TMS320C6713 to TMS320C672x Migration Guide

Brad Cobb, Anthony Seely, Nikolic Zoran *TI DSP Performance Audio Group*

This document describes the issues related to migrating from the TMS320C6713 to the TMS320C672x floating-point digital signal processors. Some issues are more fundamental in nature; these have been marked with a '**' symbol after the title.

Topic **Page**

1.1 Device Level Differences

This section describes some device-level differences between how the C6713 and C672x DSPs boot are configured, and the recommended operating conditions of the two devices.

1.1.1 Recommended Operating Conditions and Electrical Characteristics

The recommended operating conditions and electrical characteristics are largely identical between C6713 and C672x devices. The C672x supply voltage range is identical to that of the C6713 1.2V devices. The operating temperature ranges also identical. However, there are a few differences that should be noted.

1.1.1.1 Input Buffers Changed from LVTTL to LVCMOS **

The C672x DSP uses LVCMOS input buffers on all digital pins instead of the LVTTL pins used on the C6713 DSP. The VIH and VIL specifications are different for these buffers.

Table 1-1. Input Buffer Specifications

Parameter	LVTTL (C6713)	LVCMOS (C672x)	Comment
V_{IH}	2V (min)	0.7 D_{VDD} (min)	Higher on C672x. However, for D_{VDD}=3.43V V_{IH} is 2.4V, which is the min V_{OH} for LVTTL output buffers.
V_{IL}	0.8V (max)	0.3 D_{VDD} (max)	Higher on C672x, so normally not a concern.

Since the V_{IH} Parameter on C672x has no margin to the V_{OH}minimum of an LVTTL output buffer, any C6713 inputs driven by LVTTL outputs should be examined carefully when migrating to C672x devices.

1.1.1.2 Power Consumption

The C672x DSP is expected to consume approximately the same or less power than the C6713 DSP when operating at the same frequency.

1.1.2 Boot Differences

The C6713 DSP supports many different boot mode options in hardware, but the C672x DSP only supports booting from internal ROM in hardware. Additional boot modes (for example SPI, EMIF, UHPI) can be supported on C672x through the use of a ROM bootloader program. The C672x does provide hardware support to capture the state of sixteen pins at the rising edge of the $\overline{\text{RESET}}$ pin. This information is available to the ROM bootloader through the CFGPIN0 and CFGPIN1 registers. Note that typically, only a small subset of these pins are actually used by the bootloader. Refer to the C9230C100 TMS320C672x Floating-Point Digital Signal Processor ROM Data Manual (literature number SPRS277) for details on supported bootmodes and their implementation.

1.1.3 Package Options

The C672x is offered in smaller packages than the C6713 DSP with comparable features.

The C672x is offered in a 144 PowerPad TQFP (RFP) package with a 16-bit EMIF. The C6713 DSP with similar capability and 16 bit EMIF is offered in a 208 PowerPad TQFP (PYP) package.

The C672x is also offered in a 256 PBGA (GDH) package, which is a 16x16 BGA with a 1mm pitch.

*1.1.4 Soldering Power-Pad is Required for Electrical Reasons **

The C672x utilizes the Power-Pad for two purposes:

- Thermal connection for power dissipation
- **Electrical connection** for low impedance connection to V_{SS}

The second purpose is new to C672x, the C6713 only used the Power-Pad for thermal purposes. **The Power-pad must be soldered to a V_{SS} plane or grid in order to achieve the datasheet electrical performance specifications.**

*1.1.5 Peripheral Selection (DEVCFG Register and HPI_EN on C6713) Replaced with Granular Scheme ***

The C6713 DSP used global enable and disable bits in the DEVCFG register to select different peripheral functions. For example, the DEVCFG register bit 0 (MCBSP1DIS) was used to select between the MCBSP1 and I2C or McASP functions for the six McBSP1 pins.

On the C672x DSP pin multiplexing is inferred from the output enable signal of each peripheral sharing the pin. A maximum of one peripheral should configure a given IO pin as an output at a time. This peripheral will then control the value driven out the pin. The input buffer of a given pin will drive all peripherals sharing that pin in parallel.

On the C672x DSP most pin multiplexing decisions are made on a pin-by-pin rather than on a module-by-module basis. (Note that the UHPI module does group certain pins such as UHPI_HD[31:0] byte lanes when determining whether these pins are GPIO or functional).

One **important exception** exists on the C672x DSP. The multiplexing of pins EM_D[31:16]/UHPI_HA[15:0] is controlled by a single bit in the device level configuration register UHPI.

*1.1.6 Most I/O pins do not have internal pullups/pulldowns ***

On the C6713 DSP almost all I/O pins also had an internal pullup or pulldown resistor.

On the C672x DSP only a subset of pins have internal pullup or pulldown resistors. Consult the device datasheet (SPRS268) terminal functions table which lists which pins have internal resistors.

1.2 CPU Core Enhancements

The C672x DSP uses the enhanced C67x+ CPU. It is backward-compatible with the C67x CPU core used on the C6713 device. In addition, it has a significant number of enhancements.

*1.2.1 Compiler Switch Required to Enable C67x+ Optimizations ***

The C/C++ compiler requires a switch to be set to enable C67x+ specific optimizations. The -mv67p compiler switch should be used for the C672x as opposed to the -mv6700 compiler switch for the C6713. If using Code Composer Studio, this switch can be set in the project's build options using the Target Version drop-down menu under the Basic category.

1.2.2 Higher CPU Operating Frequency Available

The C672x DSP family supports operation at 300MHz in the 256 PBGA (GDH) package and 250MHz in the 144 PowerPad TQFP (RFP) package, across a C_{VDD} range of 1.14V to 1.32V.

In contrast, the C6713 DSP family is rated for 225MHz in the 272 PBGA and 208 PowerPad TQFP at the same C_{VDD} range. A 300MHz C6713 is available in the 272 PBGA package but at a higher C_{VDD} range of 1.33V to 1.47V.

1.2.3 Twice the General Purpose Registers

The C67x+ CPU on C672x has 64 general purpose registers, twice as many as the C6x CPU on the C6713 device.

The additional registers are a particular benefit to the C/C++ compilers ability to schedule loops, since register pressure is reduced. This translates to more instructions executed per cycle.

1.2.4 Execution Packets May Span Fetch Packets on C672x

The C67x+ CPU has the ability to schedule execution packets that span across two fetch packets. This results in a code density improvement since it removes the need to pad the end of some fetch packets with NOPs. This feature is not available on the C6713 CPU. *There is no cycle penalty for spanning.*

*Spanning is restricted to execution packets that are not branch targets, except where the branch is a return from an ISR.*Returning from an ISR to spanning code is supported, but *only with the B IRP, B NRP, and B ARP instructions*.

1.2.5 Restrictions on ISR Return Instructions for C672x **

On the C672x ISRs must be compiled with the C67x+ switch to ensure that only the B IRP, B NRP, and B ARP instructions are used to return from the ISR.

ISRs written in assembly language should only use the B IRP, B NRP and B ARP instructions to return.

These restrictions do not apply to the C6713 device. On the C672x these restrictions apply if **any code** including code in internal ROM has been compiled with the C67x+ Optimizations enabled.

To support legacy C67x code, the C672x is not subject to these restrictions if **all code** has been compiled for the C67x only.

1.2.6 Additional Cycle required for ISR Return on C672x

To support execution packet spanning fetch packets without restricting spanning instructions to be non-interruptible, the C67x+ CPU automatically adds an extra cycle when returning from an interrupt compared to the C67x CPU.

The extra cycle is added to all ISR returns whether or not they are to spanning execution packets.

1.2.7 S Unit Enhanced with Floating Point Adder

The two S Units on the C67x+ CPU have been enhanced with the addition of a floating point adder. This allows them to execute ADDSP, ADDDP, SUBSP and SUBDP.

The C67x+ also has the two floating point adders that are available in the C67x CPU L Units. This means that the C672x can execute up to four floating point add or subtract instructions per cycle.

1.2.8 M Unit Enhanced with Mixed Precision Floating Point Multiply Instructions

Audio applications in particular may benefit from the mixed precision floating point multiply instructions available on the C67x+ CPU. The new instructions are:

- MPYSPDP (SP x DP $\rightarrow$ DP)
- MPYSP2DP (SP x SP $\rightarrow$ DP)

These instructions provide faster alternatives to the full double precision MPYDP (DP x DP $\rightarrow$ DP).

1.2.9 Cross Path Performance Doubled

The number of times a cross-path operand can be sourced in a single cycle is increased from one to two.

Note that cross path register read(s) do not count against the limit of four reads of the same register in one cycle.

1.2.10 New Registers for Direct Communication with dMAX

Two new registers have been added which allow the C672x to communicate directly to the dMAX through it's control register file.

The DETR register allows the CPU to trigger dMAX events. The DESR register allows the CPU to check the status of dMAX completion codes and status. Both of these registers are accessed with low overhead through the use of the "MVC" Move Between Control File and the Register File instruction.

1.2.11 Changes to the CSR Register

The CPU CSR Register CPU_ID field and REV_ID field have been updated to 0x03 and 0x00 respectively on the C672x. The CPU_ID for C6713 was 0x02. This change reflects the migration from the C67x to the C67x+ CPU.

1.3 Memory System Changes

The C672x memory system is simplified compared to the memory system on the C6713 DSP. This section gives a brief overview of some of the changes.

1.3.1 L1 Program Cache **

The C672x DSP has a more powerful L1 program cache than the C6713 DSP. Some of the key differences are listed in Table 1-2.

Table 1-2. L1 Program Cache Key Differences

Feature	C6713 DSP	C672x DSP
Size	4K Bytes	32K Bytes
CPU Stall on L1P Miss	5 Cycles to Internal RAM or L2 Cache	3 Cycles to Internal ROM/RAM
Operating Modes	Direct Mapped Cache Only	Direct Mapped Cache, Freeze, Bypass

1.3.2 Addition of Internal ROM

Unline the C6713, the C672x has an internal ROM. This ROM contains the primary software bootloader as well as additional useful software. Details on the contents of the Internal ROM can be found in the *C9230C100 TMS320C672x Floating-Point Digital Signal Processor ROM Data Manual* (SPRS277).

1.3.2.1 Patching the Internal ROM

It is necessary to link the latest ROM patch to your code in order to insure correct and optimized DSP functionality. The steps necessary to obtain and install the lastest ROM patch can be found in the *C9230C100 TMS320C672x Floating-Point Digital Signal Processor ROM Data Manual* (SPRS277).

1.3.3 Data Memory: Data Cache Replaced with Banked RAM/ROM **

The C672x memory system does not use any data cache. Instead, it utilizes two pages of banked ROM and one page of banked RAM. These memories are accessible in a single cycle as long as bank and page conflicts are avoided.

1.3.4 No L2 Cache **

The internal RAM on the C672x Device cannot be configured as a L2 cache as was possible on C6713. Instead, applications that need to access a large amount of off chip data can utilize the dMAX to transfer data on and off chip.

The L1 Program cache on the C672x device does cache program code from external memory, so dynamically paging program sections into the internal memory is not usually required.

1.3.5 L1P Cache Requires Invalidate if Code Paging is Implemented **

The L1P program cache on the C672x device must be invalidated to maintain coherency whenever code is paged in and out of RAM. This must performed explicitly by the code that does the paging.

1.3.6 Crossbar replaces TC/TR Nodes and EDMA **

The C6713 TC/TR nodes have been replaced with a simpler crossbar switch fabric on the C672x. In general, the memory map, priorities, and latencies when accessing different regions is different. Refer to the device datasheet (SPRS268) for more information.

1.3.7 Bridge Reset Required on Startup and After PLL Changes **

On C672x, the CSP bridge must be reset through the CFGBRIDGE after reset and after any changes to the PLL Controller configuration; before any dMAX or UHPI operations attempt to access the internal ROM and RAM. This is a new requirement to C672x.

1.4 DMA Differences: EDMA replaced with dMAX **

The C6713 EDMA has been replaced on the C672x with the new dMAX DMA module. The dMAX retains many features of the EDMA while expanding its functionality significantly. The dMAX architecture and control registers are different than the EDMA.Table 1-3 provides an overview of some of the major feature differences between EDMA and dMAX, and Table 1-4 compares the EDMA registers with their functionally-similar dMAX control registers. See the *TMS320C672x DSP Dual Data Movement Accelerator (dMAX) Reference Guide* (SPRU795) for more detailed information.

Table 1-3. Partial List of Feature Differences Between the EDMA and dMAX

Features	EDMA Features	dMAX Features
Highest Transfer Dimension Supported	2D transfer support	3D transfer support
FIFO Read/Write Transfer Support	No FIFO Support	FIFO (Circular Buffer) Support with table-based multi-tap delay reads and writes
Alternate Transfer Completion Interrupt Support	No alternate transfer complete interrupt	Alternate transfer complete interrupt supported
Event Signal Polarity	Fixed polarity event signals	Event signal polarity selection
Transfer Index Field Sharing	Common index fields for source and destination	Independent index fields for source and destination for all transfer dimensions.
Transfer Index Units	Transfer indexes expressed in number of bytes	Transfer indexes expressed in number of elements
Transfer Index Updating	Frame index added to the start element address in a frame	Frame index added to the address of a last element in a frame
Transfer Linking Support	Transfer Linking Supported	No Transfer Linking, but each Transfer Entry supports ping-pong buffering using built-in reload registers.
Transfer Chaining Support	Transfer Chaining Supported	No Transfer Chaining
QDMA Transfer Support	QDMA Transfers Supported	No QDMA Transfers
Data Path Width	64-bit data path width	Two 32-bit wide data paths
CPU Interrupt Support	1 CPU Interrupt (for transfer completion notification)	8 CPU Interrupts (1 dedicated for transfer completion notification, 1 dedicated for FIFO status and error notifications, 6 dedicated for peripheral to CPU interrupts)
DMA Clock Rate	EDMA clock rate equals CPU clock rate	dMAX clock rate max. is ½ of CPU clock rate

Table 1-4. Control Register Comparison Between the EDMA and dMAX

Function	EDMA Registers	dMAX Registers
Event to Channel Mapping	Event Selector Registers (ESEL)	Dedicated Event Entry Registers for each event specify an associated Transfer Entry Register Set
Transfer Status	Priority Queue Status Register (PQSR)	dMAX Event Status Register (DESR) is internal to the CPU. It reports current dMAX activity and provides a low-overhead mirror of the DTCRs
Transfer Completion Interrupt Posting	Channel Interrupt Pending Register (CIPR)	dMAX Transfer Completion Register (DTCR)
Transfer Completion Interrupt Enabling	Channel Interrupt Enable Register (CIER)	None. Disabling of individual channel interrupts not supported.
Transfer Chaining	Channel Chain Enable Register (CCER)	None. Chaining not supported.
Event Capturing	Event Register (ER)	dMAX Event Registers (DER) provide the current state of event signals, and the dMAX Event Flag Register (DEFR) latches event signal transitions.
Event Enabling	Event Enable Register (EER)	dMAX Event Enable Register (DEER) and dMAX Event Disable Register (DEDR)
Event Clearing	Event Clear Register (ECR)	None. Clearing of events not supported.
Manual (CPU) Event Triggering	Event Set Register (ESR)	dMAX Event Trigger Register (DETR) can trigger a subset of the dMAX Events
Transfer Channel Parameters	Channel Parameter Entries	Event Entries and Transfer Entries combine to store the transfer parameters
Event Priority	PRI bit field in the OPT parameter of the Channel Parameter Entry	dMAX Event High Priority Register (DEHPR) and dMAX Event Low Priority Register (DELPR) control the priority group for an Event
Event Signal Polarity	None. Event signal polarity is fixed.	dMAX Event Polarity Register (DEPR) selects the polarity of the event signals
FIFO Status Reporting	None. FIFO Transfers not supported.	dMAX FIFO Status Registers (DFSR) post FIFO Transfer errors and notifications

1.5 External Interrupts **

The C672x DSP does not have any dedicated GPIO/Interrupt pins. This is different than the C6713 DSP which has up to four external interrupts.

However, the C672x DSP supports external interrupts with dedicated edge detect hardware using the McASP AMUTEIN signals.

1.5.1 External Interrupts with edge detect hardware

The AMUTEIN0, AMUTEIN1, and AMUTEIN2 signals can be used to provide external interrupt support. A very short pulse (two SYSCLK2 cycles or more) on these signals will be detected and captured in hardware inside the dMAX accelerator. The dMAX can then be configured to act on these interrupts and/or pass them on to the DSP CPU. There are seven pins from which three can be chosen to map to the AMUTEIN0, AMUTEIN1, and AMUTEIN2 interrupts.

The CFGMCASP0, CFGMCASP1, and CFGMCASP2 global control registers allow the selection of AMUTEINx sources from the following:

1. Disabled
2. AXR0[7]/SPI1_CLK
3. AXR0[8]/AXR1[5]/SPI1_SOMI
4. AXR0[9]/AXR1[4]/SPI1_SIMO
5. AHCLKR2
6. SPI0_SIMO

7. SPI0_SCS/I2C1_SCL
8. SPI0_ENA/I2C1_SDA

1.6 External Memory Interface Differences

The C672x DSP External Memory Interface (EMIF) module is a new module; it is different than the EMIF on the C6713 DSP. The following sections highlight the major differences between the C6713 EMIF and the C672x EMIF. See the C672x External Memory Interface (EMIF) Peripheral Reference Guide (spru711) for more detailed information.

1.6.1 Control Registers

All control registers are different on the C672x EMIF. Table 1-5 compares the C6713 EMIF control registers to the C672x EMIF control registers.

Table 1-5. Control Register Comparison Between the C6713 EMIF and C672x EMIF

Function	C6713 EMIF Registers	C672x EMIF Registers
Shared Memory settings, clock enabling/disabling, ARDY monitoring	Global Control Register (GBLCTL)	None. Features not supported.
CE Space Configuration and Asynchronous Timings	CE Space Control Registers (CECTL)	Asynchronous 1 Configuration Register (A1CR) contains the asynchronous memory timing parameters. No MTYPE bit-field exists because the C672x EMIF has dedicated CE spaces.
SDRAM Configuration and Timing	SDRAM Control Register (SDCTL)	SDRAM Configuration Register (SDCR) and SDRAM Timing Register (SDTR)
SDRAM Refresh Settings	SDRAM Timing Register (SDTIM)	SDRAM Refresh Control Register (SDRCR)
SDRAM Timing	SDRAM Extension Register (SDEXT)	SDRAM Timing Register (SDTR) and SDRAM Self Refresh Exit Timing Register (SDSRETR)
Wait/Ready Settings	None. ARDY polarity is fixed and there is no maximum wait time.	Asynchronous Wait Cycle Configuration Register (AWCCR) used to set the polarity of the wait/ready signal (EM_WAIT) and define the maximum extended wait period.
Interrupt Generation	None. EMIF cannot generate CPU interrupts.	EMIF Interrupt Raw/Masked/Mask Set/Mask Clear Registers (EIRR/EIMR/EIMSR/EIMCR) to control EMIF interrupt generation.

1.6.2 Pin Set Differences

The C672x EMIF pin set differs significantly from the C6713 EMIF pin set. Table 1-6 compares the pin sets of the two EMIFs.

Table 1-6. Pin Set Comparison Between the C6713 EMIF and the C672x EMIF

C6713 EMIF Pins	C672x EMIF Pins	Notes
ECLKIN	None.	External clock input not supported on C672x EMIF
ECLKOUT	EM_CLK	None.
ED[31:0]	EM_D[31:0]	None.
EA[21:2]	EM_A[12:0] and EM_BA[1:0]	C672x supports dedicated address pins (EM_A) and bank address pins (EM_BA) when interfacing to SDRAM. EM_BA[1:0] provide half-word and byte address lines when interfacing to 16- and 8-bit asynchronous memories.
CE[3:0]	EM_CS[0] and EM_CS[2]	C672x supports dedicated Chip Select pins. EM_CS[0] always provides the SDRAM chip select, while EM_CS[2] always provides the asynchronous memory chip select.
BE[3:0]	EM_WE_DQM[3:0]	Depending on the C672x EMIF's mode of operation, the EM_WE_DQM pins provide either byte-write strobes or byte enable signals when interfacing to asynchronous memories. The EM_WE_DQM pins provide the DQM signals when interfacing to SDRAM.

Table 1-6. Pin Set Comparison Between the C6713 EMIF and the C672x EMIF (continued)

C6713 EMIF Pins	C672x EMIF Pins	Notes
ARDY	EM_WAIT	EM_WAIT has programmable polarity on the C672x EMIF.
$\overline{\text{AOE}}$/$\overline{\text{SDRAS}}$/$\overline{\text{SSOE}}$	$\overline{\text{EM\_RAS}}$	The C672x EMIF does not have a signal which corresponds to the $\overline{\text{AOE}}$ signal on the C6713 EMIF. The SDRAM RAS signal is provided by the $\overline{\text{EM\_RAS}}$ pin on the C672x EMIF.
$\overline{\text{ARE}}$/$\overline{\text{SDCAS}}$/$\overline{\text{SSADS}}$	$\overline{\text{EM\_OE}}$ and $\overline{\text{EM\_CAS}}$	The C6713 EMIF $\overline{\text{ARE}}$ signal is renamed $\overline{\text{EM\_OE}}$ on the C672x EMIF and is no longer multiplexed with the SDRAM CAS signal. Instead the SDRAM CAS signal is provided by the $\overline{\text{EM\_CAS}}$ pin on the C672x EMIF.
$\overline{\text{AWE}}$/$\overline{\text{SDWE}}$/$\overline{\text{SSWE}}$	$\overline{\text{EM\_WE}}$	The asynchronous write-enable signal continues to be multiplexed with the SDRAM write-enable signal on the C672x EMIF, but now has a single name $\overline{\text{EM\_WE}}$.
None.	EM_$\overline{\text{RW}}$	The C672x EMIF provides a new signal, EM_$\overline{\text{RW}}$, that is high during asynchronous reads and low during asynchronous writes.
$\overline{\text{HOLD}}$	None.	Shared Memory Interfaces are not supported on the C672x EMIF.
$\overline{\text{HOLDA}}$	None.	Shared Memory Interfaces are not supported on the C672x EMIF.
$\overline{\text{BUSREQ}}$	None.	Shared Memory Interfaces are not supported on the C672x EMIF.

*1.6.3 No ECLKIN Pin ***

The C672x EMIF does not have an external clock input pin like the ECLKIN pin on the C6713 EMIF. Instead, the C672x EMIF clock is always sourced from the SYSCLK3 domain of the PLL Controller and can be configured by setting the PLL Controller multiplier and dividers.

1.6.4 SDRAM Data Widths: Only x16 and x32 Supported on C672x EMIF

The C672x EMIF can only be configured to operate with a x16 or x32 SDRAM data-bus width. 8-bit wide SDRAM which was supported on the C6713 EMIF is no longer supported.

1.6.5 C672x EMIF EM_WAIT replaces C6713 EMIF ARDY

The fixed polarity ARDY signal on the C6713 is replaced by the programmable polarity EM_WAIT signal on the C672x. In addition, the C672x places a user-programmable limit on length of time that an asynchronous access can be extended. When this limit is reached, the access completes even if the EM_WAIT signal is still asserted. The C672x EMIF can also be programmed to provide an interrupt to the CPU when this limit is reached.

1.6.6 C672x EMIF does not support shared external memory

The C672x EMIF does not have the Hold Interface that was available on the C6713 EMIF, therefore shared external memory is not supported with the C672x EMIF.

*1.6.7 SDRAM Address Bus Connection Differences ***

The C6713 EMIF provides all SDRAM address signals via the EA[21:2] address bus. The lower address signals provide the SDRAM row and column address, while the upper address signals provide the SDRAM bank address. The exact pins used for the SDRAM bank address varies depending on the row size of the attached SDRAM.

In contrast, the C672x EMIF's EM_A[12:0] address bus only provides the SDRAM row and column address, while the EM_BA[1:0] pins provide the SDRAM bank address. This simplifies interfacing to SDRAM devices.

1.6.8 Asynchronous Address Bus Connection Differences **

The C6713 EMIF asynchronous controller performs internal address shifting so that a word, half-word, or byte address is placed on the EA address bus when the data-bus is configured to 32-, 16-, or 8-bit respectively. The C672x EMIF instead uses a combination of the EM_A[12:0] and EM_BA[1:0] pins to provide the address for asynchronous interfaces. Regardless of the selected data-bus width, the EM_A[12:0] pins always provide a 32-bit word address. When interfacing to 16- and 8-bit devices, the EM_BA[1] and EM_BA[0] pins provide the half-word and byte address signals respectively.

1.6.9 C672x EMIF has One Dedicated SDRAM Chip Select and One Dedicated Asynchronous Chip Select **

The C6713 EMIF has 4 configurable Chip Enable spaces. The MTYPE field in the CE Space Control Registers selects which memory type is connected to each Chip Enable. Alternatively, the C672x EMIF has two dedicated Chip Selects. The EM_CS[0] chip select is dedicated to SDRAM and the EM_CS[2] chip select is dedicated to asynchronous memory.

1.6.10 SBSRAM Not Supported on C672x EMIF

The C672x EMIF does not support interfacing with SBSRAM. The only synchronous memory type supported by the C672x EMIF is SDR SDRAM.

1.6.11 Two Asynchronous Modes of Operation **

The C672x EMIF can be operated in one of two modes when interfacing with asynchronous memory: Select Strobe Mode or WE Strobe Mode. The chosen mode determines the function of the EM_$\overline{WE}$_DQM signals and the shaping of the EM_CS[0] signal. In Select Strobe Mode, the EM_CS[0] signal is only active during the STROBE period of and access and the EM_$\overline{WE}$_DQM signals behave as byte-enables. In WE Strobe Mode, the EM_CS[0] signal is active during the entire access and the EM_$\overline{WE}$_DQM signals behave as byte-write strobes. See the C672x EMIF Peripheral Reference Guide (spru711) for waveforms and further details.

Table 1-7. Chip Select and Byte Enable Timing Differences

		Asserted During		
	Mode	**Setup Period**	**Strobe Period**	**Hold Period**
C6713 $\overline{CS[3:0]}$		Yes	Yes	Yes
C672x $\overline{EM\_CS[2]}$	Select Strobe	-	Yes	-
	Write Enable Strobe	Yes	Yes	Yes
C6713 $\overline{BE[3:0]}$		Yes	Yes	Yes
C672x EM_$\overline{WE}$_DQM[3:0]	Select Strobe	Yes	Yes	Yes
	Write Enable Strobe	-	Yes	-

1.6.12 SDRAM Initialization Differences

The C6713 EMIF begins initialization of the SDRAM when the INIT bit in the SDRAM Control Register is set. In contrast, the C672x EMIF initializes the SDRAM immediately on coming out of reset or upon writing to the lower-three bytes of the SDRAM Control Register. Because the SDRAM is initialized immediately on coming out of reset, there is the possibility of violating the Power-up Constraint of the attached SDRAM device. If the Power-up Constraint is violated, a special sequence must be followed to complete the SDRAM initialization. See the C672x EMIF Peripheral Reference Guide (spru711) for further details in SDRAM initialization.

1.6.13 SDRAM Page/Bank Traversal Differences

The SDRAM address mapping scheme of the C672x EMIF differs from that of the C6713 EMIF. As the logical address is incremented, the C6713 EMIF traverses from page-to-page within a single bank, issuing the PRECHARGE command as it moves between pages. The C672x EMIF instead traverses from bank-to-bank as the logical address is incremented, resulting in more efficient linear accesses.

1.6.14 Asynchronous Parameter Differences

The C672x EMIF asynchronous HOLD period must be a minimum of one cycle, whereas the C6713 EMIF asynchronous HOLD period can be programmed to zero cycles. In addition, the exact number of cycles desired for each period of an access should be programmed on the C6713 EMIF, while a value one less than the desired number of cycles should be programmed on the C672x EMIF.

1.6.15 Data Bus Parking used on the C672x EMIF

The C6713 EMIF tri-states the data bus when idle, whereas the C672x EMIF drives the data bus when idle. This is called data bus parking. See the C672x EMIF Peripheral Reference Guide (spru711) for more details on data bus parking.

1.6.16 Self-Refresh Mode Support on the C672x EMIF

The C672x EMIF supports placing the attached SDRAM device into Self-Refresh Mode in which the SDRAM provides its own refreshes. This is useful for changing the EMIF clock without corrupting SDRAM data. Self-Refresh Mode can also be used to reduce power consumption by the SDRAM.

1.7 Host Port Interface Comparison

The C672x DSP has an enhanced Host Port Interface module (HPI). The UHPI has many additional features as compared to the C6713 HPI module. A comparison of features is given in Table 1-8.

Table 1-8. Comparison of C672x UHPI and C6713 HPI Features

Feature	C6713 HPI	C672x UHPI
Multiplexed Host Address/Data Half-word (16 Bit) Mode	Yes	Yes
Multiplexed Host Address/Data Full-word (32 Bit) Mode	No	Yes
Non-Multiplexed Host Address/Data Full-word (32 Bit) Mode	No	Yes
External Host Address Bus Size (Non-Multiplexed Mode)	N/A	16-bit
HPIC, HPIA, HPID, HPID w. Auto Increment Access Types	Yes	Yes
HCNTL[1:0] Encoding	Different - See Section 1.7.4	
Byte Enables	No	Yes - Optional
Internal Read Buffer Depth	8 Words	8 Words
Internal Write Buffer Depth	8 Words	8 Words
Host Ready Signal Polarity	Active Low ($\overline{\text{HRDY}}$)	Active Low ($\overline{\text{UHPI\_HRDY}}$)
HPIA Access	Host: Read Write	
	DSP: N/A	DSP: Read Only
HPIC Access	Host: Read Write	
	DSP: Read Write	
HPIAMSB, HPIAUMB Upper 16 Address Lines (If Enabled)	N/A	Host: None
	N/A	DSP: Read Write
Memory Regions Accessible through HPI	RAM, EMIF, and Peripherals	Only RAM, ROM, EMIF. Not Peripherals
HPI Memory Range Restriction Option	No	Yes - To a 64K Byte Page controleld by the DSP.

Table 1-8. Comparison of C672x UHPI and C6713 HPI Features (continued)

Feature	C6713 HPI	C672x UHPI
Booting Through UHPI	Supported in hardware.	Requires support in ROM bootloader. May be disabled.
Default HPI State after reset	Depends upon device configuration pin.	Disabled. Must be enabled in software.

1.7.1 New Modes Available on the C672x UHPI

The C672x UHPI supports three major modes of operation:

1. Multiplexed Host Address/Data Half-word (16 Bit) Mode - C672x and C6713
2. Multiplexed Host Address/Data Full-word (32 Bit) Mode - New to C672x
3. Non-Multiplexed Host Address/Data Full-word (32 Bit) Mode - New to C672x

In the first mode, the C672x UHPI behaves almost identically to the C6713 HPI except for the differences noted in this document.

The second mode offers additional bandwidth, extending the UHPI data bus width to 32 bits.

The third mode, Non-Multiplexed, supports separate HA[15:0] Host Address and HD[31:0] Host Data pins. In this mode the UHPI interface is very similar to an asynchronous SRAM (except for the additional $\overline{\text{UHPI\_HRDY}}$ pin). This mode is available to simplify the connection of the DSP UHPI to the DMA controller on an external host device.

1.7.2 Optional Byte Enable Pins **

The C672x UHPI allows an external host to address individual bytes through the UHPI $\overline{\text{UHPI_BE[3:0]}}$pins. This feature is not available on the C6713 DSP.

This feature is supported for 'single' word accesses:

- Multiplexed Modes, HPID Access - No Auto Increment
- Non-Multiplexed HPID Accesses

For burst accesses (Multiplexed Mode, HPID Access - Auto Increment) this feature is restricted. See Section 1.7.3.

When the external host does not require this new feature, the $\overline{\text{UHPI\_BE[3:0]}}$ pins can either be tied low, or configured as general purpose I/O pins. When configured as general purpose I/O pins, the UHPI operates as if the byte enables are active regardless of the actual pin state.

1.7.3 Restrictions on HPID - Auto-Increment Accesses **

The C672x has additional restrictions when using the HPID with Auto-Increment access type. These restrictions are not applicable to the C6713 DSP, because they involve features not available on the C6713 HPI.

When using HPID - Auto-Increment Addressing, three restrictions apply:

- The Auto-Increment Sequence must begin on a word aligned boundary.
- The Auto-Increment Sequence must end on a word aligned boundary.
- The $\overline{\text{UHPI\_BE[3:0]}}$ pins **must be asserted for all bytes within the sequence**.

1.7.4 C672x v.s. C6713 HCNTL[1:0] Encoding Differences **

The UHPI_HCNTL[1:0] encode the same four access types on the C6713 and C672x, however the encoding is different. The two different encodings are listed in Table 1-9 Note that the encodings for HPIA and HPID with Auto-Increment are swapped.

Table 1-9. Encoding Differences HCNTL[1:0]

Access Type	C6713 HCNTL[1:0]	C672x HCNTL[1:0]
HPIC Access	"00"	
HPIA Access	"01"	"10"
HPID Access - Auto Increment	"10"	"01"
HPID Access - No Auto Increment	"11"	

1.7.5 Address Spaces Accessible through HPI/UHPI

The C6713 HPI allows full access to the entire memory space of the device. The only restriction is that reserved areas in the memory map should not be accessed.

On the C672x DSP, the UHPI can only access internal ROM, RAM, and memory on the EMIF.

1.7.6 UHPI Accesses May Be Restricted to 64K Page on C672x

Certain applications require the use of the HPI for high bandwidth I/O but for security reasons do not want to allow an external host complete access to the entire DSP memory. The C672x supports optional CFGHPIAMSB, CFGHPIAUMB registers which allow the DSP to restrict host accesses to the 64 K byte page pointed to by these registers. These configuration registers may only be changed by the DSP, they are not accessible by the external host.

Note that these registers are mandatory in the non-multiplexed mode since only the lower sixteen host address lines are available externally.

*1.7.7 Default HPI State after Reset ***

On the C6713 DSP, the HPI is in one of two states after reset:

- Enabled if bootmode pin HD[14]='1' when $\overline{\text{RESET}}$ is released
- Disabled if HD[14]='0' when $\overline{\text{RESET}}$ is released

And there is no way to change this state on the C6713 DSP without going through another reset cycle.

On the C672x DSP, the UHPI is **disabled** during the assertion of $\overline{\text{RESET}}$. The UHPI also remains disabled until the DSP CPU writes to the CFGHPI register and enables the UHPI. At the same time the major modes are configured through the CFGHPI register. Note that the C672x DSP may also enable and disable the UHPI as needed.

1.8 McASP Differences

1.8.1 McASP Module Configurations

The C672x DSP has three McASPs of different configurations. The C6713 DSP has two identical McASP modules.

Table 1-10. McASP Configuration on C672x DSP Versus C6713 DSP

	C6713 DSP		C672x DSP	
	Serializers	**DIT Support**	**Serializers**	**DIT Support**
McASP0	8	Y	16	N
McASP1	8	Y	6	N
McASP2	not available		2	Y

1.8.2 McASP Pin Multiplexing Differences **

The McASP Pin multiplexing options are different on the C672x DSP from the C6713 DSP.

On the C672x DSP:

- AHCLKR0 and AHCLKR1 share a pin.
- AHCLKX0 and AHCLKX2 share a pin
- AMUTE2 and $\overline{\text{HINT}}$ share a pin
- McASP1 data pins AXR1[5:0] share pins with McASP0 data pins AXR0[8:15]
- McASP2 data pins AXR2[1:0] share pins with McASP0 data pins AXR0[14:15]

1.8.3 McASP Odd/Even Events Not Supported

The McASP module supports separate Odd/Even DMA requests. Even requests occur during even time slots (left for I2S) and odd requests during odd time slots (right for I2S). This feature is available for use on C6713 and can be used to demultiplex data from left and right channels by using two of the inherently 2D EDMA channels for each direction.

On the C672x DSP, the dMAX has the capability to perform 3D and higher transfers. The dMAX can also demultiplex TDM streams with > 2 channels per pin using only a single DMA request. Therefore support for separate Odd/Even McASP events has been removed.

1.8.4 AMUTEINx pin multiplexing is configurable

The McASP AMUTEINx signal on C672x can be multiplexed with one of seven different device pins. Also, the same device pin can be tied to more than one AMUTEINx signal. The pin multiplexing on C672x versus C6713 is given in

Table 1-11. AMUTEINx Multiplexing on C672x Versus C6713

C6713	C672x	
AMUTEIN0 multiplexed with GP0[5]/(EXT_INT5)	AMUTEIN0	Disabled
AMUTEIN1 multiplexed with GP0[4]/(EXT_INT4)		AXR0[7]/SPI1_CLK
	AMUTEIN1	AXR0[8]/AXR1[5]/SPI1_SOMI
	AMUTEIN2	AXR0[9]/AXR1[4]/SPI1_SIMO
	Individually Selectable from:	AHCLKR2
		SPI0_SIMO
		$\overline{\text{SPI0\_SCS}}$/I2C1_SCL
		$\overline{\text{SPI0\_ENA}}$/I2C1_SDA

1.8.5 No support for Emulation Suspend **

The C672x DSP does not support emulation suspend like the C6713 DSP does. This means that the state of the peripherals on the C672x DSP may change during an emulation read operation (which typically occurs during updates of a "Memory" window during a debug session).

The McASP has several registers which are affected by reading them. Viewing these registers during a debug session should be avoided since doing so causes the McASP module to change state differently than when running without a debugger attached.

Table 1-12. McASP Registers to Avoid Viewing During Debug

McASP	Register(s)	Address(x)	Effect of Viewing through Debugger
0	RBUF0 - RBUF15	0x4400_0280 - 0x4400_029C	McASP thinks it is serviced by debugger. Changes status to indicate Receiver Buffer Empty.
1	RBUF0 - RBUF5	0x4500_0280 - 0x4500_029C	
2	RBUF0 - RBUF1	0x4600_0280 - 0x4600_029C	
0	DMA Port	0x5400_0000 - 0x54FF_FFFF	McASP thinks it is serviced by debugger. Changes status to indicate Receiver Buffer Empty. May also trigger RXDMAERR interrupt if the CPU/dMAX also services the McASP.
1	DMA Port	0x5500_0000 - 0x55FF_FFFF	
2	DMA Port	0x5600_0000 - 0x56FF_FFFF	

1.9 No McBSP Modules on C672x

The C6713 DSP family has up to two McBSP modules multiplexed with other peripherals on the device. These could be used as either TDM serial ports or as control ports through the McBSP SPI emulation mode.

The C672x DSP does not have any McBSP modules. Instead, compared to the C6713 DSP it has an additional McASP module as well as two dedicated SPI ports. The SPI ports are much more versatile than the McBSP SPI emulation mode.

While the McASP modules are in general more suited to audio applications than McBSP, it should be noted that the McASP does not support some McBSP features:

- µ-Law / A-Law Companding in Hardware
- AC-97 Codec Framing
- Multichannel TDM greater than 32 channels/pin

1.10 New C672x Module - SPI

The C672x DSP has two full SPI modules. The C6713 DSP used the McBSP in SPI mode to emulate a subset of SPI features.

1.10.1 SPI Module Key Features

Some of the key features of the SPI module are:

- SPI Master and Slave Modes
- Operates at up to 10MHz (or 1/8 SYSCLK2, whichever is smaller)
- Sixteen Bit Shift Register and Buffer
- Clock Phase and Polarity Options
- Optional Chip Select Pin
- Optional Enable Pin (Hardware Handshake)
- All pins are GPIO Capable

The SPI operates in either 3 Pin, 4 Pin Enable, 4 Pin Chip Select, or 5 Pin Mode.

Unlike the McBSP SPI Emulation Mode, it is not necessary to toggle the C672x SPI chip select

1.10.2 SPI Module Chip Select Pin

In Master Mode, the C672x Chip Select pin can be configured to be either toggled or held between words on a multiple word transfer. The McBSP SPI Emulation mode does not have this feature.

In Slave Mode, the C672x Chip Select does not need to be toggled after each word in a multiple word transfer. This is required by the McBSP SPI Emulation mode.

1.10.3 No Support for Emulation Suspend **

The C672x DSP does not support emulation suspend like the C6713 DSP does. This means that the state of the peripherals on the C672x DSP may change during an emulation read operation (which typically occurs during updates of a "Memory" window during a debug session).

The SPI module has several registers which are affected by reading them. Viewing these registers during a debug session should be avoided since doing so causes the SPI module to change state differently than when running without a debugger attached.

Table 1-13. SPI Registers to Avoid Viewing During Debug

SPI	Register(s)	Address(x)	Effect of Viewing through Debugger
0	SPIFLG	0x4700 0010	Reads to this register clear any interrupt flags flags currently set.
1		0x4800 0010	
0	SPIBUF	0x4700 0040	Reads clear the receive data buffer and any interrupt flags currently set. Use SPIEMU (offset 0x44) instead to view the receive buffer without emptying it. Note that the interrupt flags are not visible through SPIEMU.
1		0x4800 0040	
0	TGINTVEC0	0x4700 0060	Reads clear the interrupt vector currently displayed. If the vector is "Receiver Overrun Interrupt" or "Receive Interrupt" the corresponding status flag bits in SPIFLG are cleared as well.
1		0x4800 0060	
0	TGINTVEC1	0x4700 0064	Reads clear the interrupt vector currently displayed. If the vector is "Receiver Overrun Interrupt" or "Receive Interrupt" the corresponding status flag bits in SPIFLG are cleared as well.
1		0x4800 0064	

1.11 C6713 Timer 0,1 replaced by C672x RTI Module

The C6713 DSP included two timer modules: Timer 0 and Timer 1. This functionality has been replaced on the C672x DSP with the RTI (Real Time Interrupt) module except for the C6713 TINP0/1 and TOUT0/1 pins. A comparison between the two modules is given in Table 1-14.

Table 1-14. Comparison of C6713 Timers to C672x RTI

Feature	C6713 Timer	C672x RTI
Number of Modules on Device	2	1
Counters Per Module	1	2
Main Counter Width	32 Bits	32 Bits
Additional Counter Prescale	No	Yes - Up to 32 Bits
Compare Events	1 (DMA Requests to EDMA or Interrupts to CPU)	4 (DMA Requests to dMAX or Interrupts to CPU)
Input Captures	No	2 (Measures McASP Data Rates). Captures Main Counter and Prescale Counter Values
External Pulse Accumulate Mode / Input Pin	1 Pin	No Pins
External Clock Generation / Output Pin	1 Pin - Tied to Compare Event	No Pins
Digital Watchdog	No	25 Bit Down Counter, 12 MSBs Programmable, Two 16-Bit Keys (0xE51A followed by 0xA35C) to service. Must be initially enabled.

Note that the C672x RTI counters are completely free running. The compare functions are implemented by adding the compare period to the compare value upon each comparison match. This avoids resetting the main counters and allows all four compares as well as the input captures to operate simultaneously. In addition, the counters may be read by software for timestamping purposes.

Finally, note that on both the C6713 and C672x DSPs, one counter and compare function is needed to generate the TICK event for DSP BIOS.

1.12 I2C Module Differences

The C672x I2C module shares the same underlying design as the C6713 I2C Module, however it is a new version of the module with many enhancements. This section gives an overview of the enhancements.

1.12.1 C672x I2C Module is GPIO Capable

The C672x I2C module has been enhanced over the C6713 I2C module through the addition of GPIO capability. GPIO capability is controlled by the set of new GPIO registers: I2CPFUNC, I2CPDIR, I2CPDIN, I2CPDOUT, I2CPDSET, and I2CPDCLR.

1.12.2 Two Options for ICXRDY Interrupt Generation in Slave Transmitter Mode

The C672x I2C module has two options for timing the generation of the ICXRDY interrupt in slave transmitter mode (Slave transmitter mode is entered when an external master reads from the slave I2C module). The initial ICXRDY interrupt is generated by the I2C module as soon as it enters slave transmitter mode.

However, there are two options for the generation of subsequent interrupts. These two options are controlled by the XRDYM bit in the new I2CEMDR register. The options are summarized in Table 1-15.

Table 1-15. I2C ICXRDY Generation Options in Slave Transmitter Mode

XRDYM	ICXRDY Generated	Advantage	Disadvantage	Comment
0 (Default)	At the end of data transfer only if master generates an ACK requesting more data.	Generates only the number of interrupts required.	Waiting for ACK before generating ICXRDY means slower overall transfers.	New default for C672x
1	As soon as ICXDR is copied to ICXSR.	Faster transfers since ICXRDY is generated as early as possible.	Generates one more interrupt than is required; so output data must be padded by one dummy byte.	Compatible with C6713

1.12.3 Change to BB Behavior when I2C is in Reset

The issue described in TMS320C6713, TMS320C6713B Digital Signal Processors Silicon Errata (SPRZ191) as "I2C: Bus Busy Bit Does Not Reflect the State of the I2C Bus When the I2C is in Reset" has been resolved however the workaround procedure described in the errata is now mandatory. On the C672x DSP, the I2C BB bit will reflect '0' whenever the I2C is held in reset due to the IRS bit being set to '0'. This is different than the C6713 I2C module which would hold the previous value of BB from before the IRS bit was set. The I2C module still does not track the I2C bus state while it is in reset, however now it always indicates 'not busy'.

1.12.4 Change to Conditions that Clear AAS bit

As described in TMS320C6713, TMS320C6713B DSPs Silicon Errata (SPRZ191), the AAS bit on the C6713 I2C would always be set properly but there were some conditions under which it would remain set instead of being cleared. To resolve this issue, the C672x I2C module has been updated to clear AAS whenever one of the two conditions occurs:

- STOP condition is detected on the I2C Bus
- Address Byte that does not match I2COAR is received

1.12.5 Additional Interrupts Improve I2C Response Tme in Slave Mode

The C672x I2C module has two additional interrupt sources which improve the response time of the module in slave mode:

- STOP Condition Detection Interrupt
- Address As Slave Detection Interrupt

These two interrupt sources have been added to the I2CIER and I2CISR registers.

1.12.6 I2CDRR and I2CDXR Values when transfer length is less than 8 bits

On the C6713 I2C module, the I2CDRR and I2CDXR would read back a concatenation of previous bits and current bits when transfers of less than 8 bits are selected. Masking off the unused bits needs to be done in software.

On the C672x, the I2CDRR and I2CDXR now read back the current bits padded with '0' bits in the unused bit positions, so masking in software is no longer needed.

1.12.7 I2CLKL/I2CLKH Registers May Need Adjustment (SYSCLK2 is faster) **

The dividers I2CLKL/I2CLKH may need to be adjusted when migrating from the C6713 DSP to the C672x DSP due to the likely increase in SYSCLK2 frequency. The divider combination needs to be set to between 7MHz and 12MHz.

1.12.8 No support for Emulation Suspend **

The C672x DSP does not support emulation suspend like the C6713 DSP does. This means that the state of the peripherals on the C672x DSP may change during an emulation read operation (which typically occurs during updates of a "Memory" window during a debug session).

The I2C module has several registers which are affected by reading them. Viewing these registers during a debug session should be avoided since doing so causes the I2C module to change state differently than when running without a debugger attached.

Table 1-16. I2C Registers to Avoid Viewing During Debug

I2C	Register(s)	Address(x)	Effect of Viewing through Debugger
0	I2CDRR	0x4900_0000	Clears Receive Data Ready Interrupt (ICRRDY) in I2CSTR. May cause incoming data to be dropped.
1	I2CDRR	0x4A00_0000	
0	I2CISR	0x4900_0000	Clears the interrupt status code. Application code will not be able to determine the reason for the interrupt.
1	I2CISR	0x4A00_0000	

1.12.9 C672x I2C Module uses different output buffers than on C6713

On the C6713 DSP, the I2C Module used open-drain output buffers specifically designed to meet the electrical requirements of the I2C Specification.

On the C672x DSP, the I2C Module is multiplexed with other peripherals and uses a +/-8mA LVCMOS I/O buffer. The I2C pins default to a 3-stated condition after reset. If controlled by the I2C module, the module will switch these pins from 3-state to driving low simulating the operation of an open drain buffer. However, the edge rates will be significantly faster than a normal I2C buffer. The system designer may decide to add series resistors to the I2C bus pins on C672x DSP in order to slow down the transition times on the bus.

1.13 Changes to the PLL Controller and addition of an On-chip Oscillator

The C6713 and C672x DSPs both include software programmable PLL. This section describes the differences in implementation between the two DSPs and also describes the C672x on-chip oscillator.

*1.13.1 On-chip Oscillator ***

The C672x DSP includes an on-chip 1.2-V oscillator that supports external crystals in the range of 12 MHz to 25 MHz. Either this 1.2-V oscillator can be used with the OSCIN pin to provide the DSP clock, or a 3.3V logic level external clock can provide the DSP clock through the CLKIN pin. The DSP input clock is the logical 'OR' of the signal at OSCIN and CLKIN. There is no additional pin to select between clock sources. It is a requirement to drive a clock into only one of the two pins, while tying the other pin to V_{SS}.

Note that improperly handling the unselected pin (for example, by allowing it to float high or driving it high) could prevent the DSP from receiving clocks entirely.

1.13.2 PLL Control Register Changes

The PLL controller on the C672x DSP has additional registers and different default values than that on the C6713 DSP. Also, note that the base address of the PLL controller is different on C672x (base address 0x4100_0000) then that on C6713 (base address 0x01B7_C000).

Table 1-17. PLL Registers on C6713 and C672x

Register Name	C6713	C672x	Differences
PLLPID	Y	Y	Same ID Value. However Modules are Different.
PLLCSR	Y	Y	PLLPWRDN moved from bit 1 to bit 4. Default value of the register is different.
PLLM	Y	Y	Defaults to times 13 on C672x. Defaults to times 7 on C6713.
PLLDIV0	Y	Y	same
PLLDIV1	Y	Y	same
PLLDIV2	Y	Y	same
PLLDIV3	Y	Y	Defaults to /3 on C672x. Defaults to /2 on C6713
OSCDIV1	Y	N	Not available on C672x. C672x does not have a CLKOUT3 pin.
PLLCMD	N	Y	Used for clock edge alignment. New feature on C672x.
PLLSTAT	N	Y	Used for clock edge alignment. New feature on C672x.
ALNCTL	N	Y	Used for clock edge alignment. New feature on C672x.
CKEN	N	Y	Allows AUXCLK to be disabled. If AUXCLK is disabled, McASP operation may be affected.
CKSTAT	N	Y	Shows status of AUXCLK
SYSTAT	N	Y	Shows status of SYSCLK1, SYSCLK2, and SYSCLK3

*1.13.3 Alignment of clocks using ALNCTL and GO bit is required ***

The C6713 DSP has no requirements on the relative phases of SYSCLK1, SYSCLK2, and SYSCLK3. However, the C672x DSP has a requirement that these clocks be aligned internally.

The PLL Controller on C672x has an additional feature to support alignment of the SYSCLKs.

The ALNCNTL register should be configured to edge-align SYSCLK1, SYSCLK2, and SYSCLK3. After reset, a value of 0x07 should be written to ALNCNTL to select alignment between SYSCLK1, SYSCLK2, and SYSCLK3. Then after the clock dividers are set to the desired values, the GOSET bit in the PLLCMD register must be written to have the divider changes actually take effect. This step automatically aligns the three SYSCLKs if the ALNCNTL register is configured properly.

1.13.4 Bridge Reset Required on Startup and After PLL Changes

Refer to Section 1.3.7.

1.13.5 PLLPWRDN bit location has moved in PLLCSR register, OSCPWRDN Added

The definition of the PLLCSR register has changed from C6713 to C672x, specifically with respect to the PLLPWRDN bit and the addition of the OSCPWRDN bit.

On C6713 the PLLCSR register is defined as:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									STABLE	Reserved		PLLRST	Rsvd	PLL PWRDN	PLLEN
R-0									R-X	R-0		R/W-1	R/W-0	R/W-0	R/W-0

LEGEND: R = Read, W = Write, n = value at reset

Figure 1-1. C6713 PLLCSR Register

On C672x the PLLCSR register is defined as:

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															
R-0															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									STABLE	Rsvd	PLL PWRDN	PLLRST	OSC PWRDN	Rsvd	PLLEN
R-0									R-X	R-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0

LEGEND: R = Read, W = Write, n = value at reset

Figure 1-2. C672x PLLCSR Register

1.13.6 No CLOCKOUT2, CLKOUT3 pins

The C672x DSP does not have CLKOUT2 or CLKOUT3 pins. The only free running external clock available on the C672x DSP is the EM_CLK pin which runs at the SYSCLK3 rate. (Some modules such as the McASP and SPI can be configured to drive out a divided down clock, these are not considered in this discussion.)

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information 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.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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