TMS320C642x DSP
Power and Sleep Controller (PSC)

User's Guide

Literature Number: SPRUEN8A
February 2008

Texas Instruments

Contents

List of Figures

List of Tables

Read This First

About This Manual

Describes the operation of the power and sleep controller (PSC) in the TMS320C642x Digital Signal Processor (DSP).

Notational Conventions

This document uses the following conventions.

- Hexadecimal numbers are shown with the suffix h. For example, the following number is 40 hexadecimal (decimal 64): 40h.
- Registers in this document are shown in figures and described in tables.
  - Each register figure shows a rectangle divided into fields that represent the fields of the register. Each field is labeled with its bit name, its beginning and ending bit numbers above, and its read/write properties below. A legend explains the notation used for the properties.
  - Reserved bits in a register figure designate a bit that is used for future device expansion.

Related Documentation From Texas Instruments

The following documents describe the TMS320C642x Digital Signal Processor (DSP). Copies of these documents are available on the Internet at www.ti.com. *Tip:* Enter the literature number in the search box provided at www.ti.com.

The current documentation that describes the C642x DSP, related peripherals, and other technical collateral, is available in the C6000 DSP product folder at: www.ti.com/c6000.

SPRUEM3 — *TMS320C642x DSP Peripherals Overview Reference Guide.* Provides an overview and briefly describes the peripherals available on the TMS320C642x Digital Signal Processor (DSP).

SPRAA84 — *TMS320C64x to TMS320C64x+ CPU Migration Guide.* Describes migrating from the Texas Instruments TMS320C64x digital signal processor (DSP) to the TMS320C64x+ DSP. The objective of this document is to indicate differences between the two cores. Functionality in the devices that is identical is not included.

SPRU732 — *TMS320C64x/C64x+ DSP CPU and Instruction Set Reference Guide.* Describes the CPU architecture, pipeline, instruction set, and interrupts for the TMS320C64x and TMS320C64x+ digital signal processors (DSPs) of the TMS320C6000 DSP family. The C64x/C64x+ DSP generation comprises fixed-point devices in the C6000 DSP platform. The C64x+ DSP is an enhancement of the C64x DSP with added functionality and an expanded instruction set.

SPRU871 — *TMS320C64x+ DSP Megamodule Reference Guide.* Describes the TMS320C64x+ digital signal processor (DSP) megamodule. Included is a discussion on the internal direct memory access (IDMA) controller, the interrupt controller, the power-down controller, memory protection, bandwidth management, and the memory and cache.

Power and Sleep Controller (PSC)

1 Power Management

1.1 Overview

In many applications, there may be specific requirements to minimize power consumption for both power supply (or battery) and thermal considerations. There are two components to power consumption: active power and leakage power. Active power is the power consumed to perform work and scales roughly with clock frequency and the amount of computations being performed. Active power can be reduced by controlling the clocks in such a way as to either operate at a clock setting just high enough to complete the required operation in the required timeline or to run at a clock setting until the work is complete and then drastically cut the clocks (that is, to PLL Bypass mode) until additional work must be performed. Leakage power is due to static current leakage and occurs regardless of the clock rate. Leakage, or standby power, is unavoidable while power is applied and scales roughly with the operating junction temperatures. Leakage power can only be avoided by removing power completely from a device or subsystem.

The TMS320C642x DSP has several means of managing the power consumption, as detailed in the following sections. There is extensive use of automatic clock gating in the design as well as software-controlled module clock gating to not only reduce the clock tree power, but to also reduce module power by basically freezing its state while not operating. Clock management enables you to slow the clocks down on the chip in order to reduce switching power. In particular, the C642x DSP includes all of the power management features described in Table 1.

Table 1. Power Management Features

Power Management Features	Description
	Clock Management
PLL power-down	The PLLs can be powered-down when not in use to reduce switching power
Module clock ON/OFF	Module clocks can be turned on/off to reduce switching power
Module clock frequency scaling	Module clock frequency can be scaled to reduce switching power
	DSP Sleep Management
DSP sleep modes	The DSP can be put into sleep mode to reduce switching power
	I/O Management
3.3 Volt I/O power-down	The 3.3 V I/Os can be powered-down to reduce I/O cell power

1.2 PSC and PLLC Overview

The power and sleep controller (PSC) plays an important role in managing system power on/off, clock on/off, and reset. Similarly, the PLL controller (PLLC) plays an important role in device clock generation. For detailed information on the PSC, see Section 2. For detailed information on the PLLC, see the *TMS320C642x DSP Phase-Locked Loop Controller (PLLC) User's Guide* (SPRUES0).

1.3 Clock Management

1.3.1 Module Clock ON/OFF

The module clock on/off feature allows software to disable clocks to module individually, in order to reduce the module's active power consumption to 0. The C642x DSP is designed in full static CMOS; thus, when a module clock stops, the module's state is preserved. When the clock is restarted, the module resumes operating from the stopping point.

Note: Stopping clocks to a module only affects active power consumption, it does not affect leakage power consumption.

If a module's clock(s) is stopped while being accessed, the access may not occur, and could potentially lock-up the device. User must ensure that all of the transactions to the module are finished prior to stopping the clocks. The power and sleep controller (PSC) controls module clock gating. The PSC provides some protection against system hang by monitoring the internal bus activity—it only gates internal clock to the module after checking that there is no access to the module from the internal bus.

The procedure to turn module clocks on/off using the PSC is described in Section 2. Furthermore, special consideration must be given to DSP clock on/off. The procedure to turn the DSP clock on/off is further described in Section 3.2.2.

Some peripherals provide additional power saving features by clock gating components within its module boundary. See peripheral-specific user's guide for more details on these additional power saving features.

1.3.2 Module Clock Frequency Scaling

Module clock frequency is scalable by programming the PLL's multiply and divide parameters. Reducing the clock frequency reduces the active switching power consumption linearly with frequency. It has no impact on leakage power consumption. The *TMS320C642x DSP Phase-Locked Loop Controller (PLLC) User's Guide* (SPRUES0) describes how to program the PLL frequency and the frequency constraints.

1.3.3 PLL Bypass and Power Down

You can bypass the PLLs in the C642x DSP. Bypassing the PLLs sends the PLL reference clock (MXI/CLKIN) instead of the PLL output (PLLOUT) to the SYSCLK dividers (PLLDIV*n*) of the PLLC. The PLL reference clock is between 15 to 30 MHZ; therefore, you can use this mode to reduce the core and module clock frequencies to very low maintenance levels without using the PLL during periods of very low system activity. Furthermore, you can power-down the PLL when bypassing it to save additional active power. The *TMS320C642x DSP Phase-Locked Loop Controller (PLLC) User's Guide* (SPRUES0) describes PLL bypass and PLL power down.

1.4 3.3 V I/O Power Down

The 3.3 V I/O drivers are fabricated out of 1.8 V transistors with design techniques that require a DC bias current. These I/O cells have a power-down mode that turns off the DC current. The VDD3P3V_PWDN register of the System Module controls this standby mode. Refer to the device-specific data manual for more details on the VDD3P3V_PWDN register.

2 Power and Sleep Controller

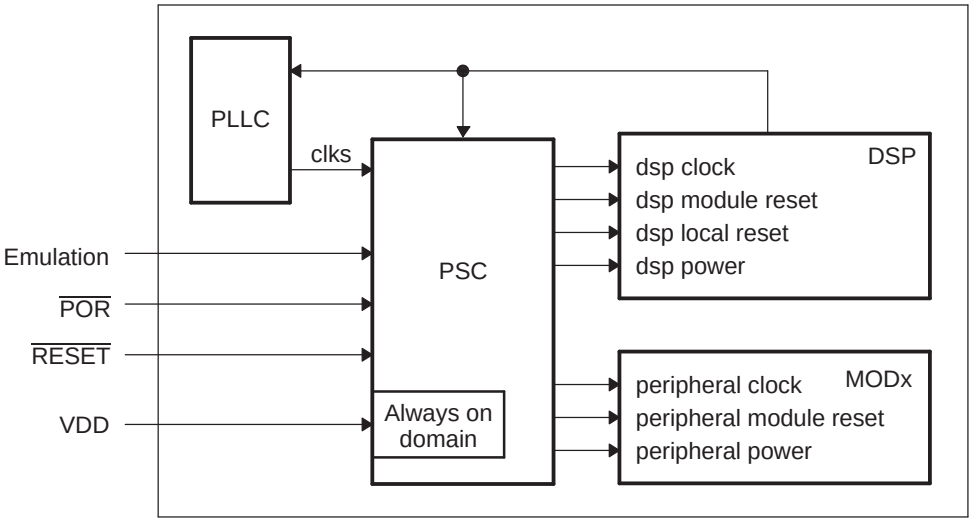
2.1 Introduction

The Power and Sleep Controller (PSC) is responsible for managing transitions of system power on/off, clock on/off, and reset. The C642x DSP only utilizes the clock gating feature of the PSC for power savings. The PSC consists of a Global PSC (GPSC) and a set of Local PSCs (LPSCs). The GPSC contains memory mapped registers, PSC interrupt control, and a state machine for each peripheral/module. An LPSC is associated with each peripheral/module and provides clock and reset control. Figure 1 shows how the PSC is integrated on the device. Many of the operations of the PSC are transparent to software, such as power-on and hard reset operations. However, the PSC provides you with an interface to control several important power, clock, and reset operations. The power, clock, and reset operations are the focus of this chapter.

The PSC includes the following features:

- Manages chip power-on/off and resets
- Provides a software interface to:
  - Control module clock ON/OFF
  - Control module resets
  - Control DSP local reset (CPU reset)
- Supports IcePick emulation features: power, clock, and reset

Figure 1. Power and Sleep Controller (PSC) Integration



NOTE: The effects of DSP local reset and DSP module reset have not been fully validated; therefore, these resets are not supported and should not be used. Instead, the $\overline{\text{POR}}$ or $\overline{\text{RESET}}$ pins should be used to reset the entire DSP.

2.2 Power Domain and Module Topology

The C642x DSP includes one power domain--the AlwaysOn power domain. The AlwaysOn power domain is always on when the chip is on. The AlwaysOn domain is powered by the V_{DD} pins of the C642x DSP (see the device-specific data manual). All of the C642x DSP modules reside within the AlwaysOn power domain. Table 2 lists all the possible peripherals on the C642x DSP, their LPSC assignments, and default module states. Refer to the device-specific data manual for the peripherals available on a given device. The module states are defined in Section 2.3.2.

Table 2. C642x DSP Default Module Configuration

LPSC Number	Module Name	Default Module State (MDSTAT.STATE)
0	Reserved	-
1	Reserved	-
2	EDMACC	SwRstDisable
3	EDMATC0	SwRstDisable
4	EDMATC1	SwRstDisable
5	EDMATC2	SwRstDisable
6	EMAC Memory Controller	SwRstDisable
7	MDIO	SwRstDisable
8	EMAC	SwRstDisable
9	McASP0	SwRstDisable
10	Reserved	-
11	VLYNQ	SwRstDisable
12	HPI	SwRstDisable
13	DDR2 Memory Controller	SwRstDisable
14	EMIFA	SwRstDisable if configuration pins AEM[2:0] = 000b Enable if configuration pins AEM[2:0] = others
15	PCI	SwRstDisable
16	McBSP0	SwRstDisable
17	McBSP1	SwRstDisable
18	I2C	SwRstDisable
19	UART0	SwRstDisable
20	UART1	SwRstDisable
21	Reserved	-
22	Reserved	-
23	PWM0	SwRstDisable
24	PWM1	SwRstDisable
25	PWM2	SwRstDisable
26	GPIO	SwRstDisable
27	TIMER0	SwRstDisable
28	TIMER1	SwRstDisable
29-38	Reserved	-
39	C64x+ CPU	Enable
40	Reserved	-

2.3 Power Domain and Module States

Note: The effects of DSP local reset and DSP module reset have not been fully validated; therefore, these resets are not supported and should not be used. Instead, the $\overline{\text{POR}}$ or $\overline{\text{RESET}}$ pins should be used to reset the entire DSP.

Table 2 shows the state of each module after chip Power-on Reset ($\overline{\text{POR}}$), Warm Reset ($\overline{\text{RESET}}$), or Max Reset. These states are defined in the following sections.

2.3.1 Power Domain States

A power domain can only be in one of two states: ON or OFF, defined as follows:

- ON: power to the power domain is on.
- OFF: power to the power domain is off.

In the C642x DSP, the AlwaysOn Power Domain is always in the ON state when the chip is powered-on.

2.3.2 Module States

A module can be in one of four states: Disable, Enable, SyncReset, or SwRstDisable. These four states correspond to combinations of module reset asserted or de-asserted and module clock on or off, as shown in Table 3.

Table 3. Module States

Module State	Module Reset	Module Clock	Module State Definition
Enable	De-asserted	On	A module in the enable state has its module reset de-asserted and it has its clock on. This is the normal run-time state for a given module.
Disable	De-asserted	Off	A module in the disable state has its module reset de-asserted and it has its clock off. This state is typically used for disabling a module clock to save power. The C642x DSP is designed in full static CMOS, so when you stop a module clock, it retains the module's state. When the clock is restarted, the module resumes operating from the stopping point.
SyncReset	Asserted	On	A module in the SyncReset state has its module reset asserted and it has its clock on. Generally, software is not expected to initiate this state.
SwRstDisable	Asserted	Off	A module in the SwResetDisable state has its module reset asserted and it has its clock set to off. After initial power-on, most modules are in the SyncRst state by default (see Table 2). Generally, software is not expected to initiate this state.

Note: Module Reset is defined to completely reset a given module, so that all hardware returns to its default state. See Section 2.3.3 for more information on module reset.

For more information on power management, see Section 1.

2.3.3 Module Reset and Local Reset

From PSC's perspective, there are two types of resets to a module/peripheral:

- Module Reset
- Local Reset

Module Reset is defined to completely reset a given module/peripheral, so that all hardware returns to its default state. The DSP or external host software can initiate a Module Reset by programming the PSC to put the module in SyncReset or SwRstDisable state. Module Reset is intended as a debug tool, not necessarily as a tool to use in production.

In addition to Module Reset, the DSP CPU can be reset using a special Local Reset. When DSP Local Reset is asserted, the DSPs' internal memories (L1P, L1D, and L2) are still accessible. The Local Reset only resets the DSP CPU core, not the rest of the C64x+ Megamodule, as the DSP Module Reset would.

Module Reset takes precedence over Local Reset; therefore, Local Reset is not useful when the DSP is in SyncReset or SwRstDisable state.

See Section 3.1 for more information on DSP Module Reset vs. Local Reset, and scenarios where they can be used.

2.4 Executing State Transitions

This section describes how to execute state transitions for device modules.

2.4.1 Power Domain State Transitions

The C642x DSP consists of only one power domain--the AlwaysOn power domain. This AlwaysOn Power Domain is always in the ON state when the chip is powered-on. You are not allowed to change this power domain state to OFF.

2.4.2 Module State Transitions

This section describes the procedure for transitioning the module state. All C642x DSP modules are on the AlwaysOn domain (Power Domain 0).

Note that some peripherals have special programming requirements and steps you must take before you can invoke the PSC module state transition. Refer to the individual peripheral reference guide for more details. For example, the DDR2 memory controller requires that you first place the DDR memory in self-refresh mode before you invoke the PSC module state transition, if you want to maintain the memory content.

Note: The following procedure is directly applicable for all modules, except for the DSP in the C642x DSP. To transition the DSP module state, you must be aware of several system considerations.

The procedure for module state transitions is as follows (where *n* corresponds to the module):

1. Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. You must wait for any previously initiated transitions to finish before initiating a new transition.
2. Set the NEXT bit in MDCTL*n* to SwRstDisable (0), SyncReset (1), Disable (2h), or Enable (3h).

   Note: You may set transitions in multiple NEXT bits in MDCTL*n* in this step. Transitions do not actually take place until you set the GO[0] bit in PTCMD in a later step.
3. Set the GO[0] bit in PTCMD to 1 to initiate the transition(s).
4. Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. The modules are safely in the new states only after the GOSTAT[0] bit in PTSTAT is cleared to 0.

2.5 IcePick Emulation Support in the PSC

The PSC supports IcePick commands that allow IcePick aware emulation tools to have some control over the state of power domains and modules. On the C642x DSP, this IcePick support only applies to the C64x+ CPU (Module number 39 in the AlwaysOn power domain 0).

In particular, Table 4 shows IcePick emulation commands recognized by the PSC, and indicated ones that apply to the C64x+ CPU on the C642x DSP.

Table 4. IcePick Emulation Commands

Power On and Enable Features	Power On and Enable Descriptions	Reset Features	Reset Descriptions
Inhibit Sleep	Allows emulation to prevent software from transitioning the module out of the enable state. Applicable to the C642x DSP.	Assert Reset	Allows emulation to assert the module's local reset. Applicable to the C642x DSP.
Force Power	Allows emulation to force the power domain into an on state. Not applicable on the C642x DSP as AlwaysOn power domain is always on.	Wait Reset	Allows emulation to keep local reset asserted for an extended period of time after software initiates local reset de-assert. Applicable to the C642x DSP.
Force Active	Allows emulation to force the module into the enable state. Applicable to the C642x DSP.	Block Reset	Allows emulation to block software initiated local and module resets. Applicable to the C642x DSP.

Note: When emulation tools remove the above commands, the PSC immediately executes a state transition based on the current values in the NEXT bit in PDCTL0 and the NEXT bit in MDCTL*n*, as set by software.

2.6 PSC Interrupts

The PSC has an interrupt that is tied to the C64x+ interrupt controller (INTC). This interrupt is named PSCINT in the interrupt map. The PSC interrupt is generated when certain IcePick emulation events occur.

2.6.1 Interrupt Events

The PSC interrupt is generated when any of the following events occur:

- Module State Emulation Event
- Module Local Reset Emulation Event

These interrupt events are summarized in Table 5 and described in more detail in this section.

Table 5. PSC Interrupt Events

Interrupt Enable Bits		
Control Register	Status Bit	Interrupt Condition
MDCTL*n*	EMUIHB	Interrupt occurs when the emulation alters the module state.
MDCTL*n*	EMURST	Interrupt occurs when the emulation alters the module's local reset.

The PSC interrupt events only apply when IcePick emulation alters the state of the module from the user-programmed state in the NEXT bit in MDCTL*n*. As discussed in Section 2.5, on the C642x DSP, IcePick support only applies to the C64x+ CPU (module 39), therefore the PSC interrupt condition only applies to module 39.

The C642x DSP is a single-processor device. The C64x+ CPU must not program its own module state. The C64x+ CPU module state can only be programmed by an external host (for example, PCI, HPI). As a result, interrupt events listed in Table 5 can only occur in the scenario where an external host programs the C64x+ CPU module state but the emulator alters that desired state.

2.6.1.1 Module State Emulation Events

A module state emulation event occurs when emulation alters the state of a module. Status is reflected in the EMUIHB bit in MDSTAT*n*. In particular, a module state emulation event occurs under the following conditions:

- When inhibit sleep is asserted by emulation and software attempts to transition the module out of the enable state.
- When force active is asserted by emulation and module is not already in the enable state.

2.6.1.2 Local Reset Emulation Events

A local reset emulation event occurs when emulation alters the local reset of a module. Status is reflected in the EMURST bit in MDSTAT*n*. In particular, a module local reset emulation event occurs under the following conditions:

- When assert reset is asserted by emulation although software de-asserted the local reset.
- When wait reset is asserted by emulation.
- When block reset is asserted by emulation and software attempts to change the state of local reset.

2.6.2 Interrupt Registers

The PSC interrupt enable bits are the EMUIHBIE bit in MDCTL39 and the EMURSTIE bit in MDCTL39.

Note: To interrupt the DSP, the power and sleep controller interrupt (PSCINT) must also be enabled in the DSP interrupt controller. See *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871) for more information on the interrupt controller.

The PSC interrupt status bits are the M[39] bit in MERRPR1, the EMUIHB bit in MDSTAT39, and the EMURST bit in MDSTAT39. The status bit in MERRPR1 is read by software to determine which module has generated an emulation interrupt, and then software can read the corresponding status bits in MDSTAT39 to determine which event caused the interrupt.

The PSC interrupt clear bit is the M[39] bit in MERRCR1.

The PSC interrupt evaluation bit is the ALLEV bit in INTEVAL. When set, this bit forces the PSC interrupt logic to re-evaluate event status. If any events are still active (if any status bits are set) when the ALLEV bit in INTEVAL is set to 1, the PSCINT is re-asserted to the DSP interrupt controller. Set the ALLEV bit in INTEVAL before exiting your PSCINT interrupt service routine to ensure that you do not miss any PSC interrupts.

See Section 2.7 for complete descriptions of all PSC registers.

2.6.3 Interrupt Handling

Handle the PSC interrupts as described in the following procedure:

First, enable the interrupt.

1. Set the EMUIHBIE bit and the EMURSTIE bit in MDCTL39 to enable the interrupt events that you want.

Note: The PSC interrupt PSCINT is sent to the DSP interrupt controller when at least one enabled event becomes active.

2. Enable the power and sleep controller interrupt (PSCINT) in the DSP interrupt controller. To interrupt the DSP, PSCINT must be enabled in the DSP interrupt controller. See *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871) for more information.

The DSP enters the interrupt service routine (ISR) when it receives the interrupt.

1. Read the M*n* bit in MERRPR1 to determine the source of the interrupt(s). Note that on the C642x DSP, only M[39] can cause an interrupt.
2. For each active event that you want to service:
   - Read the event status bits in MDSTAT39, depending on the status bits read in the previous step to determine the event that caused the interrupt.
   - Service the interrupt as required by your application.
   - Write the M[39] bit in MERRCR1 to clear corresponding status.
   - Set the ALLEV bit in INTEVAL to 1. Setting this bit reasserts the PSCINT to the DSP interrupt controller, if there are still any active interrupt events.

2.7 PSC Registers

Table 6 lists the memory-mapped registers for the PSC. See the device-specific data manual for the memory address of these registers.

Table 6. Power and Sleep Controller (PSC) Registers

Offset	Register	Description	Section
0h	PID	Peripheral Revision and Class Information Register	Section 2.7.1
18h	INTEVAL	Interrupt Evaluation Register	Section 2.7.2
44h	MERRPR1	Module Error Pending Register 1 (mod 32-63)	Section 2.7.3
54h	MERRCR1	Module Error Clear Register 1 (mod 32-63)	Section 2.7.4
120h	PTCMD	Power Domain Transition Command Register	Section 2.7.5
128h	PTSTAT	Power Domain Transition Status Register	Section 2.7.6
200h	PDSTAT[0]	Power Domain Status 0 Register	Section 2.7.7
300h	PDCTL[0]	Power Domain Control 0 Register	Section 2.7.8
800h-89Ch	MDSTAT0-39	Module Status *n* Register	Section 2.7.9
A00h-A9Ch	MDCTL0-39	Module Control *n* Register	Section 2.7.10

2.7.1 Peripheral Revision and Class Information Register (PID)

The peripheral revision and class information (PID) register is shown in Figure 2 and described in Table 7.

Figure 2. Peripheral Revision and Class Information Register (PID)

31	30	29	28	27	16
SCHEME		Reserved		FUNC	
R-1		R-0		R-482h	

15	11	10	8	7	6	5	0
RTL		MAJOR		CUSTOM		MINOR	
R-4h		R-1		R-0		R-5h	

LEGEND: R = Read only; -*n* = value after reset

Table 7. Peripheral Revision and Class Information Register (PID) Field Descriptions

Bit	Field	Value	Description
31-30	SCHEME	0-3h	Distinguishes between the old scheme and the current scheme. There is a spare bit to encode future schemes.
29-28	Reserved	0	Reserved
27-16	FUNC	0-FFFh	Indicates a software compatible module family.
15-11	RTL		RTL version.
		4h	Current RTL version.
10-8	MAJOR		Major revision.
		1h	Current major revision.
7-6	CUSTOM	0-3h	Indicates a special version for a particular device.
5-0	MINOR		Minor revision.
		5h	Current minor revision.

2.7.2 Interrupt Evaluation Register (INTEVAL)

The interrupt evaluation register (INTEVAL) is shown in Figure 3 and described in Table 8.

Figure 3. Interrupt Evaluation Register (INTEVAL)

31	16
Reserved	
R-0	

15	1	0
Reserved		ALLEV
R-0		W-0

LEGEND: R = Read only; W= Write only; -*n* = value after reset

Table 8. Interrupt Evaluation Register (INTEVAL) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved
0	ALLEV		Evaluate PSC interrupt.
		0	A write of 0 has no effect.
		1	A write of 1 re-evaluates the interrupt condition.

2.7.3 Module Error Pending Register 1 (mod 39) (MERRPR1)

The module error pending register 1 (mod 39) (MERRPR1) is shown in Figure 4 and described in Table 9. Only the C64x+ DSP (module 39) can have an error condition, as it is the only module with IcePick support. See Section 2.5 for more information.

Figure 4. Module Error Pending Register 1 (mod 39) (MERRPR1)

31	16
Reserved	
R-0	

15	8	7	6	0
Reserved		M[39]	Reserved	
R-0		R-0	R-0	

LEGEND: R = Read only; -*n* = value after reset

Table 9. Module Error Pending Register 1 (mod 39) (MERRPR1) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved
7	M[39]		Module interrupt status bit for module 39 (C64x+ DSP).
		0	Module 39 does not have an error condition.
		1	Module 39 has an error condition. See the module status 39 register (MDSTAT39) for the exact error condition.
6-0	Reserved	0	Reserved

2.7.4 Module Error Clear Register 1 (mod 39) (MERRCR1)

The module error clear register 1 (mod 39) (MERRCR1) is shown in Figure 5 and described in Table 10.

Figure 5. Module Error Clear Register 1 (mod 39) (MERRCR1)

31	16
Reserved	
R-0	

15	8	7	6	0
Reserved		M[39]	Reserved	
R-0		W-0	R-0	

LEGEND: R = Read only; W = Write only; -*n* = value after reset

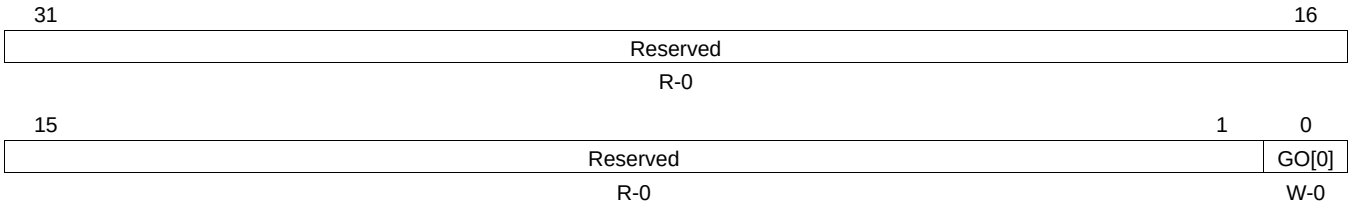
Table 10. Module Error Clear Register 1 (mod 39) (MERRCR1) Field Descriptions

Bit	Field	Value	Description
31-8	Reserved	0	Reserved
7	M[39]		Clears the interrupt status bits set in the corresponding module error pending register 1 (mod 39) (MERRPR1) and the module status 39 register (MDSTAT39). This pertains to module 39.
		0	A write of 0 has no effect.
		1	Clears module interrupt status bits: the M[39] bit in MERRPR1, the EMURST bit and the EMUIHB bit in MDSTAT39.
6-0	Reserved	0	Reserved

2.7.5 Power Domain Transition Command Register (PTCMD)

The power domain transition command register (PTCMD) is shown in Figure 6 and described in Table 11.

Figure 6. Power Domain Transition Command Register (PTCMD)



LEGEND: R = Read only; W = Write only; -*n* = value after reset

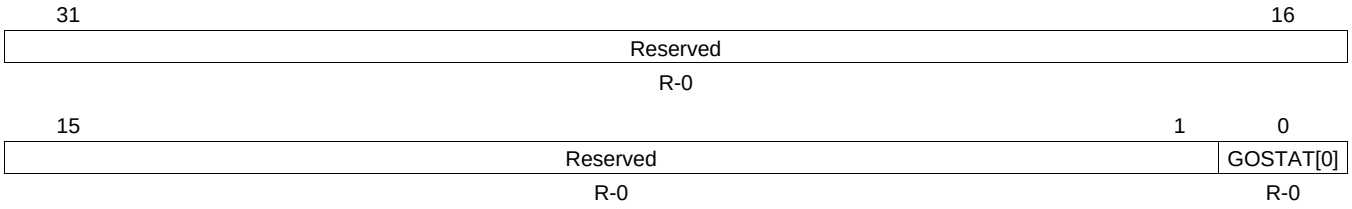
Table 11. Power Domain Transition Command Register (PTCMD) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved
0	GO[0]		AlwaysOn Power domain GO transition command.
		0	A write of 0 has no effect.
		1	Writing 1 causes the PSC to evaluate all the NEXT fields relevant to this power domain (including the NEXT bit in MDCTL*n* for all the modules residing on this domain). If any of the NEXT fields are not matching the corresponding current state (STATE bit in MDSTAT*n*), the PSC will transition those respective domain/modules to the new NEXT state.

2.7.6 Power Domain Transition Status Register (PTSTAT)

The power domain transition status register (PTSTAT) is shown in Figure 7 and described in Table 12.

Figure 7. Power Domain Transition Status Register (PTSTAT)



LEGEND: R = Read only; -*n* = value after reset

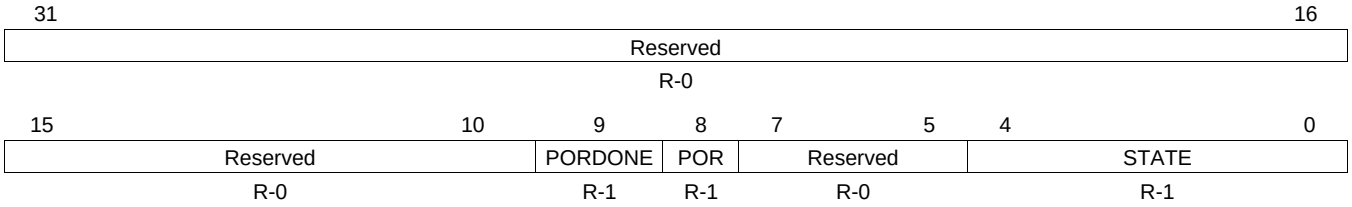
Table 12. Power Domain Transition Status Register (PTSTAT) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved
0	GOSTAT[0]		AlwaysOn Power domain transition status.
		0	No transition is in progress.
		1	Modules in AlwaysOn power domain are transitioning.

2.7.7 Power Domain Status 0 Register (PDSTAT0)

The power domain status n register (PDSTAT0) is shown in Figure 8 and described in Table 13. PDSTAT0 applies to the AlwaysOn power domain.

Figure 8. Power Domain Status 0 Register (PDSTAT0)



LEGEND: R = Read only; -*n* = value after reset

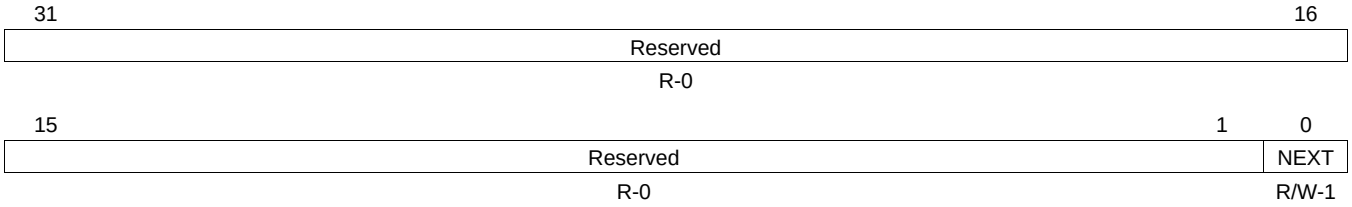
Table 13. Power Domain Status 0 Register (PDSTAT0) Field Descriptions

Bit	Field	Value	Description
31-10	Reserved	0	Reserved
9	PORDONE		Power_On_Reset (POR) done status.
		0	Power domain POR is not done.
		1	Power domain POR is done.
8	POR		Power domain Power_On_Reset (POR) status. This bit reflects the POR status for this power domain including all modules in the domain.
		0	Power domain POR is asserted.
		1	Power domain POR is de-asserted.
7-5	Reserved	0	Reserved
4-0	STATE		Power domain status
		0	Power domain is in the off state.
		1	Power domain is in the on state.

2.7.8 Power Domain Control 0 Register (PDCTL0)

The power domain control n register (PDCTL0) is shown in Figure 9 and described in Table 14. PDCTL0 applies to the AlwaysOn power domain.

Figure 9. Power Domain Control 0 Register (PDCTL0)



LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 14. Power Domain Control 0 Register (PDCTL0) Field Descriptions

Bit	Field	Value	Description
31-1	Reserved	0	Reserved
0	NEXT		Power domain next state.
		0	Power domain off.
		1	Power domain on. AlwaysOn domain must always be programmed to this value.

2.7.9 Module Status n Register (MDSTATn)

The module status *n* register (MDSTAT0-MDSTAT39) is shown in Figure 10 and described in Table 15.

Figure 10. Module Status *n* Register (MDSTAT*n*)

31																		18	17	16
Reserved																			EMUIHB	EMURST
R-0																			R-0	R-0

15	13	12	11	10	9	8	7	6	5	0
Reserved		MCKOUT	Reserved	MRST	LRSTDONE	LRST	Reserved		STATE	
R-0		R-0	R-1	R-0	R-1	R-1	R-0		R-0	

LEGEND: R = Read only; -*n* = value after reset

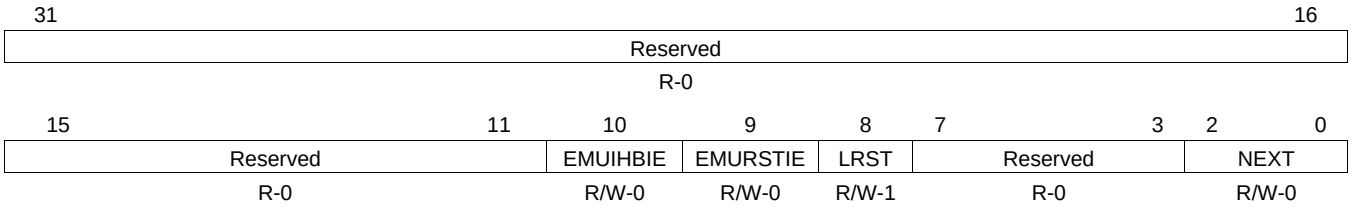
Table 15. Module Status *n* Register (MDSTAT*n*) Field Descriptions

Bit	Field	Value	Description
31-18	Reserved	0	Reserved
17	EMUIHB		Emulation Alters Module State. This bit applies to DSP module only (module 39). This field is 0 for all other modules.
		0	No emulation altering user-desired module state programmed in the NEXT bit in MDCTL39.
		1	Emulation altered user-desired state programmed in the NEXT bit in MDCTL39. If you desire to generate a PSCINT upon this event, you must set the EMUIHBIE bit in MDCTL39..
16	EMURST		Emulation Alters Module Reset. This bit applies to DSP module only (module 39). This field is 0 for all other modules.
		0	No emulation altering user-desired module reset state.
		1	Emulation altered user-desired module reset state. If you desire to generate a PSCINT upon this event, you must set the EMURSTIE bit in MDCTL39.
15-13	Reserved	0	Reserved
12	MCKOUT		Module clock output status. Shows actual status of module clock.
		0	Module clock is off.
		1	Module clock is on.
11	Reserved	1	Reserved
10	MRST		Module reset status. Reflects actual state of module reset.
		0	Module reset is asserted.
		1	Module reset is de-asserted.
9	LRSTDONE		Local reset done. Software is responsible for checking if local reset is done before accessing this module. This bit applies to the DSP module only (module 39). This field is 1 for all other modules.
		0	Local reset is not done.
		1	Local reset is done.
8	LRST		Module local reset status. This bit applies to the DSP module only (module 39).
		0	Local reset is asserted.
		1	Local reset is de-asserted.
7-6	Reserved	0	Reserved
5-0	STATE	0-3Fh	Module state status. Indicates current module status.
		0	SwRstDisable state
		1h	SyncReset state
		2h	Disable state
		3h	Enable state
		4h-3Fh	Indicates transition

2.7.10 Module Control n Register (MDCTLn)

The module control *n* register (MDCTL0-MDCTL39) is shown in Figure 11 and described in Table 16.

Figure 11. Module Control *n* Register (MDCTL*n*)



LEGEND: R/W = Read/Write; R = Read only; -*n* = value after reset

Table 16. Module Control *n* Register (MDCTL*n*) Field Descriptions

Bit	Field	Value	Description
31-11	Reserved	0	Reserved
10	EMUIHBIE		Interrupt enable for emulation alters module state. This bit applies to the DSP module only (module 39). Program this field to 0 for all other modules.
		0	Disable interrupt.
		1	Enable interrupt.
9	EMURSTIE		Interrupt enable for emulation alters reset. This bit applies to the DSP module only (module 39). Program this field to 0 for all other modules.
		0	Disable interrupt.
		1	Enable interrupt.
8	LRST		Module local reset control. This bit applies to the DSP module only (module 39). Program this field to 1 for all other modules.
		0	Assert local reset.
		1	De-assert local reset.
7-3	Reserved	0	Reserved
2-0	NEXT	0-7h	Module next state.
		0	SwRstDisable state
		1h	SyncReset state
		2h	Disable state
		3h	Enable state
		4h-7h	Reserved

3 DSP Reset and Sleep Mode Management

3.1 DSP Reset

Note: The effects of DSP local reset and DSP module reset have not been fully validated; therefore, these resets are not supported and should not be used. Instead, the $\overline{\text{POR}}$ or $\overline{\text{RESET}}$ pins should be used to reset the entire DSP.

With access to the power and sleep controller (PSC) registers, the external host (for example, PCI or HPI) can assert and de-assert DSP local reset and DSP module reset. When DSP local reset is asserted, the DSP's internal memories (L1P, L1D, and L2) are still accessible. Local reset only resets the DSP CPU. Local reset is useful when the DSP module is in the enable or disable states, since module reset is asserted in the SyncReset and SwRstDisable states and module reset supersedes local reset. The intent of DSP module reset is for the external host to completely reset the DSP. The intent of DSP local reset is to allow the external host to hold the CPU in reset while the host is loading code into the DSP internal memory—this step can be useful after the host puts the DSP in module reset and then subsequently enables the DSP. For more information on the PSC, see Section 2. This section describes how to initiate DSP local reset and module reset.

3.1.1 DSP Local Reset

The following steps describe how an external host can assert/de-assert local reset to the DSP:

1. Clear the LRST bit in MDCTL39 to 0 to assert DSP local reset.
2. Set the LRST bit in MDCTL39 to 1 to de-assert DSP local reset. If the DSP is in the enable state, it immediately executes program instructions after reset is de-asserted.

3.1.2 DSP Module Reset

The external host may program the PSC to assert DSP module reset by placing the DSP in either Software Reset Disable (SwRstDisable) state or Synchronous Reset (SyncReset) state. See Section 2 for descriptions of these PSC states.

3.1.2.1 Software Reset Disable (SwRstDisable)

In the software reset disable (SwRstDisable) state, the DSP's module reset is asserted and its module clock is turned off. You can use this state to reset the DSP. The following steps describe how to put the DSP in the software reset disable state:

- Host: Notify the DSP to prepare for power-down.
- DSP: Put the DSP in the IDLE state.
  - Set PDCCMD to 0001 5555h. PDCMD is a control register in the DSP power-down controller module.

    Note: This register can only be written while the DSP is in its supervisor mode.
  - Execute the IDLE instruction if the DSP is in the enable state. IDLE is a program instruction in the C64x+ CPU instruction set. When the CPU executes IDLE, the PDC is notified and will initiate the DSP power-down according to the bits that you set in the PDCCMD (0181 0000h) register. See the *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871) for more information on the PDC and the IDLE instruction.

- Host: Software reset disable DSP.
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. You must wait for the power domain to finish any previously initiated transitions before initiating a new transition.
  - Clear the NEXT bit in MDCTL39 to 0 to prepare the DSP module for a SwRstDisable transition.
  - Set the GO[0] bit in PTCMD to 1 to initiate the state transition.
  - Wait for GOSTAT[0] bit in PTSTAT to clear to 0. The module is safely in the new state only after the GOSTAT[0] bit is cleared to 0.
- Host: Assert the DSP local reset (Optional)
  - Clear the LRST bit in MDCTL39 to 0. This step is optional. This step asserts the DSP local reset, and is included here so that the DSP does not start running immediately upon it is subsequently enable by the host. Typically, the host only de-asserts local reset to the DSP after it makes sure that code is properly loaded.

3.1.2.2 Synchronous Reset (SyncReset)

In the synchronous reset (SyncReset) state, the DSP's module reset is asserted and its module clock is enabled. You can use this state to reset the DSP. The following steps describe how to put the DSP in the synchronous reset state:

- Host: Notify the DSP to prepare for power-down.
- DSP: Put the DSP in the IDLE state.
  - Set PDCCMD to 0001 5555h. PDCMD is a control register in the DSP power-down controller module.

    Note: This register can only be written while the DSP is in supervisor mode.
  - Execute the IDLE instruction.
- Host: Sync reset DSP
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. You must wait for the power domain to finish any previously initiated transitions before initiating a new transition.
  - Set the NEXT bit in MDCTL39 to 1 to prepare the DSP module for a SyncReset transition.
  - Set the GO[0] bit in PTCMD to 1 to initiate the state transition.
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. The module is safely in the new state only after the GOSTAT[0] bit is cleared to 0.
- Host: Assert DSP local reset (Optional)
  - Clear the LRST bit in MDCTL39 to 0. This step is optional. This step asserts the DSP local reset and is included here so that the DSP does not start running immediately upon it is subsequently enabled by the host. Typically, software de-asserts local reset to the DSP after it makes sure that code is properly loaded.

3.2 DSP Sleep Mode Management

The C64x+ DSP supports sleep mode management to reduce power:

- DSP clock can be completely shut off
- C64x+ Megamodule can be put in sleep mode
  - C64x+ CPU can be put in sleep mode

On the C642x DSP, sleep mode for the DSP internal memories (L1P, L1D, L2) is not supported.

3.2.1 DSP Sleep Modes

The C64x+ Megamodule includes a power-down controller (PDC) that controls the power-down of the C64x+ Megamodule components. The PDC can power-down all of the following components of the C64x+ Megamodule:

- C64x+ CPU
- L1P controller
- L1D controller
- L2 controller
- Extended memory controller (EMC)
- Internal direct memory access (IDMA) controller

The C64x+ Megamodule is capable of providing both dynamic and static power-down; however, only static power-down is supported on the C642x DSP. The *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871) describes the power-down control in more detail.

- Static power-down: The PDC initiates power down of the entire C64x+ Megamodule and all internal memories immediately upon command from software.

On the C642x DSP, static power-down affects all components of the C64x+ Megamodule. The C642x DSP does not support power-down of the internal memories (L1P, L1D, L2 memory). Software can initiate static power-down via a register bit in the PDC register. For more information on the PDC, see the *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871).

3.2.2 DSP Module Clock ON/OFF

As discussed in Section 3.2.1, the C64x+ Megamodule can clock gate its own components to save power. Additional power saving can be achieved by stopping the clock source to the C64x+ Megamodule by programming the power and sleep controller (PSC) to place the C64x+ Megamodule in Disable state. The C64x+ DSP cannot perform this programming task on its own, because the C64x+ DSP will not be able to complete the PSC programming sequence if the C64x+ DSP clock source is gated in the middle of the process. If stopping the clock source to the C64x+ DSP is desired for additional power saving, an external host is responsible for programming the PSC (for example, via HPI, PCI interfaces) to disable the C64x+ Megamodule. Similarly, in that case the external host is responsible for programming the PSC to enable the C64x+ Megamodule.

3.2.2.1 DSP Module Clock ON

In the clock Enable state, the DSP's module clock is enabled while DSP module reset is de-asserted. This is the state for normal DSP run-time. DSP defaults to Enable state, therefore this DSP Module Clock ON process is typically not needed. This process is only required to wake up the DSP after an external host puts the DSP in Disable state (Section 3.2.2.2).

- Host: Enable clocks to the DSP.
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. You must wait for the power domain to finish any previously initiated transitions before initiating a new transition.
  - Set the NEXT bit in MDCTL39 to 3h to prepare the DSP module for an enable transition.
  - Set the GO[0] bit in PTCMD to 1 to initiate the state transition.
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. The domain is only safely in the new state after the GOSTAT[0] bit is cleared to 0.
  - Wait for the STATE bit in MDSTAT39 to change to 3h. The module is only safely in the new state after the STATE bit in MDSTAT39 changes to reflect the new state.
- Host: Wake the DSP.
  - If transitioning from the disable state, trigger a DSP interrupt that has previously been configured as a wake-up interrupt.

Note: This step only applies if you are transitioning from the disable state. If previously in the disable state, a wake-up interrupt must be triggered in order to wake the DSP. This example assumes that the DSP enabled this interrupt before entering its IDLE state. If previously in the software reset disable or synchronous reset state, it is not necessary to wake the DSP because these states assert the DSP module reset. See Section 3.1 for information on the software reset disable and synchronous reset states. See the *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871) for more information on DSP interrupts.

3.2.2.2 DSP Module Clock Off

In the clock Disable state, the DSP's module clock is disabled, while DSP reset remains de-asserted. This state is typically used to disable the DSP clock to save power. As mentioned in Section 3.2.2, the DSP cannot put itself in Disable state. An external host is responsible for performing this task. For example, it can be an external host interfacing through the HPI or PCI peripheral.

- Host: Notify the DSP to prepare for power-down.
- DSP: Drain all existing operations and ensure there are no accesses to the C64x+ megamodule prior to DSP power-down.
  - Program the PSC to disable all master peripherals (except the Host) that are capable of initiating transfers to the C64x+ Megamodule.
  - Check EDMA transfer status to ensure there is no outstanding EDMA transfers that can access the C64x+ Megamodule.
- DSP: Prepare for power-down.
  - Set PDCCMD to 0001 5555h. PDCCMD is a control register in the DSP power-down controller module.

    Note: This register can only be written while the DSP is in supervisor mode.
  - Enable one of the interrupts that the host would like to use to wake the DSP in the DSP clock-on sequence.
  - Execute the IDLE instruction. IDLE is a program instruction in the C64x+ CPU instruction set. When the CPU executes IDLE, the PDC is notified and initiates DSP power-down according to the bits that you set in the PDCCMD (0181 0000h) register. See the *TMS320C64x+ DSP Megamodule Reference Guide* (SPRU871) for more information on the PDC and the IDLE instruction.
- Host: Disable the DSP clock.
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. You must wait for the power domain to finish any previously initiated transitions before initiating a new transition.
  - Set the NEXT bit in MDCTL39 to 2h to prepare the DSP module for a disable transition.
  - Set the GO[0] bit in PTCMD to 1 to initiate the state transition.
  - Wait for the GOSTAT[0] bit in PTSTAT to clear to 0. The domain is only safely in the new state after the GOSTAT[0] bit is cleared to 0.
  - Wait for the STATE bit in MDSTAT39 to change to 2h. The module is only safely in the new state after the STATE bit in MDSTAT39 changes to reflect the new state.

Appendix A Revision History

Table A-1 lists the changes made since the previous version of this document.

Table A-1. Document Revision History

Reference	Additions/Modifications/Deletions
Figure 1	Added Note.
Section 2.3	Added Note.
Section 3.1	Added Note.

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 TI 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. Information of third parties may be subject to additional restrictions.

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.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

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
Clocks and Timers	www.ti.com/clocks	Digital Control	www.ti.com/digitalcontrol
Interface	interface.ti.com	Medical	www.ti.com/medical
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
RFID	www.ti-rfid.com	Telephony	www.ti.com/telephony
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

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