiplexing configuration settings.

The only other differences in usage of the general-purpose timers between the two devices are in the number of off-chip I/O signals that are provided on each device. On the DM6446, there is only one external clock input available for the two timers, whereas on the DM6437, there are two. Also, on the DM6446, there are no timer outputs, whereas on the DM6437, there are two. Accordingly, the DM6437 can easily support any timer functionality required in migrating a DM6446 application to the DM6437.

Table 14 presents a comparison between the features offered by the general purpose timers on the DM6446 and the DM6437.

Table 14. DM6446 and DM6437 General-Purpose Timer Comparison

Timer Feature		DM6446	DM6437
Number of 64-bit timers		2	2
Support of dual 32-bit modes		Yes	Yes
Number of possible timer events	64-bit mode	2	2
	32-bit mode	4	4
Number of external clock inputs		1	2
Number of separate timer outputs		None	2

For detailed information regarding use of the DM6437 general purpose timers, see the *TMS320DM643x DMP 64-Bit Timer User's Guide* (SPRU989).

4.10.2 Watchdog Timer

The DM6446 and the DM6437 utilize the same watchdog timer peripheral modules; therefore, migration of an application utilizing the watchdog timers from the DM6446 to the DM6437 requires little, if any, modification of system software.

The watchdog timers can be extremely useful, especially in real-time systems, to allow the capability to recover in case of unexpected events which might otherwise cause the system to stop functioning properly. The DM6446 and DM6437 watchdog timers are 64 bits long, and both allow the capability to interrupt or reset the device if the watchdog timer is not serviced at a programmable interval set up by you. Note that the watchdog timers are clocked exclusively from internal clock sources, and do not generate any outputs from the device.

The only significant difference between the watchdog timers on these two devices is that on the DM6446, the watchdog timer can only be used to generate a device-level reset when the timeout interval expires. On the DM6437, the watchdog timer can also be used to generate a CPU interrupt as well as a device-level reset.

For detailed information regarding use of the DM6437 watchdog timer, see the *TMS320DM643x DMP 64-Bit Timer User's Guide* (SPRU989).

4.11 Ethernet MAC

The DM6446 and the DM6437 both feature the Ethernet MAC (EMAC) peripheral, which provides Ethernet interface capability and, both of these devices provide support for the IEEE 802.3 compliant 10/100 Mb/s Ethernet interface. The EMAC peripheral modules are functionally the same; therefore, migration of an application utilizing the EMAC interface from the DM6446 to the DM6437 requires little, if any, modification of hardware or software.

The Ethernet interface is comprised of the Ethernet media access controller (EMAC) and the physical layer (PHY) device management data input/output (MDIO) module. The EMAC controls the flow of packet data from the DSP to the PHY while the MDIO module controls PHY configuration and status monitoring.

Both the EMAC and the MDIO modules interface to the DSP through a custom interface that allows efficient data transmission and reception. This custom interface is referred to as the EMAC control module, and is considered integral to the EMAC/MDIO peripheral. The control module is also used to control device reset, interrupts, and system priority.

Although the DM6446 and DM6437 EMAC peripheral modules are functionally the same, there are some minor differences in their usage when migrating applications between these two devices. The most notable differences in usage between the EMAC peripheral modules on these two devices are signal locations within the device pinouts, and the pin multiplexing configuration settings.

Also, on the DM6446, the EMAC can only be controlled by the ARM CPU, and is not accessible to the DSP CPU. Therefore, code being migrated to the TMS320C64x+ DSP CPU core on the DM6437 is code that was originally running on the ARM CPU on the DM6446. Some code modifications may be required because of the differences in architecture between the ARM and the DSP.

For detailed information regarding use of the DM6437 Ethernet interface, see the *TMS320DM643x DMP Ethernet Media Access Controller (EMAC)/Management Data Input/Output (MDIO) Module User's Guide* (SPRU941).

4.12 USB Interface

The DM6446 features a USB interface, however, the DM6437 does not. Therefore, if USB capability is required in a DM6437 system, other provisions must be made for this.

One straightforward approach to providing USB capability in a DM6437 system is to make use of the Texas Instruments TUSB6020 VLYNQ to USB interface chip. This device can be connected to the VLYNQ interface on the DM6437, and then provides connectivity to a standard USB interface.

4.13 ATA/ATAPI Interface

The DM6446 features an ATA/ATAPI interface, however, the DM6437 does not. Therefore, if an ATA/ATAPI interface is required in a DM6437 system, other provisions must be made for this.

In order to provide ATA/ATAPI capability in a DM6437 system, a Texas Instruments TUSB6020 VLYNQ to USB interface chip can be used to provide connectivity to a USB compatible ATA/ATAPI interface, many of which are available in the industry.

4.14 MMC/SD Interface

The DM6446 features an MMC/SD interface, however, the DM6437 does not. Therefore, if an MMC/SD interface is required in a DM6437 system, other provisions must be made for this.

One possible approach to providing MMC/SD capability in a DM6437 system is to connect an interface device or FPGA to one of the other available interfaces on the DM6437, for example the EMIF (see Section 4.3).

4.15 I2C Interface

The DM6446 and the DM6437 feature an interface to I2C-compatible external devices. The I2C peripheral modules on these two devices are functionally the same; therefore, migration of an application utilizing the I2C interface from the DM6446 to the DM6437 requires little, if any, modification of hardware or software.

Although the DM6446 and DM6437 I2C peripheral modules are functionally the same, there are some minor differences in their usage when migrating applications between these two devices. The most notable difference in usage between the I2C peripheral modules on these two devices is the signal locations within the device pinouts.

Another difference between the I2C peripherals on these two devices is that the DM6437 has the added capability to ignore NACK responses on the I2C interface. This is controlled using the IGNACK (ignore NACK) bit in the ICEMDR register.

Finally, since the DM6437 contains an updated version of the I2C peripheral module, its peripheral ID register 1 (ICPID1) contains the value 16h, as opposed to the value on the DM6446 of 15h.

Use of the DM6437 I2C peripheral is described in detail in the *TMS320DM643x DMP Inter-Integrated Circuit (I2C) Peripheral User's Guide* (SPRU991).

4.16 High-End CAN Controller (HECC)

While the DM6446 does not feature the HECC peripheral, the DM6437 does; therefore this added capability can be utilized in DM6437 systems. The HECC peripheral uses a serial multimaster communication protocol that efficiently supports distributed real-time control, with a very high-level of security, and a communication rate of up to 1 Mbps. The HECC bus is ideal for applications operating in noisy and harsh environments, such as in the automotive and other industrial fields that require reliable communication or multiplexed wiring. For detailed information regarding use of the HECC peripheral on the DM6437, see the *TMS320DM643x DMP High-End CAN Controller (HECC) User's Guide* (SPRU981).

4.17 VLYNQ Interface

The DM6446 and DM6437 feature a VLYNQ interface, and the same peripheral module is used on both devices; therefore, migration of an application utilizing the VLYNQ interface from the DM6446 to the DM6437 requires little, if any, modification of hardware or software. The VLYNQ communications interface port is a low pin count, high-speed, point-to-point serial interface used for connecting to host processors and other VLYNQ compatible devices. This interface can be implemented in either a host-to-peripheral or peer-to-peer fashion.

The VLYNQ port utilizes a full-duplex serial bus where transmit and receive operations occur separately and simultaneously without interference. VLYNQ enables the extension of an internal bus segment to one or more external physical devices. The external devices are mapped to local physical address space, and appear as if they are on the internal bus of the DSP. The external devices must also have a VLYNQ interface.

VLYNQ uses a simple block code (8b/10b) packet format and supports in-band flow control so that no extra terminals are needed to indicate that overflow conditions might occur. The external device can also initiate read and write transactions.

Since the DM6446 and the DM6437 utilize the same VLYNQ interface peripheral module, the major significant differences in usage between the VLYNQ peripheral modules on these two devices are signal locations within the device pinouts, and the pin multiplexing configuration settings.

Other than the signal locations and pin multiplexing, the only other differences between the two VLYNQ modules are the contents of the Chip Version (CHIPVER) registers. On the DM6446, the CHIPVER register contains 25h, and on the DM6437, CHIPVER contains 2Dh.

For detailed information regarding use of the VLYNQ peripheral on the DM6437, see the *TMS320DM643x DMP VLYNQ Port User's Guide* (SPRU938).

4.18 HPI Interface

The DM6446 and the DM6437 feature the host port interface (HPI) peripheral through which an external host can communicate with the DSP device. Using the HPI, the host can access most internal memory and memory mapped resources, with only a few exceptions. The host functions as the master of this interface, which greatly increases flexibility of communications with the DSP device.

The HPI peripheral modules used on these two devices are the same; therefore, migration of an application utilizing the HPI from the DM6446 to the DM6437 requires little, if any, modification of hardware or software.

There are, however, a few differences in HPI usage between these two devices. The most basic of these differences are in the signal locations within the device pinouts, and the pin multiplexing setup on the two devices.

Another difference between the HPI on the DM6446 and the DM6437 is that the ADRMODE and CTLMODE bits in the HPI_CTL register are not supported on the DM6437, and the host always has exclusive control of the HPIC and HPIA registers.

One additional difference between the two devices is that on the DM6446, HPI access to memory mapped registers is limited to the Power and Sleep Controller (PSC) registers, the PLL1 and PLL2 registers, and the HPI configuration registers; whereas on the DM6437, the HPI can access the memory mapped registers for most of the peripheral modules on the device. The main resource that the HPI on the DM6437 cannot access is the PCI master back-end interface.

For detailed information regarding the use of the HPI interface on the DM6437, see the *TMS320DM643x DMP Host Port Interface (HPI) User's Guide* (SPRU998).

4.19 Pulse-Width Modulator (PWM) Outputs

The DM6446 and DM6437 feature dedicated PWM outputs, and the same peripheral module is used on both devices; therefore, migration of an application utilizing the PWM outputs from the DM6446 to the DM6437 requires little, if any, modification of hardware or software. Three programmable PWM outputs are provided.

On these devices, the PWM outputs provide the capability to generate a pulse periodic waveform for motor control or can act as a digital-to-analog converter with some external components. Each PWM output is implemented as a timer with a period counter and a first-phase duration comparator, where the bit width of the period and first-phase duration are both programmable. The period and the first-phase duration are controlled with 32-bit counters, and each PWM output can also be used to generate an interrupt and/or EDMA sync event.

The only differences in usage between the PWM peripheral modules on these two devices are the signal locations within the device pinouts, and the pin multiplexing configuration settings.

For detailed information regarding use of the PWM output capabilities on the DM6437, see the *TMS320DM643x DMP Pulse-Width Modulator (PWM) User's Guide* (SPRU995).

4.20 General-Purpose I/O (GPIO)

The DM6446 and the DM6437 feature a selection of pins that can be configured to provide independent single-bit general-purpose digital I/O. These GPIO bits can be used to interface to external signals, and to generate interrupts and EDMA synchronization events.

Since the GPIO configuration differs between these two devices, software modifications may be necessary in order to migrate from the DM6446 to the DM6437. Specifically, the DM6446 features 71 GPIO bits, while the DM6437 features 111 GPIO bits.

Note that although these two devices offer a different number of GPIO bits, the GPIO bit function is configured by similar sets of control registers located at the same base address on both devices. Therefore, software modifications required when migrating an application from the DM6446 to the DM6437 will frequently be minimal.

Additionally, on both devices, many GPIO pins are multiplexed with other pin functions; therefore, GPIO pin availability depends on what other functions are used on the device. For additional information regarding pin multiplexing on the DM6437, see Section 9 of this document.

For detailed information regarding the use of GPIO on the DM6437, see the *TMS320DM643x DMP General-Purpose Input/Output (GPIO) User's Guide* (SPRU988).

5 Interrupt Considerations

The DM6446 and the DM6437 both support servicing of a wide range of interrupts from a variety of sources, both on- and off-chip. Each device uses its own multiplexing scheme to select the specific sources that are allowed to interrupt their processors.

The DM6446 device supports a variety of interrupts to service the needs of its many peripherals and subsystems. Both the ARM and the C64x+™ are capable of servicing these interrupts; however, all of the device interrupts are routed to the ARM interrupt controller with only a limited set routed to the C64x+ interrupt controller. The interrupts can be selectively enabled or disabled in either of the controllers. In typical DM6446 applications, the ARM handles most of the peripheral interrupts and grants control to the C64x+, of interrupts that are relevant to DSP algorithms. Also, the ARM and DSP can communicate with each other through interrupts.

On the DM6446, the ARM CPU core accepts two basic interrupts, a normal interrupt request (IRQ) and a fast interrupt request (FIQ). There are 64 possible total interrupt sources supported by the ARM interrupt architecture, and 57 of these are used on the DM6446 device. These 57 interrupt sources are multiplexed to either the IRQ or FIQ interrupt, at eight different priority levels. This multiplexing and the interrupt priority level assignments are controlled by bits in the various interrupt controller registers.

When one of these 57 interrupts occurs, it is prioritized against any other pending and enabled interrupts, and sent to the ARM as either an IRQ or an FIQ. The ARM CPU then reads the address of the appropriate interrupt service routine to branch to from the interrupt entry table, loaded by the processor initialization routine. In this way, any of the 57 possible interrupts may be serviced.

The DM6437 interrupt structure is significantly different from that of the ARM CPU core; however, it is functionally the same as that of the DSP CPU core interrupt structure on the DM6446. Therefore, it is useful to compare the DM6446 DSP core interrupt structure to that of the DM6437 when migrating an application from the DM6446 to the DM6437, since on the DM6437 device, all of the interrupts required by the system must be handled by the DSP CPU core.

On the DM6446 and the DM6437, the DSP CPU core accepts a total of 16 possible independent interrupt inputs to the CPU, although some of these are not used. On the DSP CPU core, sources that are allowed to interrupt the CPU may be chosen from a possible selection of 128 system events, in addition to three interrupts which are always selected, allocated to reset, NMI, and a hardware exception interrupt. Note, however, that on the DM6437, although support for NMI is included in the interrupt controller structure, the actual NMI interrupt is not available; therefore, the NMI function is not used on this device.

The 128 possible system events are mapped to 12 independent interrupts using the interrupt selector, interrupt combiner, and exception combiner modules. The resultant 15 interrupt signals are sent to the CPU. Some of the possible interrupts on the DSP CPU can be generated from GPIO signals, and the polarities of these signals used to generate the interrupts are programmable through the GPIO control registers.

Table 15 presents a comparison of the interrupt capabilities of the DM6446 and the DM6437 devices.

Table 15. Interrupt Capability Comparison

Timer Feature	DM642		DM6437
CPU	**ARM CPU CORE**	**DSP CPU CORE**	**DSP CPU CORE**
Total number of CPU interrupts	2	16	
Fixed	2	3: Reset, NMI [1], H/W exception	3: Reset, NMI [1], H/W exception
Programmable	All sources	12	12
Total number of sources	64	128	128
Source selected by	Interrupt controller registers	Interrupt/ exception selector/ combiners	Interrupt/ exception selector/ combiners
Selectable polarity	GPIO interrupts only	GPIO interrupts only	GPIO interrupts only

[1] Note that NMI is not used on the DM6437.

The 128 system event sources from which the 12 interrupts routed to the CPU are chosen are hard-wired on the DM6446 and the DM6437. The interrupts that are actually routed to the CPU in each device are chosen from this selection as described above. This interrupt selection structure allows for maximum flexibility within the system for allocation of necessary interrupt servicing.

Table 16 presents a comparison of the 128 possible system events for generating interrupts available on the DM6446 and the DM6437.

Table 16. Interrupt System Event Comparison

	DM6446		DM6437	
Number	**Acronym**	**Source**	**Acronym**	**Source**
0	EVT0	C64x+ Int Ctl 0	EVT0	C64x+ Int Ctl 0
1	EVT1	C64x+ Int Ctl 1	EVT1	C64x+ Int Ctl 1
2	EVT2	C64x+ Int Ctl 2	EVT2	C64x+ Int Ctl 2
3	EVT3	C64x+ Int Ctl 3	EVT3	C64x+ Int Ctl 3
4	TINT0	Timer 0 – TINT12	TINTL0	Timer 0 – TINT12
5	TINT1	Timer 0 – TINT34	TINTH0	Timer 0 – TINT34
6	TINT2	Timer 1 – TINT12	TINTL1	Timer 1 – TINT12

Table 16. Interrupt System Event Comparison (continued)

	DM6446		DM6437	
Number	Acronym	Source	Acronym	Source
7	TINT3	Timer 1 – TINT34	TINTH1	Timer 1 – TINT34
8		Reserved	WDINT	Timer 2 – TINT12
9	EMU_DTDMA	C64x+ EMC	EMU_DTDMA	C64x+ EMC
10		Reserved		Reserved
11	EMU_RTDXRX	C64x+ RTDX	EMU_RTDXRX	C64x+ RTDX
12	EMU_RTDXTX	C64x+ RTDX	EMU_RTDXTX	C64x+ RTDX
13	IDMAINT0	C64x+ EMC 0	IDMAINT0	C64x+ EMC 0
14	IDMAINT1	C64x+ EMC 1	IDMAINT1	C64x+ EMC 1
15		Reserved		Reserved
16	ARM2DSP0	ARM to DSP Controller 0		Reserved
17	ARM2DSP1	ARM to DSP Controller 1		Reserved
18	ARM2DSP2	ARM to DSP Controller 2		Reserved
19	ARM2DSP3	ARM to DSP Controller 3		Reserved
20		Reserved		Reserved
21		Reserved		Reserved
22		Reserved		Reserved
23		Reserved		Reserved
24		Reserved	VDINT0	VPSS – CCDC 0
25		Reserved	VDINT1	VPSS – CCDC 1
26		Reserved	VDINT2	VPSS – CCDC 2
27		Reserved	HISTINT	VPSS – Histogram
28		Reserved	H3AINT	VPSS – AE/AWB/AF
29		Reserved	PRVUINT	VPSS – Previewer
30		Reserved	RSZINT	VPSS – Resizer
31		Reserved		Reserved
32		Reserved	VENCINT	VPSS – VPBE (VENC)
33		Reserved		Reserved
34		Reserved	EDMA3CC_GINT	EDMACC Global Interrupt
35		Reserved	EDMA3CC_INT0	EDMACC Interrupt Region 0
36	EDMA3CC_INT1	EDMACC Interrupt Region 1	EDMA3CC_INT1	EDMACC Interrupt Region 1
37	EDMA3CC_ERRINT	EDMA CC Error	EDMA3CC_ERRINT	EDMA CC Error
38	EDMA3TC_ERRINT0	EDMA TC0 Error	EDMA3TC_ERRINT0	EDMA TC0 Error
39	EDMA3TC_ERRINT1	EDMA TC1 Error	EDMA3TC_ERRINT1	EDMA TC1 Error
40	PSCINT	PSC ALLINT	EDMA3TC_ERRINT2	EDMA TC2 Error
41		Reserved	PSCINT	PSC ALLINT
42		Reserved		Reserved
43		Reserved	EMACINT	EMAC Memory Controller
44		Reserved		Reserved
45		Reserved		Reserved
46		Reserved		Reserved
47		Reserved	HPIINT	HPI
48	ASPXINT	ASP Transmit	MBXINT0	McBSP0 Transmit
49	ASPRINT	ASP Receive	MBRINT0	McBSP0 Receive
50		Reserved	MBXINT1	McBSP1 Transmit
51		Reserved	MBRINT1	McBSP1 Receive
52		Reserved		Reserved

Table 16. Interrupt System Event Comparison (continued)

	DM6446		DM6437	
Number	Acronym	Source	Acronym	Source
53		Reserved	DDRINT	DDR2 Memory Controller
54		Reserved	EMIFAINT	EMIFA
55		Reserved	VLQINT	VLYNQ
56		Reserved	PCIINT	PCI
57		Reserved	HECC0INT	HECC Interrupt 0
58		Reserved	HECC1INT	HECC Interrupt 1
59		Reserved	AXINT0	McASP0 Transmit
60		Reserved	ARINT0	McASP0 Receive
61		Reserved		Reserved
62		Reserved		Reserved
63		Reserved		Reserved
64		Reserved	GPIO0	GPIO
65		Reserved	GPIO1	GPIO
66		Reserved	GPIO2	GPIO
67		Reserved	GPIO3	GPIO
68		Reserved	GPIO4	GPIO
69		Reserved	GPIO5	GPIO
70		Reserved	GPIO6	GPIO
71		Reserved	GPIO7	GPIO
72		Reserved	GPIOBNK0	GPIO
73		Reserved	GPIOBNK1	GPIO
74		Reserved	GPIOBNK2	GPIO
75		Reserved	GPIOBNK3	GPIO
76		Reserved	GPIOBNK4	GPIO
77		Reserved	GPIOBNK5	GPIO
78		Reserved	GPIOBNK6	GPIO
79		Reserved		Reserved
80		Reserved	PWM0	PWM0
81		Reserved	PWM1	PWM1
82		Reserved	PWM2	PWM2
83		Reserved	IICINT0	I2C
84		Reserved	UARTINT0	UART0
85		Reserved	UARTINT1	UART1
86		Reserved		Reserved
87		Reserved		Reserved
88		Reserved		Reserved
89		Reserved		Reserved
90		Reserved		Reserved
91		Reserved		Reserved
92		Reserved		Reserved
93		Reserved		Reserved
94		Reserved		Reserved
95		Reserved		Reserved
96	INTERR	C64x+ Interrupt Controller Dropped CPU Interrupt Event	INTERR	C64x+ Interrupt Controller Dropped CPU Interrupt Event
97	EMC_IDMAERR	C64x+ EMC Invalid IDMA Parameters	EMC_IDMAERR	C64x+ EMC Invalid IDMA Parameters

Table 16. Interrupt System Event Comparison (continued)

	DM6446		DM6437	
Number	Acronym	Source	Acronym	Source
98		Reserved		Reserved
99		Reserved		Reserved
100		Reserved		Reserved
101		Reserved		Reserved
102		Reserved		Reserved
103		Reserved		Reserved
104		Reserved		Reserved
105		Reserved		Reserved
106		Reserved		Reserved
107		Reserved		Reserved
108		Reserved		Reserved
109		Reserved		Reserved
110		Reserved		Reserved
111		Reserved		Reserved
112	PMC_ED	C64x+ PMC		Reserved
113		Reserved	PMC_ED	C64x+ PMC
114		Reserved		Reserved
115		Reserved		Reserved
116	UMCED1	C64x+ UMC 1	UMCED1	C64x+ UMC 1
117	UMCED2	C64x+ UMC 2	UMCED2	C64x+ UMC 2
118	PDCERR	C64x+ PDC	PDCINT	C64x+ PDC
119	PVCINT	C64x+ PDC	SYSCMPA	C64x+ SYS
120	PMCCMPA	C64x+ PMC	PMCCMPA	C64x+ PMC
121	PMCDMPA	C64x+ PMC	PMCDMPA	C64x+ PMC
122	DMCCMPA	C64x+ DMC	DMCCMPA	C64x+ DMC
123	DMCDMPA	C64x+ DMC	DMCDMPA	C64x+ DMC
124	UMCCMPA	C64x+ UMC	UMCCMPA	C64x+ UMC
125	UMCDMPA	C64x+ UMC	UMCDMPA	C64x+ UMC
126	EMCCMPA	C64x+ EMC	EMCCMPA	C64x+ EMC
127	EMCDMPA	C64x+ EMC	EMCBUSERR	C64x+ EMC

For detailed information regarding the handling of interrupts on the DM6437, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345), the *TMS320DM643x DMP DSP Subsystem Reference Guide* (SPRU978), and the *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871).

6 Bootloading Capabilities

The DM6446 and the DM6437 provide the capability to transfer code from an external location into RAM to be executed following reset.

On both devices, the states of various input pins are sampled following reset, and the selected boot modes are determined based on these states.

On the DM6446, since this device contains two CPU cores, bootloading of each of these CPU cores must be considered separately, because both devices can execute their own independent code. Additionally, the ARM CPU core controls the reset of the DSP CPU core; therefore, to some extent, the ARM CPU serves as the master of bootloading the DSP CPU. Also, the DSP CPU does not contain its own dedicated bootloader program; therefore, all bootloading of code for the DSP is either performed by the ARM CPU or by an external device.

The ARM CPU contains the bootloader program which controls bootloading operations on the DM6446 device. Again, since the ARM CPU controls the reset of the DSP CPU, the ARM CPU bootloads first, and then handles bootloading of the DSP CPU.

The ARM CPU provides four bootload modes: host bootloading through the HPI, EMIF bootloading from NAND flash, serial bootloading through UART0, and an EMIF boot mode which does not transfer any code, but simply branches directly to external location 02000000h where it expects code to be located in NOR flash memory.

For host bootloading through the HPI, the ARM bootloader expects that the code to be executed is loaded through the HPI by an external host, and branches to this code once loading is completed.

If EMIF or serial bootloading is selected, the bootloader loads the code provided through either of these two interfaces by an external device, and then transfers control to this code.

The ARM boot modes are selected by the BTSEL[1:0] pins, and, when EMIF booting is used, the configuration of the data and address bus width of the EMIF are selected by the EM_WIDTH and AEAW[4:0] pins.

For the DSP CPU core, there are also four possible boot modes, which are controlled by the ARM CPU. These four boot modes comprise two types of host boot, and two types of boot in which no code is transferred, but a branch is simply performed to an external memory address.

In the two host boot modes, the code is loaded into the DSP memory either by the ARM or by an external host through the HPI, and control is then transferred to this code after the DSP is released from reset by the ARM.

In the other two boot modes for the DSP, no code is actually transferred, but the DSP is programmed to branch either to the fixed address of 42200000h or to an address provided by the ARM after the DSP is released from reset by the ARM.

The DSP boot modes are selected by the DSP_BT pin, and, when EMIF booting is used, the configuration of the data and address bus width of the EMIF are selected by the EM_WIDTH and AEAW[4:0] pins.

The DM6437 provides four basic bootloading capabilities - host boot through PCI or HPI, asynchronous EMIF boot, serial boot through I2C, SPI or UART, and EMU or no boot. In addition, the DM642 offers a *fast boot* selection which allows for various options for speeding up the DSP clock frequency during the bootload process, allowing for faster bootload times. The fast boot option can use either fixed or user-selected PLL multipliers for determining clock frequency. On the DM6437, the bootloader program located in the internal ROM on the device performs the bootloading function.

Table 17 shows a comparison of the bootloading capabilities of the DM6446 and the DM6437.

Table 17. Comparison of Bootloading Capabilities on the DM6446 and DM6437

Timer Feature		**DM6446**		**DM6437**
CPU		**ARM**	**DSP** [1]	
Number of Modes		4	4	Many: 4 basic + fast boot options
Modes	Host	Yes - HPI	Yes - HPI or ARM	Yes - HPI or PCI
	EMIF	Yes - NAND flash	No	Yes
	Serial	Yes - UART0	No	Yes - I2C, SPI, UART
	No boot	Yes - "EMIFA boot" [2]	Yes [3]	Yes - Through EMIF [2]
				Yes - “EMU Boot”
Selected by		BTSEL[1:0] pins	DSP_BT pin	BOOTMODE[3:0] pins
		EM_WIDTH pin	EM_WIDTH pin	PCIEN pin
		AEAW[4:0] pins	AEAW[4:0] pins	AEM[2:0] pins
				PLLMS[2:0] pins
				FASTBOOT pin

[1] DM6446 DSP does not have its own dedicated bootloader.
[2] Branches directly to EMIF address space.
[3] Can branch to address provided by ARM or directly to EMIF address 42200000h

For detailed information regarding use of the bootloader on the DM6437, see the *Using the TMS320DM643x Bootloader Application Report* (SPRAAG0).

7 Power Management

In most DSP systems, power management is an important concern to allow DSP functions to perform with the lowest power cost and minimal battery drain possible. The DM6446 and the DM6437 offer numerous options for power management.

On the DM6446, there are several basic categories of power management options, and many different variations of these options within these basic categories.

To manage power in the circuitry of the various peripherals and functional modules within the DM6446, the dedicated power and sleep controller (PSC) module implements the capability to turn on or off the clock to each module, therefore, controlling power usage on a module-by-module basis. The PSC is composed of the global PSC (GPSC) module, which contains memory mapped registers, PSC interrupt control, a state machine for each peripheral/module, and the local PSCs (LPSCs) for each peripheral/module controlled.

The DSP CPU on the DM6446 can also be completely powered down using the DSPPWRON bit in the CHP_SHRTSW register. Power management of the ARM CPU core is implemented with the ARM wait for interrupt instruction.

To manage power due to I/O buffers on the device, the DM6446 also provides the capability to power up or down groups of I/O pin buffers. This feature is controlled by writing to bits in the VDD3P3V_PWDN register. Note that this register controls only the power supply to the 3.3 V I/O buffers for each designated group of pins. The PSC controls enabling or disabling of the clock for each module.

In addition, the DM6446 also offers several options for managing power within the C64x+ DSP CPU core. These capabilities are implemented by various instructions and control registers within the DSP CPU, and control power within the CPU and internal memories.

The DM6446 also provides the capability to enable or disable the oscillator and PLL in software through the PLLCTL register, and to enable or disable the video DACs through the video port VPBE VENC VMOD and DACTST registers.

The DM6437 provides the same basic power management options featured on the DM6446; however, since there is no ARM CPU core on the DM6437, there is no ARM wait for the interrupt power management option. In addition, there is no option to power down the whole DSP subsystem as with the DSPPWRON bit in the CHP_SHRTSW register, since the DM6437 only contains the DSP CPU core. Otherwise, the power management options featured on the DM6437 are the same as those offered on the DM6446.

Table 18 summarizes the power management options available on the DM6446 and the DM6437.

Table 18. DM6446 and DM6437 Power Management Options

Option	Controlled By	Controls
A	GPSC/LPSC registers	Individual peripherals/modules
B	VDD3P3V_PWDN register	3.3 V I/O pin group buffers
C	Various instructions/control registers	DSP CPU core + memory
D	PLLCTL register	PLL and oscillator
E	VMOD/DACTST registers	Video DACs
F	CHP_SHRTSW register DSPPWRON bit	Full DSP powerdown [1]
G	ARM wait for interrupt instruction	ARM powerdown mode [1]

[1] Available on DM6446 only.

For the DM6446 and the DM6437, there are 40 possible LPSCs that are available to control clocks for various peripheral/modules, although some are reserved on each device. Table 19 presents a comparison of the 40 possible LPSC peripheral/module allocation assignments on the DM6446 and the DM6437.

Table 19. LPSC Peripheral/Module Allocation Assignment Comparison

LPSC Number	Peripheral/Module	
	DM6446	DM6437
0	VPSS DMA	VPSS DMA
1	VPSS MMR	VPSS MMR
2	EDMACC	EDMACC
3	EDMATC0	EDMATC0
4	EDMATC1	EDMATC1
5	EMAC	EDMATC2
6	EMAC Memory Controller	EMAC Memory Controller
7	MDIO	MDIO
8	Reserved	EMAC
9	USB	McASP0
10	ATA/CF	Reserved
11	VLYNQ	VLYNQ
12	HPI	HPI
13	DDR2 Memory Controller	DDR2 Memory Controller
14	EMIFA	EMIFA
15	MMC/SD/SDIO	PCI
16	Reserved	McBSP0
17	ASP	McBSP1
18	I2C	I2C
19	UART0	UART0
20	UART1	UART1
21	UART2	Reserved
22	SPI	HECC
23	PWM0	PWM0
24	PWM1	PWM1

Table 19. LPSC Peripheral/Module Allocation Assignment Comparison (continued)

LPSC Number	Peripheral/Module	
	DM6446	DM6437
25	PWM2	PWM2
26	GPIO	GPIO
27	TIMER0	TIMER0
28	TIMER1	TIMER1
29	Reserved	Reserved
30	Reserved	Reserved
31	Reserved	Reserved
32	Reserved	Reserved
33	Reserved	Reserved
34	Reserved	Reserved
35	Reserved	Reserved
36	Reserved	Reserved
37	Reserved	Reserved
38	Reserved	Reserved
39	C64x+ CPU	C64x+ CPU
40	VICP	Reserved

For detailed information regarding power management on the DM6437, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345), the *TMS320DM643x DMP DSP Subsystem Reference Guide* (SPRU978), and the *TMS320DM643x DMP High-End CAN Controller (HECC) User's Guide* (SPRU981).

8 PLL/Clock Modes at Reset

The DM6446 and DM6437 feature flexible clock generators which provide clocking to satisfy a wide variety of system requirements. To properly start up and initialize a DSP system, the clock generator must be able to provide appropriate clock signals to the device even before the device is released from reset. The DM6446 and DM6437 clock generators are designed to provide this capability.

Both of these devices can be clocked either by an external oscillator, or by using a crystal with the internal oscillator. The internal oscillator is enabled by default when reset is asserted, but can be disabled through software by writing to the PLLCTL register, if desired. The default clock mode for both devices at reset is a multiply by 1 of the input clock. Once the device is released from reset, the clock frequency can be changed through software by writing to registers in the PLL module. For additional information regarding considerations of PLL operation, see Section 4.8 of this document. The input clock frequency range on both devices is 20-30 MHz.

Table 20 presents a summary of the DM6446 and DM6437 PLL and clock mode initialization.

Table 20. DM6446 and DM6437 PLL/Clock Modes at Reset

Feature	DM6446	DM6437
Input clock frequency range	20-30 MHz	20-30 MHz
On-chip oscillator	Yes [1]	Yes [1]
Clock mode at reset	x1, oscillator enabled	x1, oscillator enabled
Clock mode at reset determined by	Not variable [2]	Not variable [2]
S/W can change clock rate after reset	Yes	Yes

[1] Oscillator is on by default. Turn it off in software after reset, if desired.

[2] Bootloader may change clock frequency after resets, see Section 6.

For detailed information regarding PLL and clock mode initialization, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345) and the *TMS320DM643x DMP DSP Subsystem Reference Guide* (SPRU978).

9 Pin Multiplexing

The DM6446 and the DM6437 use multiplexing of functions on various pins in order to maximize device features and flexibility, while minimizing pin count, and therefore package size and cost. You can accomplish pin multiplexing in a similar fashion on both devices; however, the actual implementation on each is different, therefore, software changes are necessary. Consult the DM6437 documentation for operational details.

Functionally, on both devices, the default pin multiplexing configuration is determined by the state of various input pins at reset, and the reset state of various register bits. Following reset, the pin multiplexing configuration can be modified through software using two dedicated memory mapped registers called PINMUX0 and PINMUX1. Writing to these two registers can be used to modify the pin multiplexing configuration to suit a wide variety of system applications. Note, however, that since pinout and pin multiplexing configurations are different on these two devices, bit assignments and programming of the PINMUX0 and PINMUX1 registers on these two devices is different.

For detailed information regarding pin multiplexing and its control on the DM6437, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345) and the device-specific user's guides for the specific peripherals being used.

In addition to the available device documentation, an interactive software utility is available to assist in determining the correct values to load into the DM6437 PINMUX0 and PINMUX1 registers. Using the graphical user interface of this software utility, select the desired peripheral mix from the possible configuration options on the device, and the software utility generates the PINMUX0 and PINMUX1 register settings. This software utility is available with the *TMS320DM643x Pin Multiplexing Utility Application Report* (SPRAAN3).

10 Power Supplies

The DM6446 and the DM6437 utilize multiple power supplies to maximize flexibility, performance, and minimize power dissipation, as well as to adhere to industry standards for various external interfaces. Additionally, some of the different clock speed versions of the devices have different power supply voltage level requirements.

The DM6446 utilizes a 1.2 V supply for the DSP and ARM core CPUs and internal logic, and 3.3 V and 1.8 V power supplies for its I/O pins. On the DM6446, the 1.8 V power supply is used by all of the I/O circuitry except the EMAC, MDIO, MMC, and SDIO interfaces and GPIOV33_[16:0]. The EMAC, MDIO, MMC, and SDIO interfaces and GPIOV33_[16:0] utilize the 3.3 V power supply.

The DM6437 also utilizes a 3.3 V supply for its I/O pins, and either a 1.2 V supply for 400, 500 or 600 MHz operation, or a 1.05 V supply for 400 MHz operation only. Additionally, the DM6437 also requires a 1.8 V supply for the DDR2 interface, and for the device PLLs.

Table 19 summarizes the power supply requirements for the DM6446 and the DM6437.

Table 21. Power Supply Requirements at Reset

Supply		DM6446	DM6437	
Core	Speed (MHz)	All	400	400/500/600
	Voltage	1.2 V	1.05 V	1.2 V
I/O 1	Voltage	3.3 V	3.3 V	3.3 V
I/O 2	Voltage	1.8 V	1.8 V	1.8 V

For additional detailed information regarding power supply requirements on the DM6437, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345).

11 Package and Pin Count Comparisons

The DM6446 and the DM6437 are provided in cost-efficient, high-density BGA packages. Since the two devices have different pinouts, the pin connections and locations are different between the two devices; therefore, PC board layout and signal connection modifications are necessary when migrating from a DM6446 to a DM6437 device. Note that for added flexibility, the DM6437 is available in two different package types, and is offered in both commercial and extended operating temperature range versions.

Table 22 shows a comparison of the packages and pin counts for the DM6446 and the DM6437.

Table 22. Package and Pin Count Comparison

Characteristic	DM6446	DM6437
Designator	361 pin ZWT (Pb-free)	361 pin ZWT (Pb-free)
Ball Pitch	0.8 mm	0.8 mm
Dimensions	16 × 16 mm	16 × 16 mm
Maximum Case Temperature	85°C	90°C (commercial) 125°C (extended)
Designator		376 pin ZDU (Pb-free)
Ball Pitch		1.00 mm
Dimensions		23 × 23 mm
Maximum Case Temperature		90°C (commercial) 125°C (extended)

For detailed information regarding pinout and mechanical dimensions of the DM6437 packages, see the *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345).

12 References

- *TMS320DM6437 Digital Media Processor Data Manual* (SPRS345)
- *TMS320C64x+ DSP Cache User's Guide* (SPRU862)
- *TMS320DM6437/35/33/31 Digital Media Processor (DMP) Silicon Revisions 1.2, 1.1, and 1.0 Silicon Errata* (SPRZ250).
- *TMS320DM643x DMP Video Processing Front End (VPFE) User's Guide* (SPRU977)
- *TMS320DM643x DMP Video Processing Back End (VPBE) User's Guide* (SPRU952)
- *TMS320DM643x DMP DDR2 Memory Controller User's Guide* (SPRU986)
- *TMS320DM643x DMP Asynchronous External Memory Interface (EMIF) User's Guide* (SPRU984)
- *TMS320DM643x DMP Enhanced Direct Memory Access (EDMA3) Controller User's Guide* (SPRU987)
- *TMS320DM643x DMP Peripheral Component Interconnect (PCI) User's Guide* (SPRU985)
- *TMS320DM643x DMP Multichannel Buffered Serial Port (McBSP) User's Guide* (SPRU943)
- *TMS320DM643x DMP Multichannel Audio Serial Port (McASP) User's Guide* (SPRU980)
- *TMS320DM643x DMP Universal Asynchronous Receiver/Transmitter (UART) User's Guide* (SPRU997)
- *TMS320DM643x DMP 64-Bit Timer User's Guide* (SPRU989)
- *TMS320DM643x DMP Ethernet Media Access Controller (EMAC)/Management Data Input/Output (MDIO) Module User's Guide* (SPRU941)
- *TMS320DM643x DMP High-End CAN Controller (HECC) User's Guide* (SPRU981)
- *TMS320DM643x DMP VLYNQ Port User's Guide* (SPRU938)
- *TMS320DM643x DMP Host Port Interface (HPI) User's Guide* (SPRU998)
- *TMS320DM643x DMP General-Purpose Input/Output (GPIO) User's Guide* (SPRU988)
- *TMS320DM643x DMP DSP Subsystem Reference Guide* (SPRU978)
- *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871)
- *Using the TMS320DM643x Bootloader Application Report* (SPRAAG0)

- *TMS320DM643x Pin Multiplexing Utility Application Report* (SPRAAN3)

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