- **Powerful 16-Bit TMS320C5x CPU**
- **20-, 25-, 35-, and 50-ns Single-Cycle Instruction Execution Time for 5-V Operation**
- **25-, 40-, and 50-ns Single-Cycle Instruction Execution Time for 3-V Operation**
- **Single-Cycle 16 × 16-Bit Multiply/Add**
- **224K × 16-Bit Maximum Addressable External Memory Space (64K Program, 64K Data, 64K I/O, and 32K Global)**
- **2K, 4K, 8K, 16K, 32K × 16-Bit Single-Access On-Chip Program ROM**
- **1K, 3K, 6K, 9K × 16-Bit Single-Access On-Chip Program/Data RAM (SARAM)**
- **1K Dual-Access On-Chip Program/Data RAM (DARAM)**
- **Full-Duplex Synchronous Serial Port for Coder/Decoder Interface**
- **Time-Division-Multiplexed (TDM) Serial Port**
- **Hardware or Software Wait-State Generation Capability**
- **On-Chip Timer for Control Operations**
- **Repeat Instructions for Efficient Use of Program Space**
- **Buffered Serial Port**
- **Host Port Interface**
- **Multiple Phase-Locked Loop (PLL) Clocking Options (×1, ×2, ×3, ×4, ×5, ×9 Depending on Device)**
- **Block Moves for Data/Program Management**
- **On-Chip Scan-Based Emulation Logic**
- **Boundary Scan**
- **Five Packaging Options**
  - **100-Pin Quad Flat Package (PJ Suffix)**
  - **100-Pin Thin Quad Flat Package (PZ Suffix)**
  - **128-Pin Thin Quad Flat Package (PBK Suffix)**
  - **132-Pin Quad Flat Package (PQ Suffix)**
  - **144-Pin Thin Quad Flat Package (PGE Suffix)**
- **Low Power Dissipation and Power-Down Modes:**
  - **47 mA (2.35 mA/MIP) at 5 V, 40-MHz Clock (Average)**
  - **23 mA (1.15 mA/MIP) at 3 V, 40-MHz Clock (Average)**
  - **10 mA at 5 V, 40-MHz Clock (IDLE1 Mode)**
  - **3 mA at 5 V, 40-MHz Clock (IDLE2 Mode)**
  - **5 μA at 5 V, Clocks Off (IDLE2 Mode)**
- **High-Performance Static CMOS Technology**
- **IEEE Standard 1149.1† Test-Access Port (JTAG)**

description

The TMS320C5x generation of the Texas Instruments (TI™) TMS320 digital signal processors (DSPs) is fabricated with static CMOS integrated circuit technology; the architectural design is based upon that of an earlier TI DSP, the TMS320C25. The combination of advanced Harvard architecture, on-chip peripherals, on-chip memory, and a highly specialized instruction set is the basis of the operational flexibility and speed of the ’C5x‡ devices. They execute up to 50 million instructions per second (MIPS).

The ’C5x devices offer these advantages:

- Enhanced TMS320 architectural design for increased performance and versatility
- Modular architectural design for fast development of spin-off devices
- Advanced integrated-circuit processing technology for increased performance
- Upward-compatible source code (source code for ’C1x and ’C2x DSPs is upward compatible with ’C5x DSPs.)
- Enhanced TMS320 instruction set for faster algorithms and for optimized high-level language operation
- New static-design techniques for minimizing power consumption and maximizing radiation tolerance



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

TI is a trademark of Texas Instruments Incorporated.
† IEEE Standard 1149.1–1990, IEEE Standard Test-Access Port and Boundary-Scan Architecture
‡ References to ’C5x in this document include both TMS320C5x and TMS320LC5x devices unless specified otherwise.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

Copyright © 1996, Texas Instruments Incorporated

description (continued)

Table 1 provides a comparison of the devices in the ’C5x generation. It shows the capacity of on-chip RAM and ROM memories, number of serial and parallel I/O ports, execution time of one machine cycle, and type of package with total pin count.

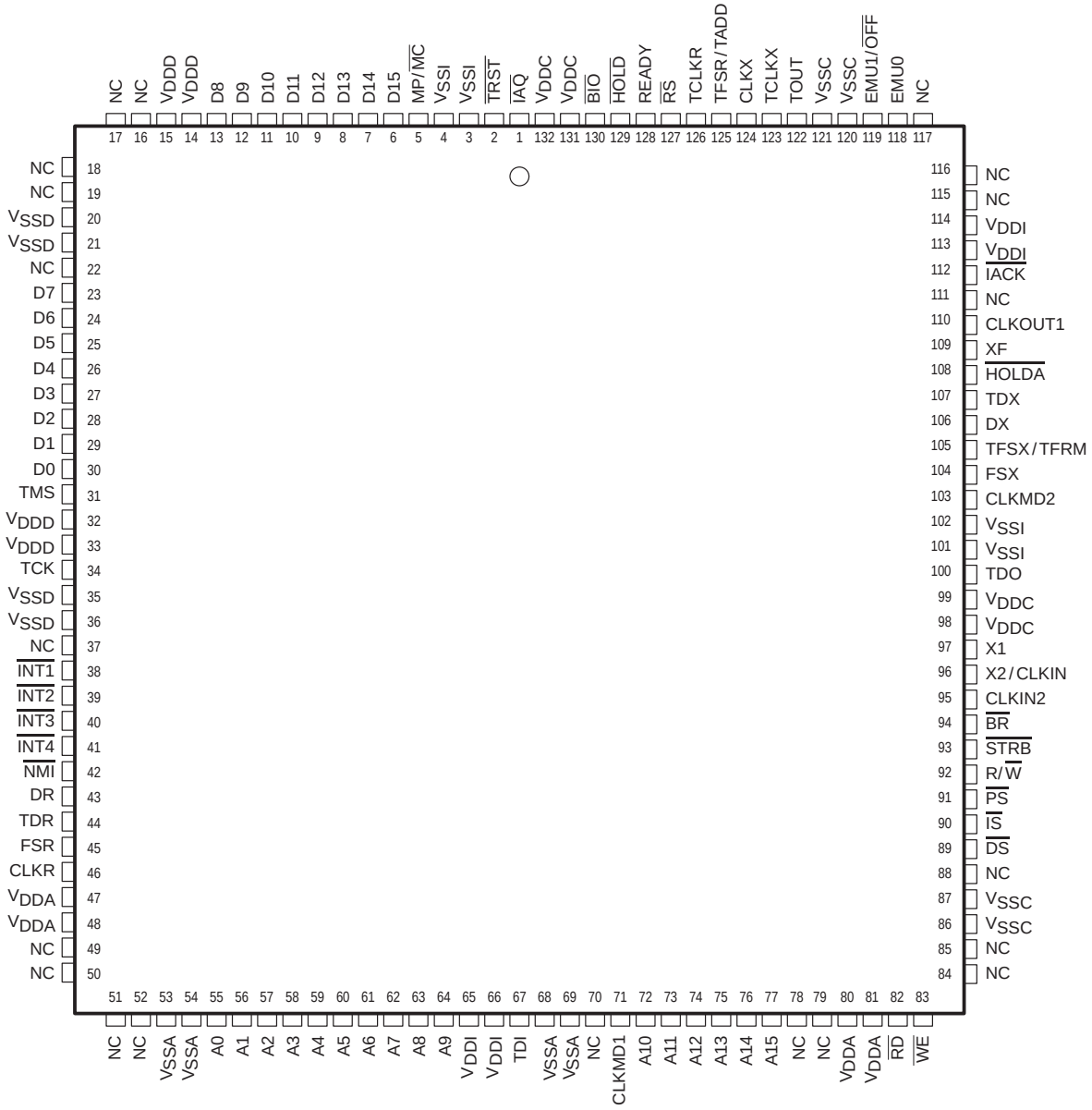
Table 1. Characteristics of the ’C5x Processors

TMS320 DEVICES	ON-CHIP MEMORY (16-BIT WORDS)				I/O PORTS		POWER SUPPLY (V)	CYCLE TIME (ns)	PACKAGE TYPE QFP‡
	DARAM		SARAM	ROM					
	DATA	DATA + PROG	DATA + PROG	PROG	SERIAL	PARALLEL†			
TMS320C50	544	512	9K	2K§	2	64K	5	50/35/25	132 pin
TMS320LC50	544	512	9K	2K§	2	64K	3.3	50/40/25	132 pin
TMS320C51	544	512	1K	8K§	2	64K	5	50/35/25/20	100/132 pin
TMS320LC51	544	512	1K	8K§	2	64K	3.3	50/40/25	100/132 pin
TMS320C52	544	512	–	4K§	1¶	64K	5	50/35/25/20	100 pin
TMS320LC52	544	512	–	4K§	1¶	64K	3.3	50/40/25	100 pin
TMS320C53	544	512	3K	16K§	2	64K	5	50/35/25	132 pin
TMS320LC53	544	512	3K	16K§	2	64K	3.3	50/40/25	132 pin
TMS320C53S	544	512	3K	16K§	2¶	64K	5	50/35/25	100 pin
TMS320LC53S	544	512	3K	16K§	2¶	64K	3.3	50/40/25	100 pin
TMS320LC56	544	512	6K	32K	2#	64K	3.3	35/25	100 pin
TMS320LC57	544	512	6K	32K	2#	64K + HPI‖	3.3	35/25	128 pin
TMS320C57S	544	512	6K	2K§	2#	64K + HPI‖	5	50/35/25	144 pin
TMS320LC57S	544	512	6K	2K§	2#	64K + HPI‖	3.3	50/35	144 pin

† Sixteen of the 64K parallel I/O ports are memory mapped.
‡ QFP = Quad flatpack
§ ROM boot loader available
¶ TDM serial port not available
# Includes auto-buffered serial port (BSP) but TDM serial port not available
‖ HPI = Host port interface

Pinouts for each package are device-specific.

TMS320C50, TMS320LC50, TMS320C51, TMS320LC51, TMS320C53, TMS320LC53
PQ PACKAGE
(TOP VIEW)



NOTE: NC = No connect (These pins are reserved.)

Pin Functions for Devices in the PQ Package

SIGNAL	TYPE	DESCRIPTION
PARALLEL INTERFACE BUS		
A0–A15	I/O/Z	16-bit external address bus (MSB: A15, LSB: A0)
D0–D15	I/O/Z	16-bit external data bus (MSB: D15, LSB: D0)
$\overline{PS}$, $\overline{DS}$, $\overline{IS}$	O/Z	Program, data, and I/O space select outputs, respectively
$\overline{STRB}$	I/O/Z	Timing strobe for external cycles and external DMA
R/$\overline{W}$	I/O/Z	Read/write select for external cycles and external DMA
$\overline{RD}$, $\overline{WE}$	O/Z	Read and write strobes, respectively, for external cycles
READY	I	External bus ready/wait-state control input
$\overline{BR}$	I/O/Z	Bus request. Arbitrates global memory and external DMA
SYSTEM INTERFACE/CONTROL SIGNALS		
$\overline{RS}$	I	Reset. Initializes device and sets PC to zero
MP/$\overline{MC}$	I	Microprocessor/microcomputer mode select. Enables internal ROM
$\overline{HOLD}$	I	Puts parallel I/F bus in high-impedance state after current cycle
$\overline{HOLDA}$	O/Z	Hold acknowledge. Indicates external bus in hold state
XF	O/Z	External flag output. Set/cleared through software
$\overline{BIO}$	I	I/O branch input. Implements conditional branches
TOUT	O/Z	Timer output signal. Indicates output of internal timer
$\overline{IAQ}$	O/Z	Instruction acquisition signal
$\overline{IACK}$	O/Z	Interrupt acknowledge signal
$\overline{INT1}$–$\overline{INT4}$	I	External interrupt inputs
$\overline{NMI}$	I	Nonmaskable external interrupt
SERIAL PORT INTERFACE (SPI)		
DR	I	Serial receive-data input
DX	O/Z	Serial transmit-data output. In high-impedance state when not transmitting
CLKR	I	Serial receive-data clock input
CLKX	I/O/Z	Serial transmit-data clock. Internal or external source
FSR	I	Serial receive-frame-synchronization input
FSX	I/O/Z	Serial transmit-frame-synchronization signal. Internal or external source
TDM SERIAL-PORT INTERFACE		
TDR	I	TDM serial receive-data input
TDX	O/Z	TDM serial transmit-data output. In high-impedance state when not transmitting
TCLKR	I	TDM serial receive-data clock input
TCLKX	I/O/Z	TDM serial transmit-data clock. Internal or external source
TFSR / TADD	I/O/Z	TDM serial receive-frame-synchronization input. In the TDM mode, TFSR/TADD is used to output/input the address of the port.
TFSX/TFRM	I	TDM serial transmit-frame-synchronization signal. Internal or external source. In the TDM mode, TFSX/TFRM becomes TFRM, the TDM frame synchronization.

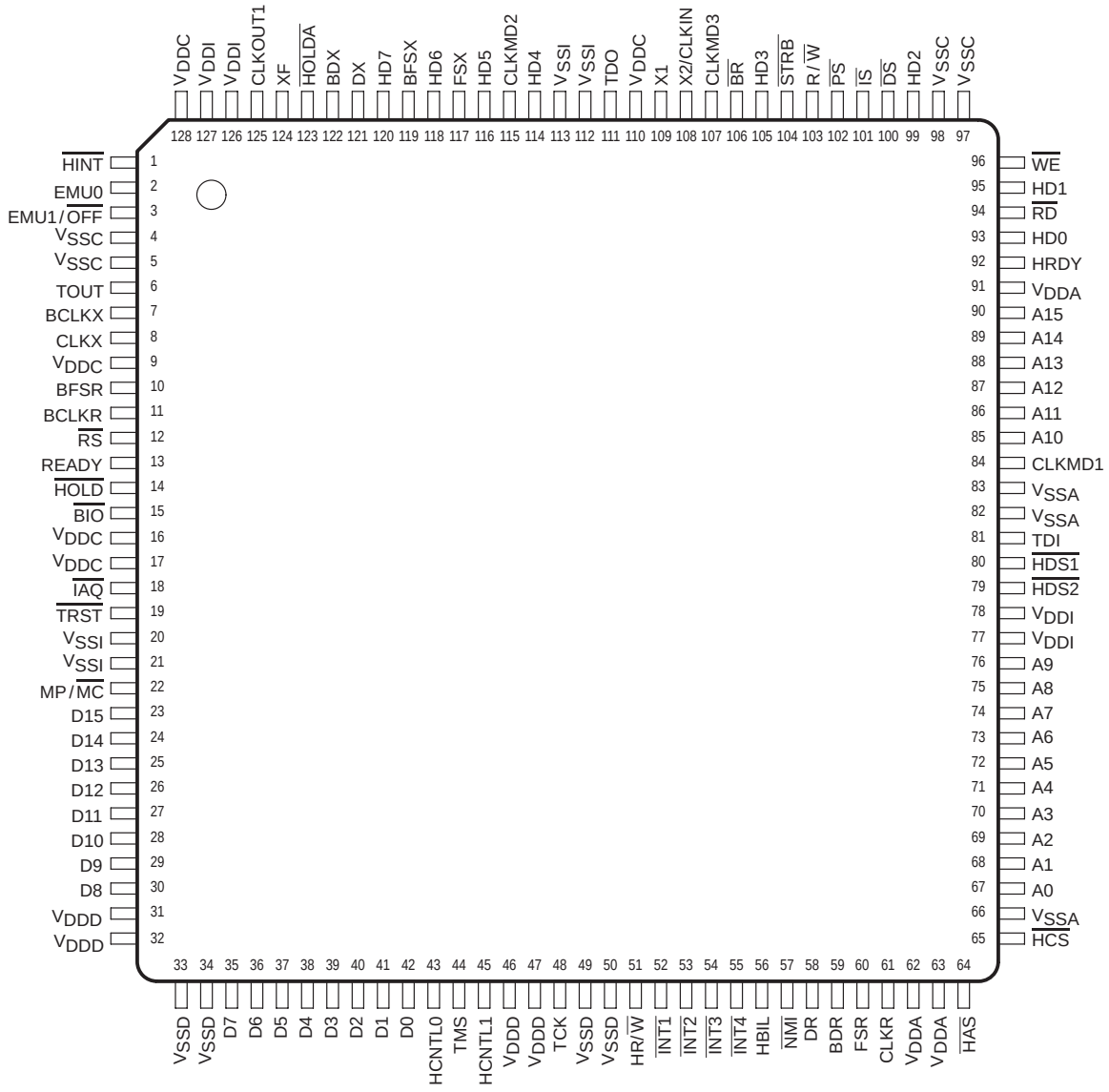
LEGEND:
I = Input
O = Output
Z = High impedance

Pin Functions for Devices in the PQ Package (Continued)

EMULATION/IEEE STANDARD 1149.1 TEST ACCESS PORT (TAP)		
TDI	I	TAP scan data input
TDO	O/Z	TAP scan data output
TMS	I	TAP mode select input
TCK	I	TAP clock input
$\overline{\text{TRST}}$	I	TAP reset (with pulldown resistor). Disables TAP when low
EMU0	I/O/Z	Emulation control 0. Reserved for emulation use
EMU1/$\overline{\text{OFF}}$	I/O/Z	Emulation control 1. Puts outputs in high-impedance state when low
CLOCK GENERATION AND CONTROL		
X1	O	Oscillator output
X2/CLKIN	I	Clock/oscillator input
CLKIN2	I	Clock input
CLKMD1, CLKMD2	I	Clock-mode select inputs
CLKOUT1	O/Z	Device system-clock output
POWER SUPPLY CONNECTIONS		
V_{DDA}	S	Supply connection, address-bus output
V_{DDD}	S	Supply connection, data-bus output
V_{DDC}	S	Supply connection, control output
V_{DDI}	S	Supply connection, internal logic
V_{SSA}	S	Supply connection, address-bus output
V_{SSD}	S	Supply connection, data-bus output
V_{SSC}	S	Supply connection, control output
V_{SSI}	S	Supply connection, internal logic

LEGEND:
I = Input
O = Output
S = Supply
Z = High impedance

TMS320LC57
PBK PACKAGE
(TOP VIEW)



Pin Functions for the TMS320LC57 in the PBK Package

SIGNAL	TYPE	DESCRIPTION
PARALLEL INTERFACE BUS		
A0–A15	I/O/Z	16-bit external address bus (MSB: A15, LSB: A0)
D0–D15	I/O/Z	16-bit external data bus (MSB: D15, LSB: D0)
$\overline{PS}$, $\overline{DS}$, $\overline{IS}$	O/Z	Program, data, and I/O space select outputs, respectively
$\overline{STRB}$	I/O/Z	Timing strobe for external cycles and external DMA
R/$\overline{W}$	I/O/Z	Read/write select for external cycles and external DMA
$\overline{RD}$, $\overline{WE}$	O/Z	Read and write strobes, respectively, for external cycles
READY	I	External bus ready/wait-state control input
$\overline{BR}$	I/O/Z	Bus request. Arbitrates global memory and external DMA
SYSTEM INTERFACE/CONTROL SIGNALS		
$\overline{RS}$	I	Reset. Initializes device and sets PC to zero
MP/$\overline{MC}$	I	Microprocessor/microcomputer mode select. Enables internal ROM
$\overline{HOLD}$	I	Puts parallel I/F bus in high-impedance state after current cycle
$\overline{HOLDA}$	O/Z	Hold acknowledge. Indicates external bus in hold state
XF	O/Z	External flag output. Set/cleared through software
$\overline{BIO}$	I	I/O branch input. Implements conditional branches
TOUT	O/Z	Timer output signal. Indicates output of internal timer
$\overline{IAQ}$	O/Z	Instruction acquisition signal
$\overline{INT1}$–$\overline{INT4}$	I	External interrupt inputs
$\overline{NMI}$	I	Nonmaskable external interrupt
SERIAL PORT INTERFACE		
DR	I	Serial receive-data input
DX	O/Z	Serial transmit-data output. In high-impedance state when not transmitting
CLKR	I	Serial receive-data clock input
CLKX	I/O/Z	Serial transmit-data clock. Internal or external source
FSR	I	Serial receive-frame-synchronization input
FSX	I/O/Z	Serial transmit-frame-synchronization signal. Internal or external source
HOST PORT INTERFACE (HPI)		
HCNTL0	I	HPI mode control 1
HCNTL1	I	HPI mode control 2
$\overline{HINT}$	O/Z	Host interrupt
$\overline{HDS1}$	I	HPI data strobe 1
$\overline{HDS2}$	I	HPI data strobe 2
HR/$\overline{W}$	I	HPI read/write strobe
$\overline{HAS}$	I	HPI address strobe
HRDY	O/Z	HPI ready signal
$\overline{HCS}$	I	HPI chip select
HBIL	I	HPI byte identification input
HD0–HD7	I/O/Z	HPI data bus

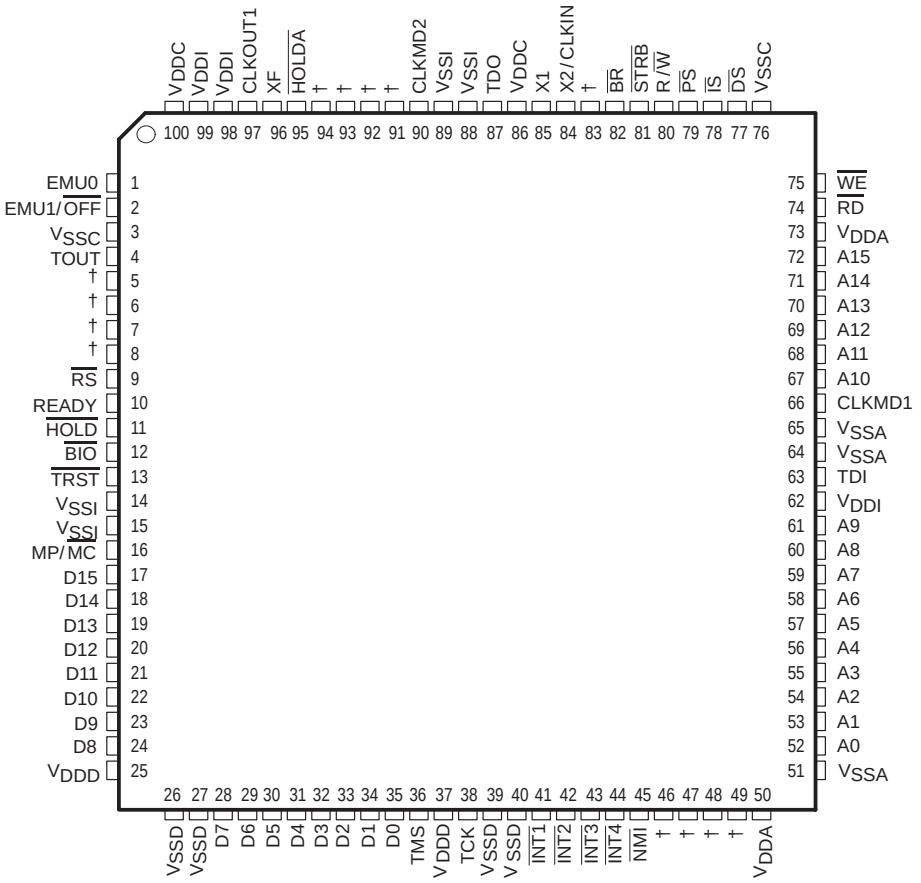
LEGEND:
I = Input
O = Output
Z = High impedance

Pin Functions for the TMS320LC57 in the PBK Package (Continued)

SIGNAL	TYPE	DESCRIPTION
BUFFERED SERIAL PORT		
BDR	I	BSP receive data input
BDX	O/Z	BSP transmit data output; in high-impedance state when not transmitting
BCLKR	I	BSP receive-data clock input
BCLKX	I/O/Z	BSP transmit-data clock; internal or external source
BFSR	I	BSP receive frame-synchronization input
BFSX	I/O/Z	BSP transmit frame-synchronization signal; internal or external source
EMULATION/JTAG INTERFACE		
TDI	I	JTAG-test-port scan data input
TDO	O/Z	JTAG-test-port scan data output
TMS	I	JTAG-test-port mode select input
TCK	I	JTAG-port clock input
$\overline{\text{TRST}}$	I	JTAG-port reset (with pull-down resistor). Disables JTAG when low
EMU0	I/O/Z	Emulation control 0. Reserved for emulation use
EMU1/$\overline{\text{OFF}}$	I/O/Z	Emulation control 1. Puts outputs in high-impedance state when low
CLOCK GENERATION AND CONTROL		
X1	O	Oscillator output
X2/CLKIN	I	Clock input
CLKMD1, CLKMD2, CLKMD3	I	Clock-mode select inputs
CLKOUT1	O/Z	Device system-clock output
POWER SUPPLY CONNECTIONS		
V_{DDA}	S	Supply connection, address-bus output
V_{DDD}	S	Supply connection, data-bus output
V_{DDC}	S	Supply connection, control output
V_{DDI}	S	Supply connection, internal logic
V_{SSA}	S	Supply connection, address-bus output
V_{SSD}	S	Supply connection, data-bus output
V_{SSC}	S	Supply connection, control output
V_{SSI}	S	Supply connection, internal logic

LEGEND:
I = Input
O = Output
S = Supply
Z = High impedance

TMS320C51, TMS320LC51, TMS320C52, TMS320LC52, TMS320C53S, TMS320LC53S, TMS320LC56
PZ PACKAGE
(TOP VIEW)



NOTE: NC = No connect (These pins are reserved.)
† See Table 2 for device-specific pinouts.

Table 2. Device-Specific Pinouts for the PZ Package

PIN	'C51, 'LC51	'C52, 'LC52	'C53S, 'LC53S	'LC56‡
5	TCLKX	V_{SSI}	CLKX2	BCLKX
6§	CLKX	CLKX	CLKX1	CLKX
7	TFSR/TADD	V_{SSI}	FSR2	BFSR
8	TCLKR	V_{SSI}	CLKR2	BCLKR
46§	DR	DR	DR1	DR
47	TDR	V_{SSI}	DR2	BDR
48§	FSR	FSR	FSR1	FSR
49§	CLKR	CLKR	CLKR1	CLKR
83	CLKIN2	CLKIN2	CLKIN2	CLKMD3
91§	FSX	FSX	FSX1	FSX
92	TFSX/TFRM	V_{SSI}	FSX2	BFSX
93§	DX	DX	DX1	DX
94	TDX	NC	DX2	BDX

‡ Pin names beginning with "B" indicate signals on the buffered serial port (BSP).
§ No functional change

Pin Functions for Devices in the PZ Package

SIGNAL	TYPE	DESCRIPTION
PARALLEL INTERFACE BUS		
A0–A15	I/O/Z	16-bit external address bus (MSB: A15, LSB: A0)
D0–D15	I/O/Z	16-bit external data bus (MSB: D15, LSB: D0)
$\overline{PS}$, $\overline{DS}$, $\overline{IS}$	O/Z	Program, data, and I/O space select outputs, respectively
$\overline{STRB}$	I/O/Z	Timing strobe for external cycles and external DMA
R/$\overline{W}$	I/O/Z	Read/write select for external cycles and external DMA
$\overline{RD}$, $\overline{WE}$	O/Z	Read and write strobes, respectively, for external cycles
READY	I	External bus ready/wait-state control input
$\overline{BR}$	I/O/Z	Bus request. Arbitrates global memory and external DMA
SYSTEM INTERFACE/CONTROL SIGNALS		
$\overline{RS}$	I	Reset. Initializes device and sets PC to zero
MP/$\overline{MC}$	I	Microprocessor/microcomputer mode select. Enables internal ROM
$\overline{HOLD}$	I	Puts parallel I/F bus in high-impedance state after current cycle
$\overline{HOLDA}$	O/Z	Hold acknowledge. Indicates external bus in hold state
XF	O/Z	External flag output. Set/cleared through software
$\overline{BIO}$	I	I/O branch input. Implements conditional branches
TOUT	O/Z	Timer output signal. Indicates output of internal timer
$\overline{INT1}$–$\overline{INT4}$	I	External interrupt inputs
$\overline{NMI}$	I	Nonmaskable external interrupt
SERIAL PORT INTERFACE		
DR, DR1, DR2	I	Serial receive-data input
DX, DX1, DX2	O/Z	Serial transmit-data output. In high-impedance state when not transmitting
CLKR, CLKR1, CLKR2	I	Serial receive-data clock input
CLKX, CLKX1, CLKX2	I/O/Z	Serial transmit-data clock. Internal or external source
FSR, FSR1, FSR2	I	Serial receive-frame-synchronization input
FSX, FSX1, FSX2	I/O/Z	Serial transmit-frame-synchronization signal. Internal or external source
BUFFERED SERIAL PORT (BSP) (SEE NOTE 1)		
BDR	I	BSP receive data input
BDX	O/Z	BSP transmit data output; in high-impedance state when not transmitting
BCLKR	I	BSP receive-data clock input
BCLKX	I/O/Z	BSP transmit-data clock; internal or external source
BFSR	I	BSP receive frame-synchronization input
BFSX	I/O/Z	BSP transmit frame-synchronization signal; internal or external source

LEGEND:
I = Input
O = Output
Z = High impedance
NOTE 1: 'LC56 devices only

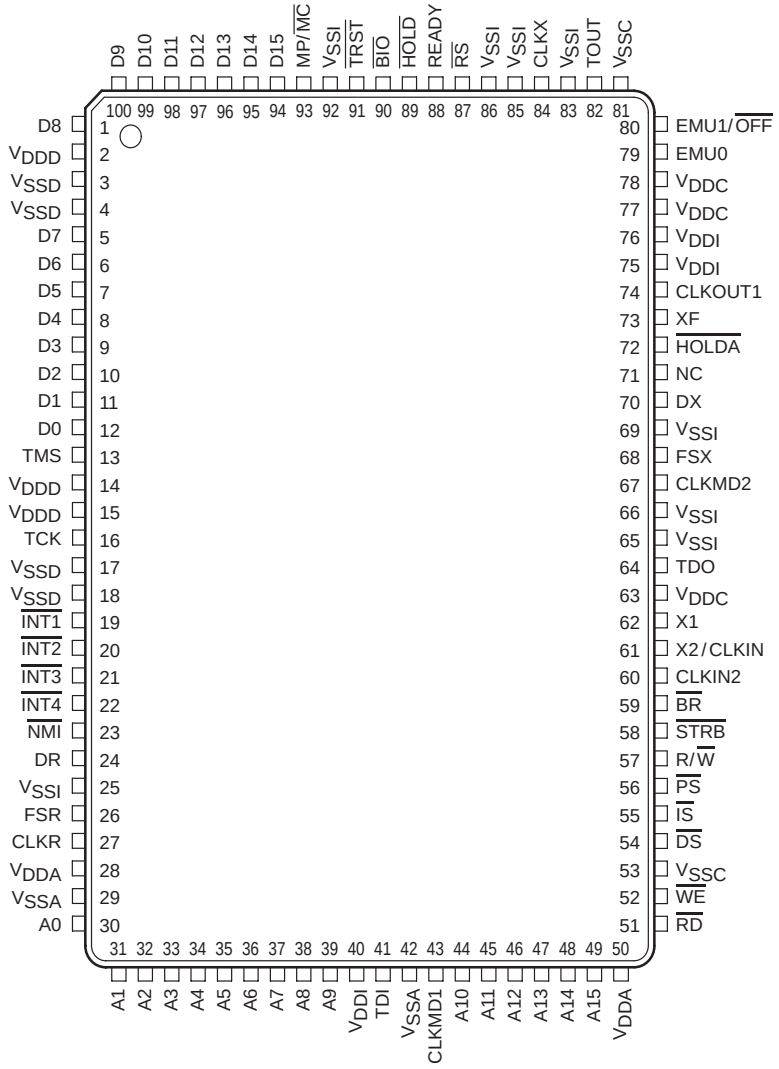
Pin Functions for Devices in the PZ Package (Continued)

SIGNAL	TYPE	DESCRIPTION
TDM SERIAL PORT INTERFACE		
TDR	I	TDM serial receive-data input
TDX	O/Z	TDM serial transmit-data output. In high-impedance state when not transmitting
TCLKR	I	TDM serial receive-data clock input
TCLKX	I/O/Z	TDM serial transmit-data clock. Internal or external source
TFSR / TADD	I/O/Z	TDM serial receive-frame-synchronization input. In the TDM mode, TFSR/TADD is used to output/input the address of the port
TFSX / TFRM	I	TDM serial transmit-frame-synchronization signal. Internal or external source. In the TDM mode, TFSX/TFRM becomes TFRM, the TDM frame sync.
EMULATION/JTAG INTERFACE		
TDI	I	JTAG-test-port scan data input
TDO	O/Z	JTAG-test-port scan data output
TMS	I	JTAG-test-port mode select input
TCK	I	JTAG-port clock input
$\overline{\text{TRST}}$	I	JTAG-port reset (with pull-down resistor). Disables JTAG when low
EMU0	I/O/Z	Emulation control 0. Reserved for emulation use
EMU1/$\overline{\text{OFF}}$	I/O/Z	Emulation control 1. Puts outputs in high-impedance state when low
CLOCK GENERATION AND CONTROL (SEE NOTE 2)		
X1	O	Oscillator output
X2/CLKIN	I	Clock/oscillator input (PLL clock input for ’C56)
CLKIN2	I	Clock input (PLL clock input for ’C50, ’C51, ’C52, ’C53, ’C53S)
CLKMD1, CLKMD2, CLKMD3	I	Clock-mode select inputs
CLKOUT1	O/Z	Device system-clock output
POWER SUPPLY CONNECTIONS		
V_{DDA}	S	Supply connection, address-bus output
V_{DDD}	S	Supply connection, data-bus output
V_{DDC}	S	Supply connection, control output
V_{DDI}	S	Supply connection, internal logic
V_{SSA}	S	Supply connection, address-bus output
V_{SSD}	S	Supply connection, data-bus output
V_{SSC}	S	Supply connection, control output
V_{SSI}	S	Supply connection, internal logic

LEGEND:
I = Input
O = Output
S = Supply
Z = High impedance

NOTE 2: CLKIN2 pin is replaced by CLKMD3 pin on ’LC56 devices.

TMS320C52, TMS320LC52
PJ PACKAGE
(TOP VIEW)



NOTE: NC = No connect (These pins are reserved.)

Pin Functions for the TMS320C52, TMS320LC52 in the PJ Package

SIGNAL	TYPE	DESCRIPTION
PARALLEL INTERFACE BUS		
A0–A15	I/O/Z	16-bit external address bus (MSB: A15, LSB: A0)
D0–D15	I/O/Z	16-bit external data bus (MSB: D15, LSB: D0)
$\overline{PS}$, $\overline{DS}$, $\overline{IS}$	O/Z	Program, data, and I/O space select outputs, respectively
$\overline{STRB}$	I/O/Z	Timing strobe for external cycles and external DMA
R/$\overline{W}$	I/O/Z	Read/write select for external cycles and external DMA
$\overline{RD}$, $\overline{WE}$	O/Z	Read and write strobes, respectively, for external cycles
READY	I	External bus ready/wait-state control input
$\overline{BR}$	I/O/Z	Bus request. Arbitrates global memory and external DMA
SYSTEM INTERFACE/CONTROL SIGNALS		
$\overline{RS}$	I	Reset. Initializes device and sets PC to zero
MP/$\overline{MC}$	I	Microprocessor/microcomputer mode select. Enables internal ROM
$\overline{HOLD}$	I	Puts parallel I/F bus in high-impedance state after current cycle
$\overline{HOLDA}$	O/Z	Hold acknowledge. Indicates external bus in hold state
XF	O/Z	External flag output. Set/cleared through software
$\overline{BIO}$	I	I/O branch input. Implements conditional branches
TOUT	O/Z	Timer output signal. Indicates output of internal timer
$\overline{INT1}$–$\overline{INT4}$	I	External interrupt inputs
$\overline{NMI}$	I	Nonmaskable external interrupt
SERIAL PORT INTERFACE		
DR	I	Serial receive-data input
DX	O/Z	Serial transmit-data output. In high-impedance state when not transmitting
CLKR	I	Serial receive-data clock input
CLKX	I/O/Z	Serial transmit-data clock. Internal or external source
FSR	I	Serial receive-frame-synchronization input
FSX	I/O/Z	Serial transmit-frame-synchronization signal. Internal or external source
EMULATION/JTAG INTERFACE		
TDI	I	JTAG-test-port scan data input
TDO	O/Z	JTAG-test-port scan data output
TMS	I	JTAG-test-port mode select input
TCK	I	JTAG-port clock input
$\overline{TRST}$	I	JTAG-port reset (with pulldown resistor). Disables JTAG when low
EMU0	I/O/Z	Emulation control 0. Reserved for emulation use
EMU1/$\overline{OFF}$	I/O/Z	Emulation control 1. Puts outputs in high-impedance state when low

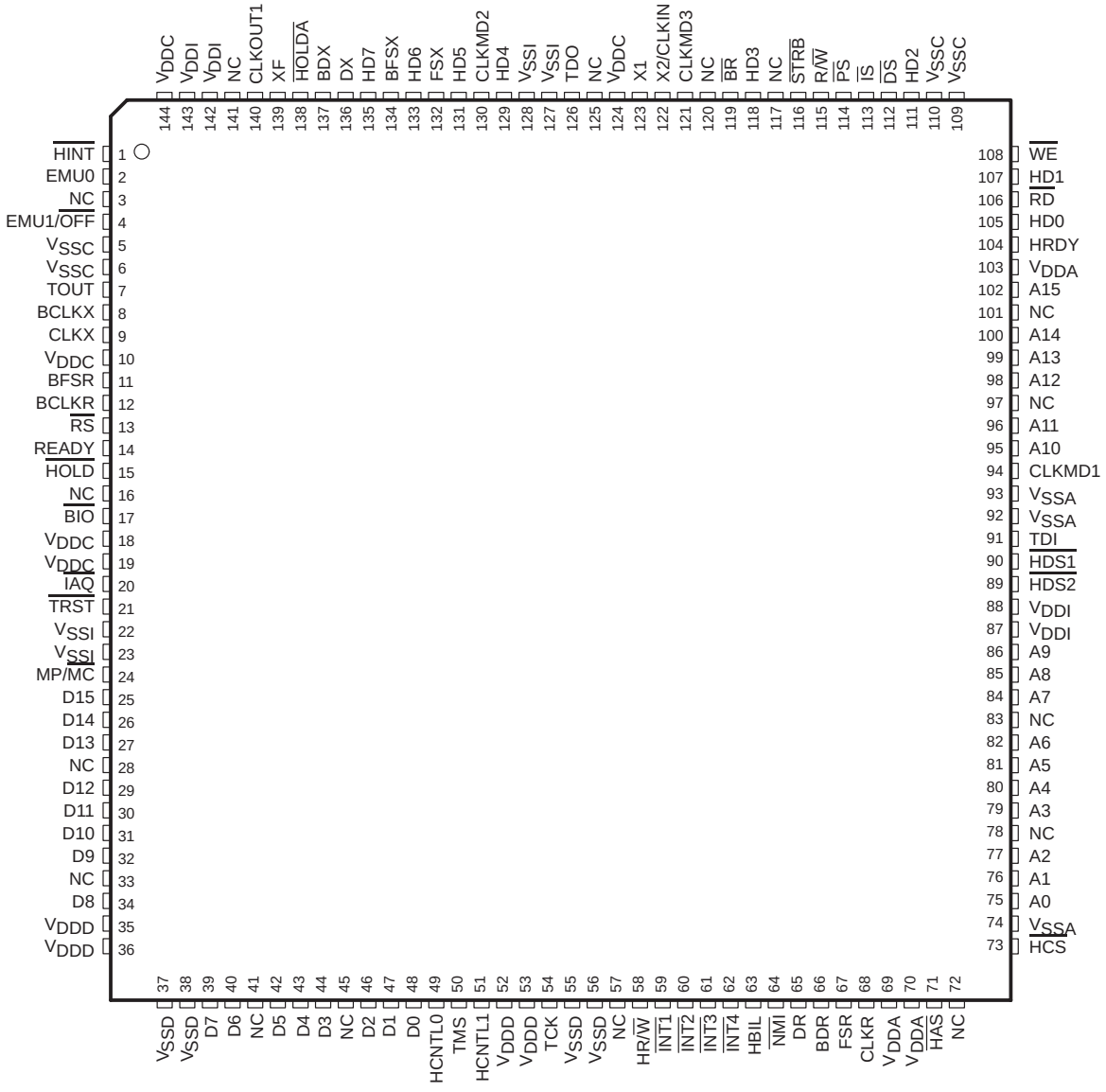
LEGEND:
I = Input
O = Output
Z = High impedance

Pin Functions for the TMS320C52, TMS320LC52 in the PJ Package (Continued)

SIGNAL	TYPE	DESCRIPTION
CLOCK GENERATION AND CONTROL		
X1	O	Oscillator output
X2/CLKIN	I	Clock/oscillator input
CLKIN2	I	Clock input (PLL clock input for ’C52, ’LC52)
CLKMD1, CLKMD2	I	Clock-mode select inputs
CLKOUT1	O/Z	Device system-clock output
POWER SUPPLY CONNECTIONS		
V_{DDA}	S	Supply connection, address-bus output
V_{DDD}	S	Supply connection, data-bus output
V_{DDC}	S	Supply connection, control output
V_{DDI}	S	Supply connection, internal logic
V_{SSA}	S	Supply connection, address-bus output
V_{SSD}	S	Supply connection, data-bus output
V_{SSC}	S	Supply connection, control output
V_{SSI}	S	Supply connection, internal logic

LEGEND:
I = Input
O = Output
S = Supply

TMS320C57S, TMS320LC57S
PGE PACKAGE
(TOP VIEW)



NOTE: NC = No connect (These pins are reserved.)

ADVANCE INFORMATION

Pin Functions for the TMS320C57S, TMS320LC57S in the PGE Package

SIGNAL	TYPE	DESCRIPTION
PARALLEL INTERFACE BUS		
A0–A15	I/O/Z	16-bit external address bus (MSB: A15, LSB: A0)
D0–D15	I/O/Z	16-bit external data bus (MSB: D15, LSB: D0)
$\overline{PS}$, $\overline{DS}$, $\overline{IS}$	O/Z	Program, data, and I/O space select outputs, respectively
$\overline{STRB}$	I/O/Z	Timing strobe for external cycles and external DMA
R/$\overline{W}$	I/O/Z	Read/write select for external cycles and external DMA
$\overline{RD}$, $\overline{WE}$	O/Z	Read and write strobes, respectively, for external cycles
READY	I	External bus ready/wait-state control input
$\overline{BR}$	I/O/Z	Bus request. Arbitrates global memory and external DMA
SYSTEM INTERFACE/CONTROL SIGNALS		
$\overline{RS}$	I	Reset. Initializes device and sets PC to zero
MP/$\overline{MC}$	I	Microprocessor/microcomputer mode select. Enables internal ROM
$\overline{HOLD}$	I	Puts parallel I/F bus in high-impedance state after current cycle
$\overline{HOLDA}$	O/Z	Hold acknowledge. Indicates external bus in hold state
XF	O/Z	External flag output. Set/cleared through software
$\overline{BIO}$	I	I/O branch input. Implements conditional branches
TOUT	O/Z	Timer output signal. Indicates output of internal timer
$\overline{IAQ}$	O/Z	Instruction acquisition signal
$\overline{INT1}$–$\overline{INT4}$	I	External interrupt inputs
$\overline{NMI}$	I	Nonmaskable external interrupt
SERIAL PORT INTERFACE (SPI)		
DR	I	Serial receive-data input
DX	O/Z	Serial transmit-data output. In high-impedance state when not transmitting
CLKR	I	Serial receive-data clock input
CLKX	I/O/Z	Serial transmit-data clock. Internal or external source
FSR	I	Serial receive-frame-synchronization input
FSX	I/O/Z	Serial transmit-frame-synchronization signal. Internal or external source
HOST PORT INTERFACE (HPI)		
HCNTL0	I	HPI mode control 1
HCNTL1	I	HPI mode control 2
$\overline{HINT}$	O/Z	Host interrupt
$\overline{HDS1}$	I	HPI data strobe 1
$\overline{HDS2}$	I	HPI data strobe 2
HR/$\overline{W}$	I	HPI read/write strobe
$\overline{HAS}$	I	HPI address strobe
HRDY	O/Z	HPI ready signal
$\overline{HCS}$	I	HPI chip select
HBIL	I	HPI byte identification input
HD0–HD7	I/O/Z	HPI data bus

LEGEND:
I = Input
O = Output
Z = High impedance

Pin Functions for the TMS320C57S, TMS320LC57S in the PGE Package (Continued)

SIGNAL	TYPE	DESCRIPTION
BUFFERED SERIAL PORT		
BDR	I	BSP receive data input
BDX	O/Z	BSP transmit data output; in high-impedance state when not transmitting
BCLKR	I	BSP receive-data clock input
BCLKX	I/O/Z	BSP transmit-data clock; internal or external source
BFSR	I	BSP receive frame-synchronization input
BFSX	I/O/Z	BSP transmit frame-synchronization signal; internal or external source
EMULATION/JTAG INTERFACE		
TDI	I	JTAG-test-port scan data input
TDO	O/Z	JTAG-test-port scan data output
TMS	I	JTAG-test-port mode select input
TCK	I	JTAG-port clock input
$\overline{\text{TRST}}$	I	JTAG-port reset (with pulldown resistor). Disables JTAG when low
EMU0	I/O/Z	Emulation control 0. Reserved for emulation use
EMU1/$\overline{\text{OFF}}$	I/O/Z	Emulation control 1. Puts outputs in high-impedance state when low
CLOCK GENERATION AND CONTROL		
X1	O	Oscillator output
X2/CLKIN	I	PLL clock input
CLKMD1, CLKMD2, CLKMD3	I	Clock-mode select inputs
CLKOUT1	O/Z	Device system-clock output
POWER SUPPLY CONNECTIONS		
V_{DDA}	S	Supply connection, address-bus output
V_{DDD}	S	Supply connection, data-bus output
V_{DDC}	S	Supply connection, control output
V_{DDI}	S	Supply connection, internal logic
V_{SSA}	S	Supply connection, address-bus output
V_{SSD}	S	Supply connection, data-bus output
V_{SSC}	S	Supply connection, control output
V_{SSI}	S	Supply connection, internal logic

LEGEND:
I = Input
O = Output
S = Supply
Z = High impedance

architecture

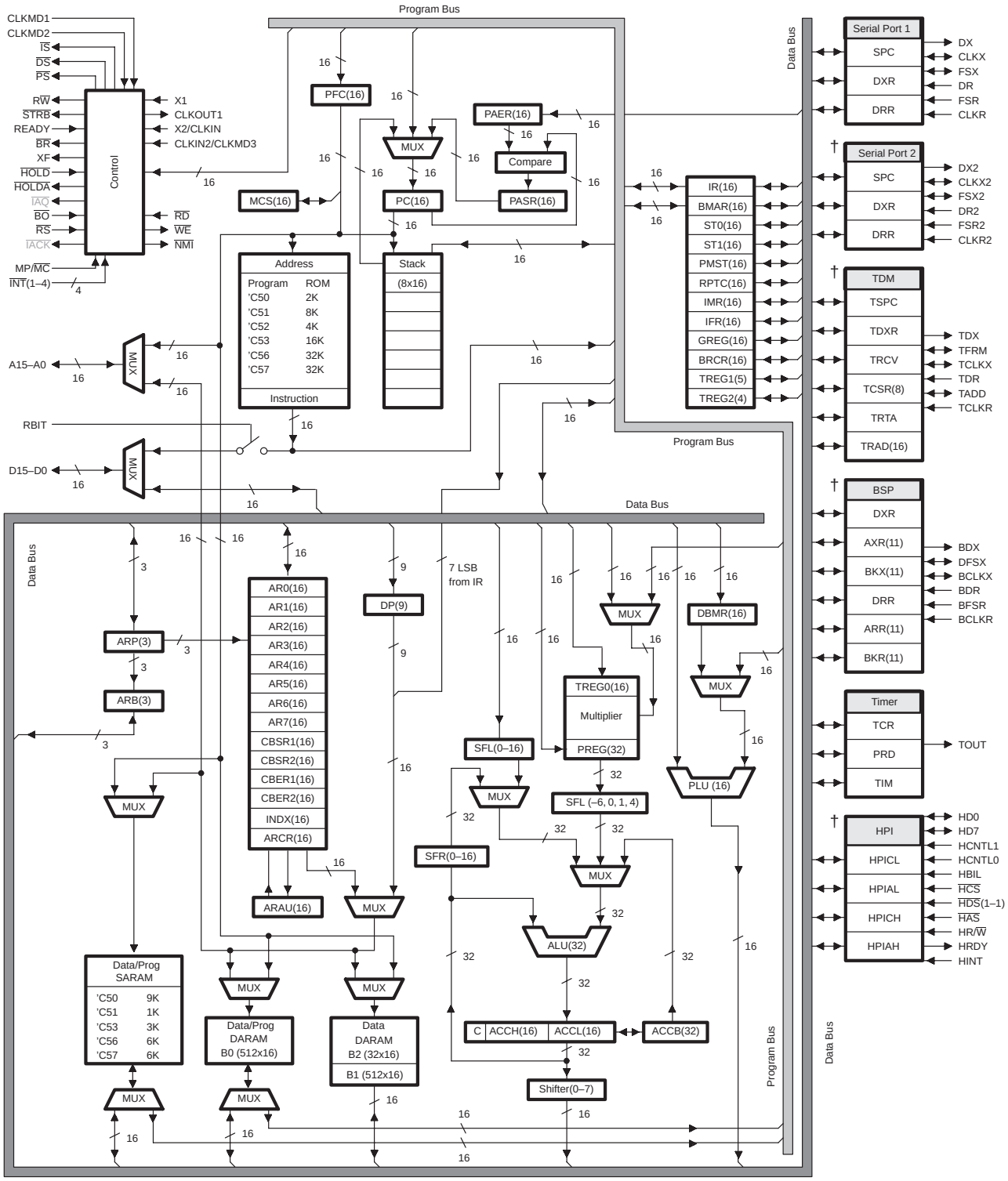
The ’C5x’s advanced Harvard-type architecture maximizes the processing power by maintaining two separate memory bus structures, program and data, for full-speed execution. Instructions support data transfers between the two spaces. This architecture permits coefficients stored in program memory to be read into the RAM, eliminating the need for a separate coefficient ROM. The ’C5x architecture also makes available immediate instructions and subroutines based on computed values. Increased throughput on the ’C5x for many DSP applications is accomplished using single-cycle multiply/accumulate instructions with a data-move option, up to eight auxiliary registers with a dedicated arithmetic unit, a parallel logic unit, and faster I/O necessary for data-intensive signal processing. The architectural design emphasizes overall speed, communication, and flexibility in processor configuration. Control signals and instructions provide floating-point support, block-memory transfers, communication to slower off-chip devices, and multiprocessing implementations as shown in the functional block diagram.

Table 3 explains the symbols that are used in the functional block diagram.

Table 3. Symbols Used in Functional Block Diagram

SYMBOL	DESCRIPTION	SYMBOL	DESCRIPTION
ABU	Auto-buffering unit	IFR	Interrupt-flag register
ACCB	Accumulator buffer	IMR	Interrupt-mask register
ACCH	Accumulator high	INDX	Indirect-addressing-index register
ACCL	Accumulator low	IR	Instruction register
ALU	Arithmetic logic unit	MCS	Microcall stack
ARAU	Auxiliary-register arithmetic unit	MUX	Multiplexer
ARB	Auxiliary-register pointer buffer	PAER	Block-repeat-address end register
ARCR	Auxiliary-register compare register	PASR	Block-repeat-address start register
ARP	Auxiliary-register pointer	PC	Program counter
ARR	Address-receive register (ABU)	PFC	Prefetch counter
AR0–AR7	Auxiliary registers	PLU	Parallel logic unit
AXR	Address-transmit register (ABU)	PMST	Processor-mode-status register
BKR	Receive-buffer-size register (ABU)	PRD	Timer-period register
BKX	Transmit-buffer-size register (ABU)	PREG	Product register
BMAR	Block-move-address register	RPTC	Repeat-counter register
BRCR	Block-repeat-counter register	SARAM	Single-access RAM
BSP	Buffered serial port	SFL	Left shifter
C	Carry bit	SFR	Right shifter
CBER1	Circular buffer 1 end address	SPC	Serial-port interface-control register
CBER2	Circular buffer 2 end address	ST0,ST1	Status registers
CBSR1	Circular buffer 1 start address	TCSR	TDM channel-select register
CBSR2	Circular buffer 2 start address	TCR	Timer-control register
DARAM	Dual-access RAM	TDM	Time-division-multiplexed serial port
DBMR	Dynamic bit manipulation register	TDXR	TDM data transmit register
DP	Data memory page pointer	TIM	Timer-count register
DRR	Serial-port data receive register	TRAD	TDM received-address register
DXR	Serial-port data transmit register	TRCV	TDM data-receive register
GREG	Global memory allocation register	TREG0	Temporary register for multiplication
HPI	Host port interface	TREG1	Temporary register for dynamic shift count
HPIAH	HPI-address register (high bytes)	TREG2	Temporary register used as bit pointer in dynamic-bit test
HPIAL	HPI-address register (low bytes)	TRTA	TDM receive-/transmit-address register
HPICH	HPI-control register (high bytes)	TSPC	TDM serial-port-control register
HPICL	HPI-control register (low bytes)		

functional block diagram



† Not available on all devices (see Table 1).

NOTES: A. Signals in shaded text are not available on 100-pin QFP packages.
B. Symbol descriptions appear in Table 3.

32-bit ALU/accumulator

The 32-bit ALU and accumulator implement a wide range of arithmetic and logical functions, the majority of which execute in a single cycle. The ALU is a general-purpose arithmetic/logic unit that operates on 16-bit words taken from data memory or derived from immediate instructions. In addition to the usual arithmetic instructions, the ALU can perform Boolean operations, facilitating the bit manipulation ability required of a high-speed controller. One input to the ALU always is supplied by the accumulator, and the other input can be furnished from the product register (PREG) of the multiplier, the accumulator buffer (ACCB), or the output of the scaling shifter [which has been read from data memory or from the accumulator (ACC)]. After the ALU performs the arithmetic or logical operation, the result is stored in the ACC where additional operations, such as shifting, can be performed. Data input to the ALU can be scaled by the scaling shifter. The 32-bit ACC is split into two 16-bit segments for storage in data memory. Shifters at the output of the ACC provide a left shift of 0 to 7 places. This shift is performed while the data is being transferred to the data bus for storage. The contents of the ACC remain unchanged. When the postscaling shifter is used on the high word of the ACC (bits 31–16), the most significant bits (MSBs) are lost and the least significant bits (LSBs) are filled with bits shifted in from the low word (bits 15–0). When the postscaling shifter is used on the low word, the LSBs are filled with zeros.

The ’C5x supports floating-point operations for applications requiring a large dynamic range. By performing left shifts, the normalization instruction (NORM) is used to normalize fixed-point numbers contained in the ACC. The four bits of the TREG1 define a variable shift through the scaling shifter for the ADDT/LACT/SUBT instructions (add to/load to/subtract from ACC with shift specified by TREG1). These instructions are useful in denormalizing a number (converting from floating point to fixed point). They are also useful for executing an automatic gain control (AGC) going into a filter.

The single-cycle 1-bit to 16-bit right shift of the ACC efficiently aligns the ACC’s contents. This, coupled with the 32-bit temporary buffer on the ACC, enhances the effectiveness of the ALU in extended-precision arithmetic. The ACCB provides a temporary storage place for a fast save of the ACC. The ACCB also can be used as an input to the ALU. The minimum or maximum value in a string of numbers is found by comparing the contents of the ACCB with the contents of the ACC. The minimum or maximum value is placed in both registers, and, if the condition is met, the carry bit (C) is set to 1. The minimum and maximum functions are executed by the CRLT and CRGT instructions, respectively.

scaling shifters

The ’C5x provides a scaling shifter that has a 16-bit input connected to the data bus and a 32-bit output connected to the ALU. This scaling shifter produces a left shift of 0 to 16 bits on the input data. The shift count is specified by a constant embedded in the instruction word or by the value in TREG1. The LSBs of the output are filled with zeros; the MSBs may be either filled with zeros or sign extended, depending upon the value of the sign-extension mode (SXM) bit of status register ST1.

The ’C5x also contains several other shifters that allow it to perform numerical scaling, bit extraction, extended-precision arithmetic, and overflow prevention. These shifters are connected to the output of the product register and the ACC.

parallel logic unit

The parallel logic unit (PLU) is a second logic unit, additional to the main ALU, that executes logic operations on data without affecting the contents of the ACC. The PLU provides the bit-manipulation ability required of a high-speed controller and simplifies control/status register operations. The PLU provides a direct logic operation path to data memory space and can set, clear, test, or toggle multiple bits directly in a data memory location, a control/status register, or any register that is mapped into data memory space.

16 × 16-bit parallel multiplier

The ’C5x uses a 16 × 16-bit hardware multiplier that is capable of computing a signed or an unsigned 32-bit product in a single machine cycle. All multiply instructions, except the MPYU (multiply unsigned) instruction, perform a signed multiply operation in the multiplier. That is, two numbers being multiplied are treated as 2s-complement numbers, and the result is a 32-bit 2s-complement number.

There are two registers associated with the multiplier: TREG0, a 16-bit temporary register that holds one of the operands for the multiplier, and PREG, the 32-bit product register that holds the product. Four product shift modes (PM) are available at the PREG’s output. These shift modes are useful for performing multiply/accumulate operations, performing fractional arithmetic, or justifying fractional products. The PM field of status register ST1 specifies the PM shift mode.

The product can be shifted one bit to compensate for the extra sign bit gained in multiplying two 16-bit 2s-complement numbers (MPY). A 4-bit shift is used in conjunction with the MPY instruction with a short immediate value (13 bits or less) to eliminate the four extra sign bits gained in multiplying a 16-bit number by a 13-bit number. Finally, the output of PREG can, instead, be right-shifted 6 bits to enable the execution of up to 128 consecutive multiply/accumulates without the possibility of overflow.

The load-TREG0 (LT) instruction normally loads TREG0 to provide one operand (from the data bus), and the MPY instruction provides the second operand (also from the data bus). A multiplication also can be performed with a short or long immediate operand by using the MPY instruction with an immediate operand. A product is obtained every two cycles except when a long immediate operand is used.

Four multiply/accumulate instructions (MAC, MACD, MADD, and MADS as defined in Table 7) fully utilize the computational bandwidth of the multiplier, allowing both operands to be processed simultaneously. The data for these operations is transferred to the multiplier during each cycle through the program and data buses. This facilitates single-cycle multiply/accumulates when used with repeat (RPT and RPTZ) instructions. In these instructions, the coefficient addresses are generated by the PC, while the data addresses are generated by the ARAU. This allows the repeated instruction to access the values sequentially from the coefficient table and step through the data in any of the indirect addressing modes. The RPTZ instruction also clears the accumulator and the product register to initialize the multiply/accumulate operation.

The MACD and MADD instructions, when repeated, support filter constructs (weighted running averages) so that as the sum-of-products is executed, the sample data is shifted in memory to make room for the next sample and to eliminate the oldest sample. Circular addressing with MAC and MADS instructions also can be used to support filter implementation.

auxiliary registers and auxiliary-register arithmetic unit (ARAU)

The ’C5x provides a register file containing eight auxiliary registers (AR0–AR7). The auxiliary registers are used for indirect addressing of the data memory or for temporary data storage. Indirect auxiliary-register addressing allows placement of the data memory address of an instruction operand into one of the auxiliary registers. These registers are referenced with a 3-bit auxiliary register pointer (ARP) that is loaded with a value from 0 through 7, designated AR0 through AR7, respectively. The auxiliary registers and the ARP can be loaded from data memory, the ACC, the product register, or by an immediate operand defined in the instruction. The contents of these registers can be stored in data memory or used as inputs to the central arithmetic logic unit (CALU). These registers are accessible as memory-mapped locations within the ’C5x data-memory space.

The auxiliary register file (AR0–AR7) is connected to the auxiliary register arithmetic unit (ARAU). The ARAU can autoindex the current auxiliary register while the data memory location is being addressed. Indexing can be performed either by ±1 or by the contents of the INDX register. As a result, accessing tables of information does not require the CALU for address manipulation; thus, the CALU is free for other operations in parallel.

memory

The ’C5x implements three separate address spaces for program memory, data memory, and I/O. Each space accommodates a total of 64K 16-bit words (see Figures 1 through 7). Within the 64K words of data space, the 256 to 32K words at the top of the address range can be defined to be external global memory in increments of powers of two, as specified by the contents of the global memory allocation register (GREG). Access to global memory is arbitrated using the global memory bus request ($\overline{BR}$) signal.

The ’C5x devices include a considerable amount of on-chip memory to aid in system performance and integration including ROM, single-access RAM (SARAM), and dual-access RAM (DARAM). The amount and types of memory available on each device are shown in Table 1.

On the ’C5x, the first 96 (0–5Fh) data-memory locations are allocated for memory-mapped registers. This memory-mapped register space contains various control and status registers including those for the CPU, serial port, timer, and software wait-state generators. Additionally, the first 16 I/O port locations are mapped into this data-memory space, allowing them to be accessed either as data memory using single-word instructions or as I/O locations with two-word instructions. Two-word instructions allow access to the full 64K words of I/O space.

The mask-programmable ROM is located in program memory space. Customers can arrange to have this ROM programmed with contents unique to to any particular application. The ROM is enabled or disabled by the state of the MP/$\overline{MC}$ control input upon resetting the device or by manipulating the MP/$\overline{MC}$ bit in the PMST status register after reset. The ROM occupies the lowest block of program memory when enabled. When disabled, these addresses are located in the device’s external program-memory space.

The ’C5x also has a mask-programmable option that provides security protection for the contents of on-chip ROM. When this internal option bit is programmed, no externally-originating instruction can access the on-chip ROM. This feature can be used to provide security for proprietary algorithms.

An optional boot loader is available in the device’s on-chip ROM. This boot loader can be used to transfer a program automatically from data memory or the serial port to anywhere in program memory. In data memory, the program can be located on any 1K-word boundary and can be in either byte-wide or 16-bit word format. Once the code is transferred, the boot loader releases control to the program for execution.

The ’C5x devices provide two types of RAM: single-access RAM (SARAM) and dual-access RAM (DARAM). The single-access RAM requires a full machine cycle to perform a read or a write; however, this is not one large RAM block in which only one access per cycle is allowed. It is made up of 2K-word size-independent RAM blocks and each one allows one CPU access per cycle. The CPU can read or write one block while accessing another block at the same time. All ’C5x processors support multiple accesses to its SARAM in one cycle as long as they go to different RAM blocks. If the total SARAM size is not a multiple of two, one block is made smaller than 2K words. With an understanding of this structure, programmers can arrange code and data appropriately to improve code performance. Table 4 shows the sizes of available SARAM on the applicable ’C5x devices.

Table 4. SARAM Block Sizes

DEVICE	NUMBER OF SARAM BLOCKS
’C50 / ’LC50	Four 2K blocks and one 1K block
’C51 / ’LC51	One 1K block
’C53 / ’C53S / ’LC53	One 2K block and one 1K block
’LC56	Three 2K blocks
’C57S / ’LC57/’LC57S	Three 2K blocks

memory (continued)

The ’C5x dual-access RAM (DARAM) allows writes to, and reads from, the RAM in the same cycle without the address restrictions of the SARAM. The dual-access RAM is configured in three blocks: block 0 (B0), block 1 (B1), and block 2 (B2). Block 1 is 512 words in data memory and block 2 is 32 words in data memory. Block 0

is a 512-word block which can be configured as data or program memory. The CLRC CNF (configure B0 as data memory) and SETC CNF (configure B0 as program memory) instructions allow dynamic configuration of the memory maps through software. When using block 0 as program memory, instructions can be downloaded from external program memory into on-chip RAM and then executed.

When using on-chip RAM, ROM, or high-speed external memory, the ’C5x runs at full speed with no wait states. The ability of the DARAM to allow two accesses to be performed in one cycle, coupled with the parallel nature of the ’C5x architecture, enables the device to perform three concurrent memory accesses in any given machine cycle. Externally, the READY line can be used to interface the ’C5x to slower, less expensive external memory. Downloading programs from slow off-chip memory to on-chip RAM can speed processing while cutting system costs.

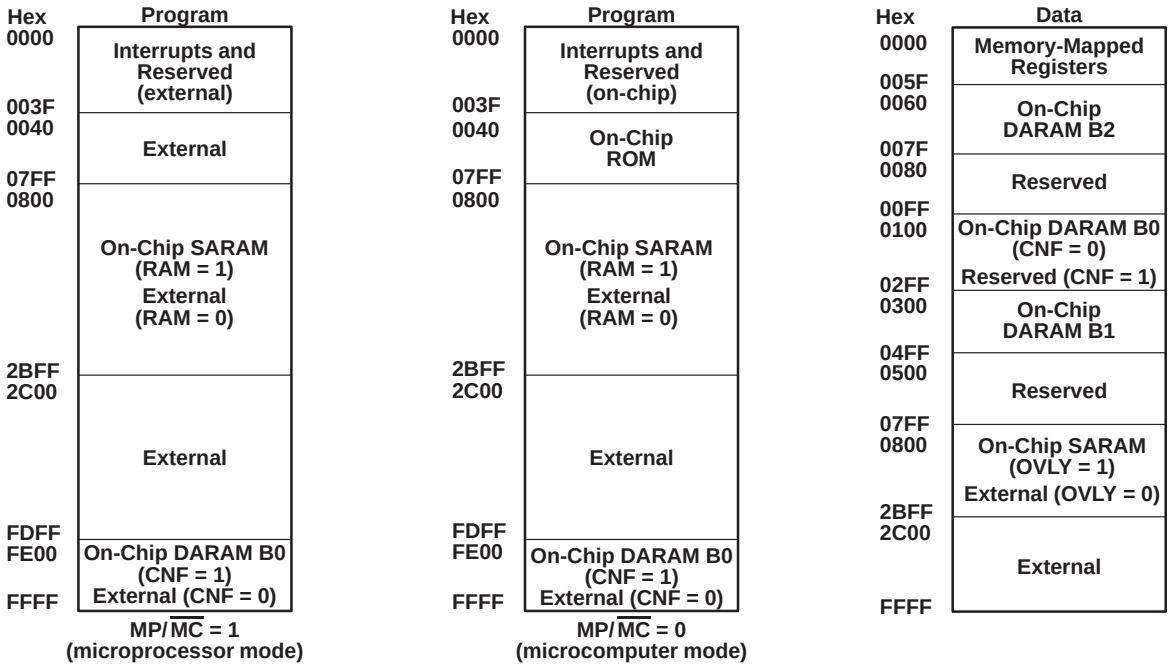


Figure 1. TMS320C50 and TMS320LC50 Memory Map

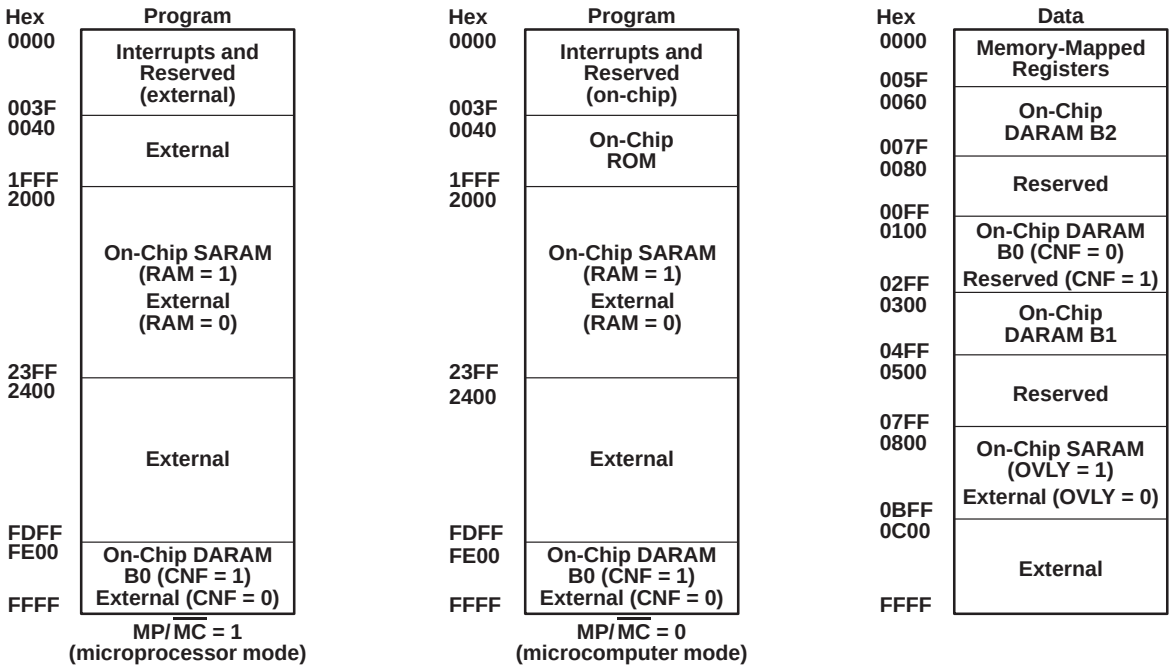


Figure 2. TMS320C51 and TMS320LC51 Memory Map

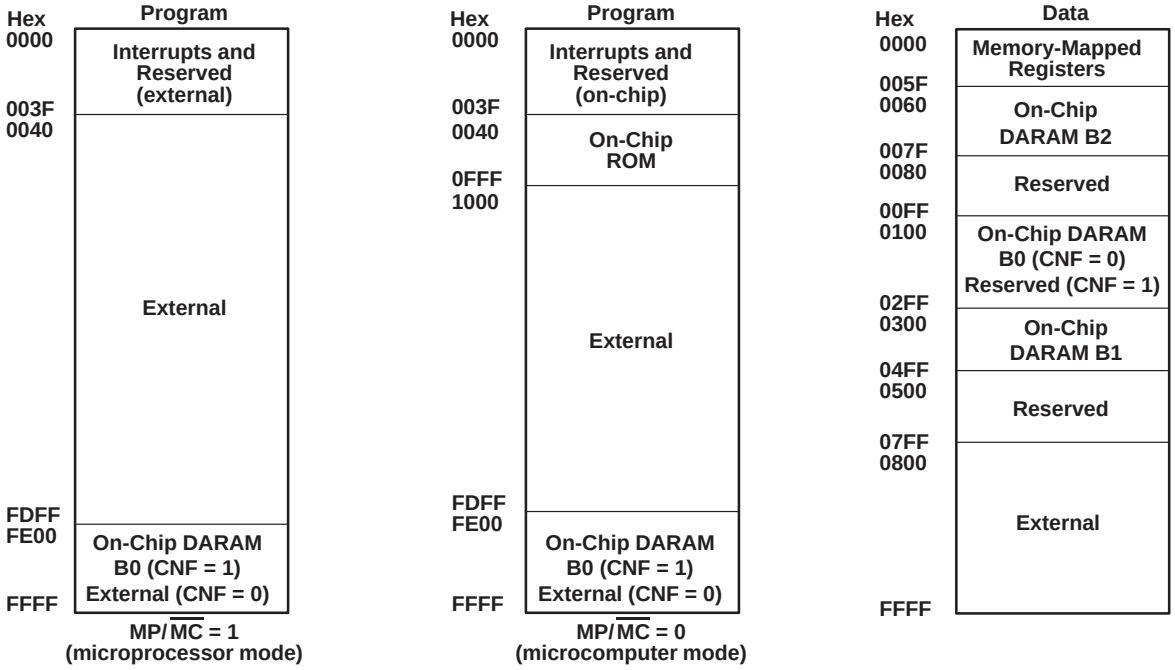


Figure 3. TMS320C52 and TMS320LC52 Memory Map

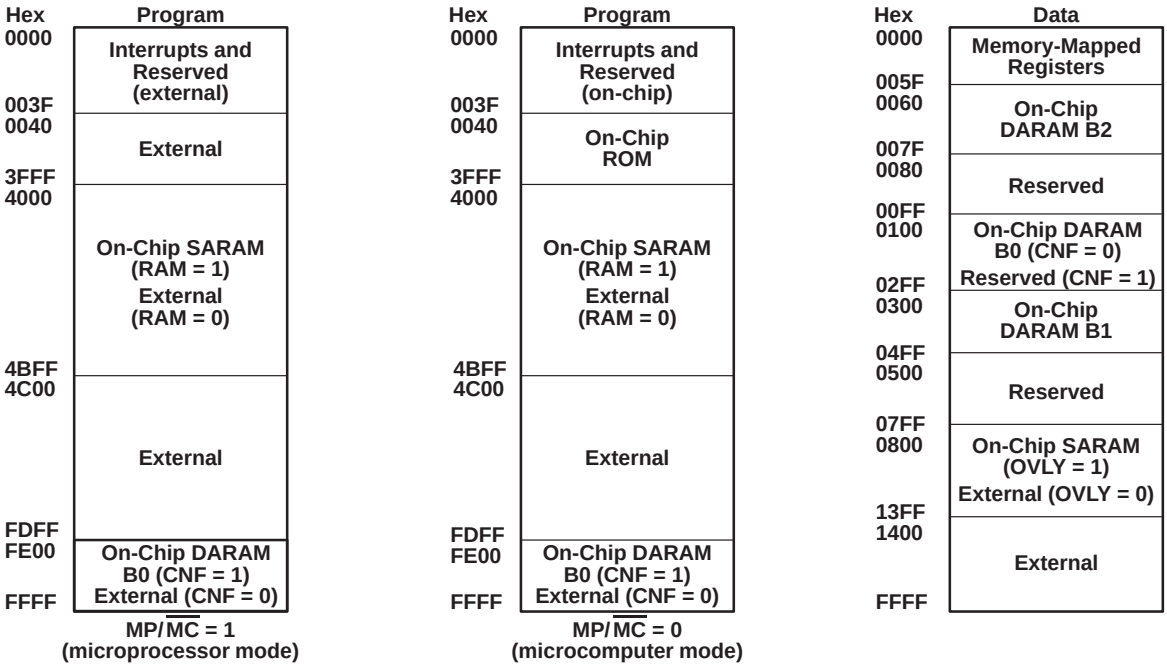


Figure 4. TMS320C53, TMS320C53S, TMS320LC53, and TMS320LC53S Memory Map

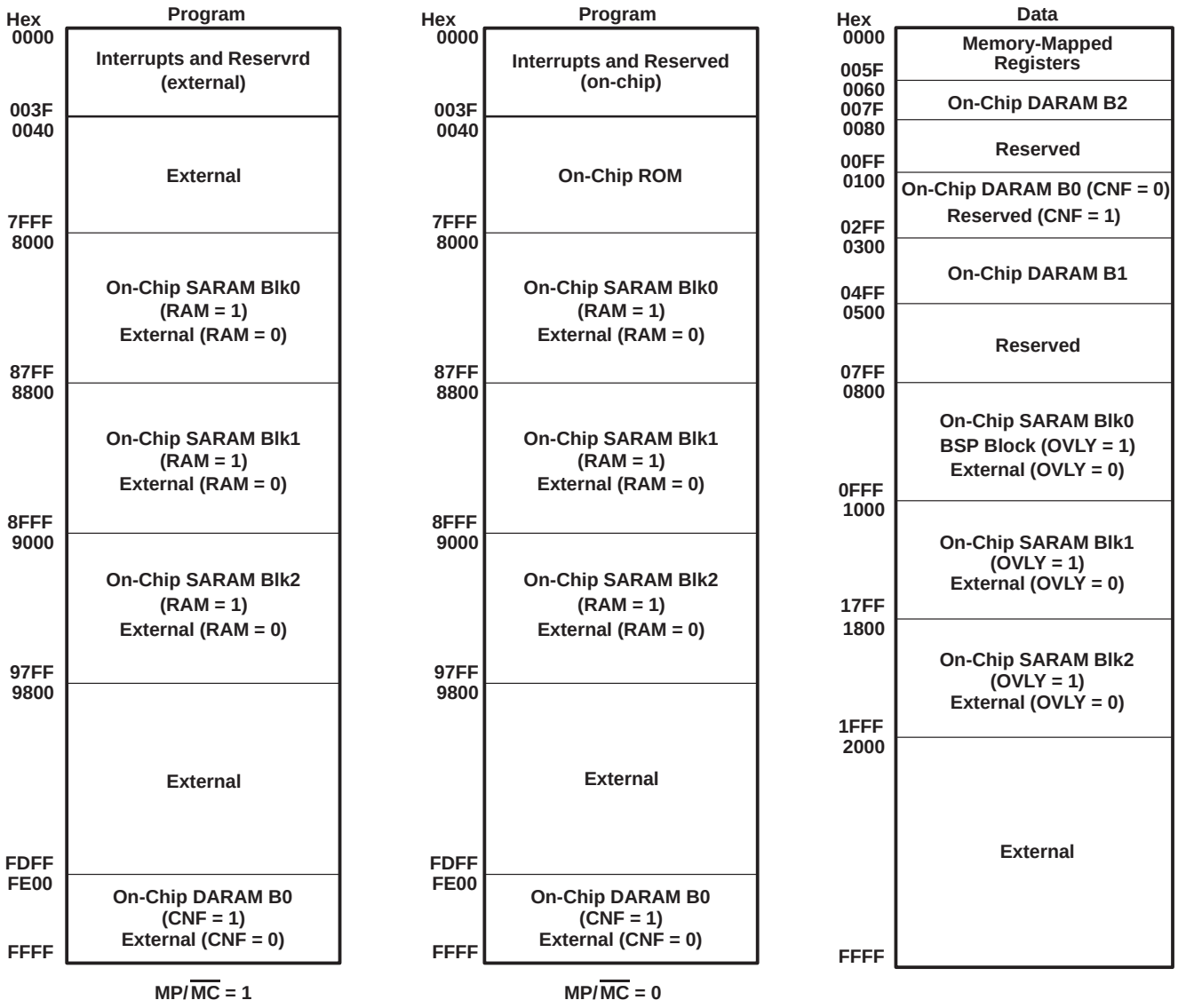


Figure 5. TMS320LC56 Memory Map

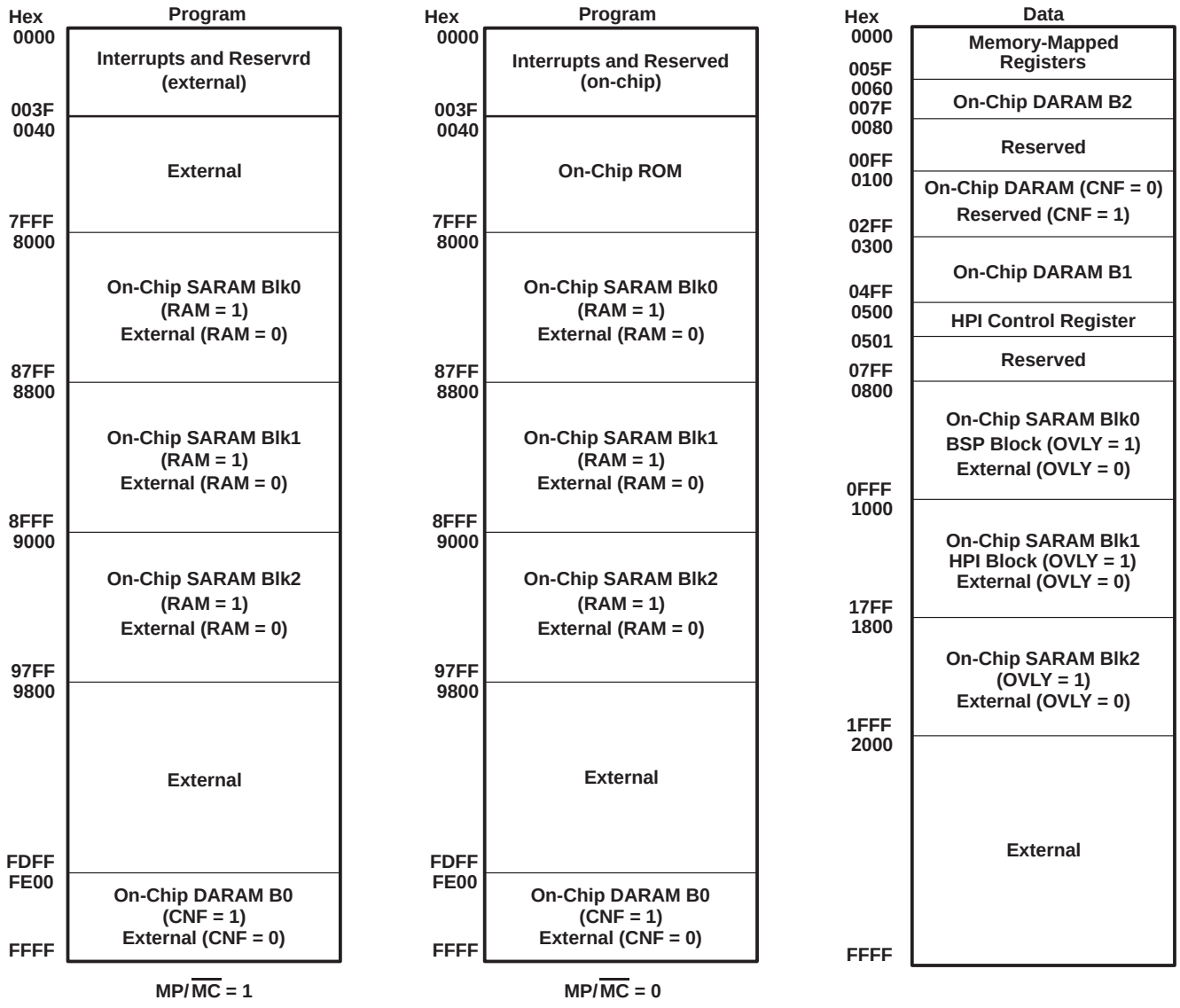


Figure 6. TMS320LC57 Memory Map

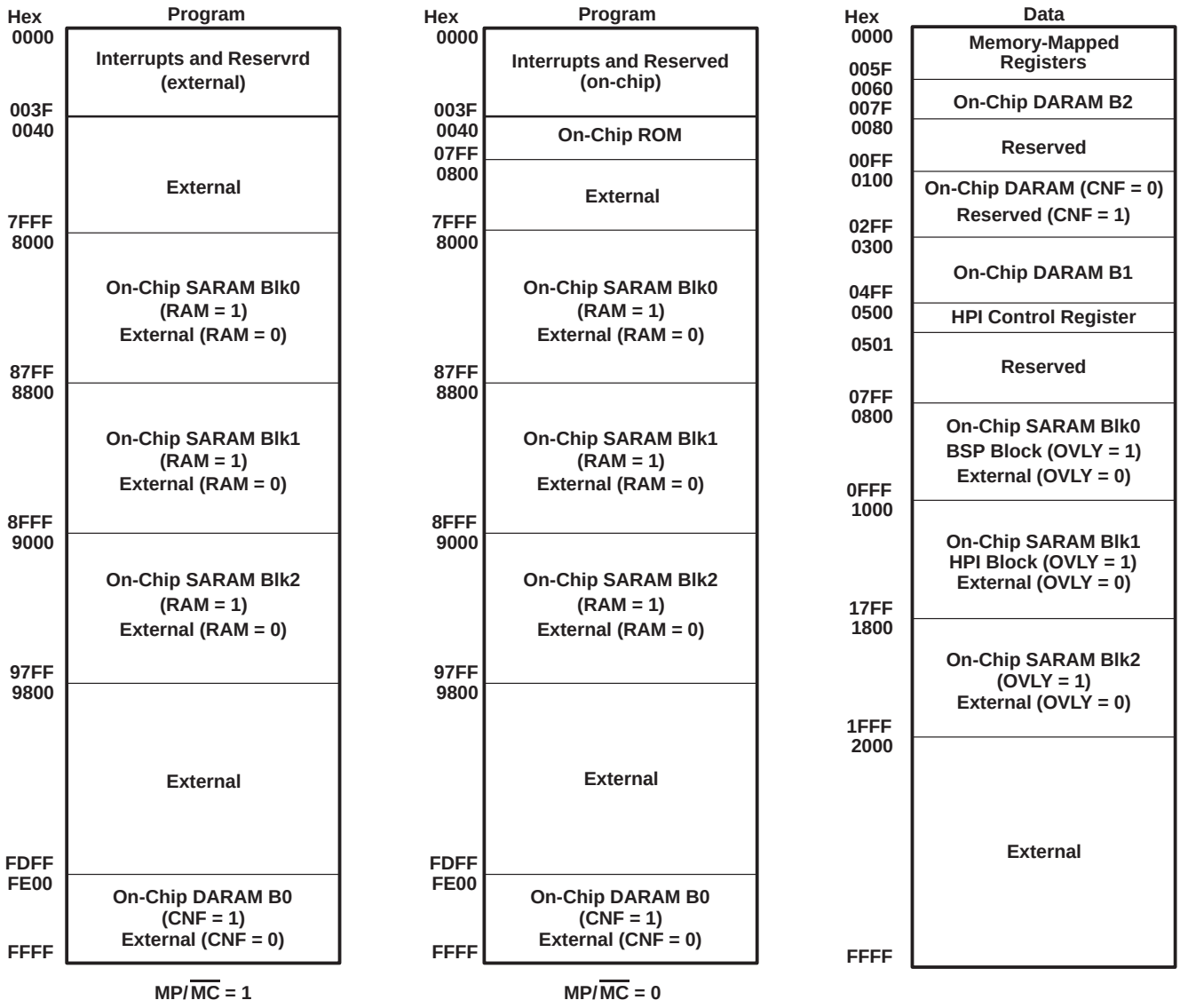


Figure 7. TMS320C57S Memory Map

interrupts and subroutines

The ’C5x implements four general-purpose interrupts, $\overline{\text{INT4}}$–$\overline{\text{INT1}}$, along with reset ($\overline{\text{RS}}$) and the nonmaskable interrupt ($\overline{\text{NMI}}$) which are available for external devices to request the attention of the processor. Internal interrupts are generated by the serial port (RINT and XINT), by the timer (TINT), and by the software-interrupt (TRAP, INTR, and NMI) instructions. Interrupts are prioritized with $\overline{\text{RS}}$ having the highest priority, followed by $\overline{\text{NMI}}$, and $\overline{\text{INT4}}$ having the lowest priority. Additionally, any interrupt except $\overline{\text{RS}}$ and $\overline{\text{NMI}}$ can be masked individually with a dedicated bit in the interrupt mask register (IMR) and can be cleared, set, or tested using its own dedicated bit in the interrupt flag register (IFR). The reset and NMI functions are not maskable.

All interrupt vector locations are on two-word boundaries so that branch instructions can be accommodated in those locations. While normally located at program memory address 0, the interrupt vectors can be remapped to the beginning of any 2K-word page in program memory by modifying the contents of the interrupt vector pointer (IPTR) located in the PMST status register.

A built-in mechanism protects multicycle instructions from interrupts. If an interrupt occurs during a multicycle instruction, the interrupt is not processed until the instruction completes execution. This mechanism applies to instructions that are repeated (using the RPT instruction) and to instructions that become multicycle because of wait states.

Each time an interrupt is serviced or a subroutine is entered, the PC is pushed onto an internal hardware stack, providing a mechanism for returning to the previous context. The stack contains eight locations, allowing interrupts or subroutines to be nested up to eight levels deep.

In addition to the eight-level hardware PC stack, eleven key CPU registers are equipped with an associated single-level stack or shadow register into which the registers’ contents are saved upon servicing an interrupt. The contents are restored into their particular CPU registers once a return-from-interrupt instruction (RETE or RETI) is executed. The registers that have the shadow-register feature include the ACC and buffer, product register, status registers, and several other key CPU registers. The shadow-register feature allows sophisticated context save and restore operations to be handled automatically in cases where nested interrupts are not required or if interrupt servicing is performed serially.

power-down modes

The ’C5x implements several power-down modes in which the ’C5x core enters a dormant state and dissipates considerably less power. A power-down mode is invoked either by executing the IDLE/IDLE2 instructions or by driving the $\overline{\text{HOLD}}$ input low. When the $\overline{\text{HOLD}}$ signal initiates the power-down mode, on-chip peripherals continue to operate; this power-down mode is terminated when $\overline{\text{HOLD}}$ goes inactive.

While the ’C5x is in a power-down mode, all internal contents are maintained; this allows operation to continue unaltered when the power-down mode is terminated. All CPU activities are halted when the IDLE instruction is executed, but the CLKOUT1 pin remains active. The peripheral circuits continue to operate, allowing peripherals such as serial ports and timers to take the CPU out of its powered-down state. A power-down mode, when initiated by an IDLE instruction, is terminated upon receipt of an interrupt.

The IDLE2 instruction is used for a complete shutdown of the core CPU as well as all on-chip peripherals. In IDLE2, the power is reduced significantly because the entire device is stopped. The power-down mode is terminated by activating any of the external interrupt pins ($\overline{\text{RS}}$, $\overline{\text{NMI}}$, $\overline{\text{INT1}}$, $\overline{\text{INT2}}$, $\overline{\text{INT3}}$, and $\overline{\text{INT4}}$) for at least five machine cycles.

bus-keeper circuitry (TMS320LC56/’C57S/’LC57)

The TMS320LC56/’C57S/’LC57 devices provide built-in bus keeper circuitry which holds the last state driven on the data bus by either the DSP or an external device after the bus is no longer being driven. This capability prevents excess power consumption caused by a floating bus, thus allowing optimization of power consumption without the need for external pullup resistors.

external interface

The ’C5x supports a wide range of system interfacing requirements. Program, data, and I/O address spaces provide interface to memory and I/O, maximizing system throughput. The full 16-bit address and data bus, along with the $\overline{PS}$, $\overline{DS}$, and $\overline{IS}$ space select signals, allow addressing of 64K 16-bit words in each of the three spaces.

I/O design is simplified by having I/O treated the same way as memory. I/O devices are mapped into the I/O address space using the processor’s external address and data buses in the same manner as memory-mapped devices.

The ’C5x external parallel interface provides various control signals to facilitate interfacing to the device. The R/$\overline{W}$ output signal is provided to indicate whether the current cycle is a read or a write. The $\overline{STRB}$ output signal provides a timing reference for all external cycles. For convenience, the device also provides the $\overline{RD}$ and the $\overline{WE}$ output signals, which indicate a read and a write cycle, respectively, along with timing information for those cycles. The availability of these signals minimizes external gating necessary for interfacing external devices to the ’C5x.

Interface to memory and I/O devices of varying speeds is accomplished by using the READY line. When transactions are made with slower devices, the ’C5x processor waits until the other device completes its function and signals the processor via the READY line. Once a ready indication is provided back to the ’C5x from the external device, execution continues.

The bus request ($\overline{BR}$) signal is used in conjunction with the other ’C5x interface signals to arbitrate external global-memory accesses. Global memory is external data-memory space in which the $\overline{BR}$ signal is asserted at the beginning of the access. When an external global-memory device receives the the bus request, the external device responds by asserting the READY signal after the global memory access is arbitrated and the global access is completed.

external direct-memory access (DMA) capability

All ’C5x devices with single-access RAM offer a unique feature allowing another processor to read and write to the ’C5x internal memory. To initiate a read or write operation to the ’C5x single-access RAM, the host or master processor requests a hold state on the DSP’s external bus. When acknowledged with $\overline{HOLDA}$, the host can request access to the internal bus by pulling the $\overline{BR}$ signal low. Unlike the hold mode, which allows the current operation to complete and allows CPU operation to continue (if status bit HM=0), a $\overline{BR}$-requested DMA always halts the operation currently being executed by the CPU. Access to the internal bus always is granted on the third clock cycle after the $\overline{BR}$ signal is received. In the PQ package, the $\overline{IAQ}$ pin also indicates when bus access has been granted. In the PZ package, this pin is not present so the host is required to wait two clock cycles after driving the bus request low before beginning DMA transfer.

host port interface (HPI) (TMS320C57S, TMS320LC57, TMS320LC57S only)

The HPI is an 8-bit parallel port used to interface a host processor to the ’C57S/’LC57. The host port is connected to a 2k word on-chip buffer through a dedicated internal bus. The dedicated bus allows the CPU to work uninterrupted while the host processor accesses the host port. The HPI memory buffer is a single-access RAM block which is accessible by both the CPU and the host. The HPI memory also can be used as general-purpose data or program memory. Both the CPU and the host have access to the HPI control register (HPIC) and the host can address the HPI memory through the HPI address register (HPIA).

Data transfers of 16-bit words occur as two consecutive bytes with a dedicated pin, HBIL, indicating whether the high or low byte is being transmitted. Two control pins, HCNTL1 and HCNTL0, control host access to the HPIA, HPI data (with an optional automatic address increment), or the HPIC. The host can interrupt the ’C57S/’LC57 by writing to HPIC. The ’C57S/’LC57 can interrupt the host with a dedicated $\overline{HINT}$ pin that the host acknowledges and clears.

host port interface (continued)

The HPI has two modes of operation, shared-access mode (SAM) and host-only mode (HOM). In SAM, the normal mode of operation, both the ’C57S/’LC57 and the host can access HPI memory. In this mode, asynchronous host accesses are resynchronized internally and, in case of conflict, the host has access priority and the ’C57S/’LC57S waits one cycle. Host and CPU accesses to the HPI memory can be resychronized through polling of a command word or through interrupts to prevent stalling the CPU for one cycle. The HOM capability allows the host to access HPI memory while the ’C57S/’LC57 is in IDLE2 mode (all internal clocks stopped) or in reset mode. The external ’C57S/’LC57S clock even can be stopped. The host can, therefore, access the HPI RAM while the ’C57S/’LC57 is in its optimum configuration in terms of power consumption.

The HPI control register has two data strobes, $\overline{\text{HDS1}}$ and $\overline{\text{HDS2}}$, a read/write strobe HR/$\overline{\text{W}}$, and an address strobe $\overline{\text{HAS}}$, to enable a glueless interface to a variety of industry-standard host devices. The HPI is easily interfaced to hosts with multiplexed address/data bus, separate address and data buses, one data strobe, and a read/write strobe, or two separate strobes for read and write. An HPI-ready pin, HRDY, is provided to specify wait states for hosts that support an asynchronous input. When the ’C57S/’LC57 operating frequency is variable, or when the host is capable of accessing at a faster rate than the maximum shared-access mode access rate, the HRDY pin provides a convenient way to adjust the host access rate automatically (no software handshake needed) to a change in the ’C57S/’LC57 clock rate or an HPI-mode switch.

The HPI supports high-speed back-to-back accesses. In the shared-access mode, the HPI can handle one byte every five ’C57S/’LC57 periods (that is, 64 Mb/s with a 40-MHz ’C57S/’LC57). The HPI is designed so that the host can take advantage of this high bandwidth and run at frequencies up to $(f \times n) \div 5$, where n is the number of host cycles for an external access and f is the ’C57S/’LC57 frequency. In host-only mode, the HPI supports even higher speed back-to-back host accesses: 1 byte every 50 ns (that is, 160 Mb/s) independently of the ’C57S/’LC57 clock rate.

serial ports

The ’C5x provides high-speed full-duplex serial ports that allow direct interface to other ’C5x devices, codecs, and other devices in a system. There is a general-purpose serial port, a time-division-multiplexed (TDM) serial port, and an auto-buffered serial port (BSP).

The general-purpose serial port uses two memory-mapped registers for data transfer: the data-transmit register (DXR) and the data-receive register (DRR). Both registers can be accessed in the same manner as any other memory location. The transmit and receive sections of the serial port each have associated clocks, frame-synchronization pulses, and serial shift registers, and serial data can be transferred either in bytes or in 16-bit words. Serial port receive and transmit operations can generate their own maskable transmit and receive interrupts (XINT and RINT), allowing serial port transfers to be managed by way of software. The ’C5x serial ports are double-buffered and fully static.

The TDM port allows the device to communicate through time-division multiplexing with up to seven other ’C5x devices with TDM ports. Time-division multiplexing is the division of time intervals into a number of subintervals with each subinterval representing a prespecified communications channel. The TDM port serially transmits 16-bit words on a single data line (TDAT) and destination addresses on a single address line (TADD). Each device can transmit data on a single channel and receive data from one or more of the eight channels providing a simple and efficient interface for multiprocessing applications. A frame synchronization pulse occurs once every 128 clock cycles corresponding to transmission of one 16-bit word on each of the eight channels. Like the general-purpose serial port, the TDM port is double-buffered on both input and output data. The TDM port also can be configured in software to operate as a general-purpose serial port as described above. Both types of ports are capable of operating at up to one-fourth the machine cycle rate (CLKOUT1).

The buffered serial port (BSP) consists of a full-duplex double-buffered serial port interface (SPI) and an auto-buffering unit (ABU). The SPI block of the BSP is an enhanced version of the general-purpose serial port. The auto-buffering unit allows the SPI to read/write directly to ’C5x internal memory using a dedicated bus independently of the CPU. This results in minimum overhead for SPI transactions and faster data rates.

serial ports (continued)

When auto-buffering capability is disabled (standard mode), transfers with SPI are performed under software control through interrupts. In this mode, the ABU is transparent and the word-based interrupts (WXINT and WRINT) provided by the SPI are sent to the CPU as transmit interrupt (XINT) and receive interrupt (RINT). When auto buffering is enabled, word transfers are done directly between the SPI and the ’C5x internal memory, using ABU-embedded address generators.

The ABU has its own set of circular addressing registers with corresponding address-generation units. Memory for the buffers resides in 2K words of ’C5x internal memory. The length and starting addresses of the buffers are user-programmable. A buffer-empty/-full interrupt can be posted to the CPU. Buffering is halted easily because of an auto-disabling capability. Auto-buffering capability can be enabled separately for transmit and receive sections. When auto-buffering is disabled, operation is similar to the general-purpose serial port.

The SPI allows transfer of 8-, 10-, 12-, or 16-bit data packets. In burst mode, data packets are directed by a frame-synchronization pulse for every packet. In continuous mode, the frame-synchronization pulse occurs when the data transmission is initiated and no further pulses occur. The frame and clock strobes are frequency and polarity programmable. The SPI is fully static and operates at arbitrarily low clock frequencies. The maximum operating frequency is CLKOUT1 (28.6 Mb/s at 35 ns, 40 Mb/s at 25 ns). The SPI transmit section also includes a pulse-coded modulation (PCM) mode that allows easy interface with a PCM line.

Most ’C5x devices provide one general-purpose serial port and one TDM port. The ’C52 provides one general-purpose serial port and no TDM port. The ’C53SX provides two general-purpose serial ports and no TDM port. The ’LC56, ’C57S, and ’LC57 devices provide one general-purpose serial port and one buffered serial port.

software wait-state generators

Software wait-state generation is incorporated in the ’C5x without any external hardware for interfacing with slower off-chip memory and I/O devices. The circuitry consists of 16 wait-state generating circuits and is user-programmable to operate with 0, 1, 2, 3, or 7 wait states. For off-chip memory accesses, these wait-state generators are mapped on 16K-word boundaries in program memory, data memory, and the I/O ports.

The ’C53S/’C57S and ’LC56/57 devices have software-programmable wait-state generators that are controlled by one 16-bit wait-state register PDWSR at address 0x28. The programmed number of wait states (0 through 7) applies to all external addresses at the corresponding address space (program, data, I/O) regardless of address value.

timer

The ’C5x features a 16-bit timing circuit with a 4-bit prescaler. This timer clocks between one-half and one thirty-second the machine rate of the device itself, depending on the programmable timer’s divide-down ratio. This timer can be stopped, restarted, reset, or disabled by specific status bits.

The timer can be used to generate CPU interrupts periodically. The timer is decremented by one at every CLKOUT1 cycle. A timer interrupt (TINT) and a pulse equal to the duration of a CLKOUT1 cycle on the external TOUT pin are generated each time the counter decrements to zero. The timer provides a convenient means of performing periodic I/O or other functions. When the timer is stopped, the internal clocks to the timer are shut off, allowing the device to run in a low-power mode of operation.

IEEE 1149.1 boundary scan interface

The IEEE 1149.1 boundary-scan interface is used for emulation and test purposes. The IEEE 1149.1 scanning logic provides the boundary-scan path to and from the interfacing devices. Also, it can be used to test pin-to-pin continuity as well as to perform operational tests on those peripheral devices that surround the ’C5x. On ’C5x devices which do not provide boundary-scan capability, the IEEE 1149.1 interface is used for emulation purposes only. It is interfaced to other internal scanning logic circuitry, which has access to all of the on-chip resources. Thus, the ’C5x can perform on-board emulation by means of IEEE 1149.1 serial pins and the emulation-dedicated pins (see IEEE Standard 1149.1 for more details). Table 5 shows IEEE 1149.1 and boundary-scan functions supported by the ’C5x family of devices.

Table 5. IEEE 1149.1 Interface/Boundary Scan/On-Chip Analysis Block Configurations on the ’C5x/’LC5x Device Family

DEVICE TYPE	IEEE 1149.1 INTERFACE	BOUNDARY-SCAN CAPABILITY	ON-CHIP ANALYSIS BLOCK
’C50/’LC50	Yes	Yes	Full
’C51/’LC51	Yes	Yes	Full
’C52/’LC52	Yes	No	Full
’C53/’LC53	Yes	Yes	Full
’C53S/’LC53S	Yes	No	Reduced
’LC56	Yes	No	Full
’C57S	Yes	Yes	Full
’LC57	Yes	No	Full

on-chip analysis block

The on-chip analysis block, in conjunction with the ’C5x EVM, provides the capability to perform a variety of debugging and performance evaluation functions in a target system. The full analysis block provides capability for message passing by a combination of monitor mode and scan, flexible breakpoint setup based on events, counting of events, and a PC discontinuity trace buffer. Breakpoints can be triggered based on the following events: program fetches/reads/writes, EMU0/1 pin activity (used in multiprocessing), data reads/writes, CPU events (calls, returns, interrupts/traps, branches, pipeline clock), and event-counter overflow. The event counter is a 16-bit counter which can be used for performance analysis. The event counter can be incremented based on the occurrence of the following events: CPU clocks (performance monitoring), pipeline advances, instruction fetches (used to count instructions for an algorithm), branches, calls, returns, interrupts/traps, program reads/writes, or data reads/writes. The PC discontinuity-trace buffer provides a method to monitor program counter flow.

These analysis functions are available on all ’C5x devices except the ’C53S and ’LC53S which have a reduced analysis block (see Table 5). The reduced analysis block provides capability for message passing and breakpoints based on program fetches/reads/writes and EMU0/1 pin activity.

multiprocessing

The flexibility of the ’C5x allows configurations to satisfy a wide range of system requirements; the device can be used in a variety of system configurations, including, but not limited to, the following:

- A standalone processor
- A multiprocessor with devices in parallel
- A slave/host multiprocessor with global-memory space
- A peripheral processor interfaced via processor-controlled signals to another device

multiprocessing (continued)

For multiprocessing applications, the ’C5x is capable of allocating global-memory space and communicating with that space via the $\overline{BR}$ and ready control signals. Global memory is data memory shared by more than one device. Global memory access must be arbitrated. The 8-bit memory-mapped global memory allocation register (GREG) specifies part of the ’C5x’s data memory as global external memory. The contents of the register determine the size of the global memory space. If the current instruction addresses an operand within that space, $\overline{BR}$ is asserted to request control of the bus. The length of the memory cycle is controlled by the READY line.

The ’C5x supports direct memory access (DMA) to its external program, data, and I/O spaces using the $\overline{HOLD}$ and $\overline{HOLDA}$ signals. Another device can take complete control of the ’C5x’s external memory interface by asserting $\overline{HOLD}$ low. This causes the ’C5x to to place its address, data, and control lines in the high-impedance state and assert $\overline{HOLDA}$. While external memory is being accessed, program execution from on-chip memory can proceed concurrently when the device is in hold mode.

Multiple ’C5x devices can be interconnected through their serial ports. This form of interconnection allows information to be transferred at high speed while using a minimum number of signal connections. A complete full-duplex serial-port interconnection between multiple processors can be accomplished with as few as four signal lines.

instruction set

The ’C5x microprocessor implements a comprehensive instruction set that supports both numeric-intensive signal processing operations and general-purpose applications, such as multiprocessing and high-speed control. Source code for the ’C1x and ’C2x DSPs is upward compatible with the ’C5x.

For maximum throughput, the next instruction is prefetched while the current one is being executed. Because the same data lines are used to communicate to external data, program, or I/O space, the number of cycles an instruction requires to execute varies, depending on whether the next data operand fetch is from internal or external memory. Highest throughput is achieved by maintaining data memory on chip and using either internal or fast external program memory.

addressing modes

The ’C5x instruction set provides six basic memory-addressing modes: direct, indirect, immediate, register, memory mapped, and circular addressing.

In direct addressing, the instruction word contains the lowest seven bits of the data-memory address. This field is concatenated with the nine bits of the data-memory page pointer (DP) to form the 16-bit data-memory address. Therefore, in the direct-addressing mode, data memory is paged effectively with a total of 512 pages, each of which contains 128 words.

Indirect addressing accesses data memory through the auxiliary registers. In indirect addressing mode, the address of the instruction operand is contained in the currently selected auxiliary register. Eight auxiliary registers (AR0–AR7) provide flexible and powerful indirect addressing. To select a specific auxiliary register, the auxiliary register pointer (ARP) is loaded with a value from 0 to 7 for AR0 through AR7, respectively.

There are seven types of indirect addressing: autoincrement or autodecrement, postindexing by either adding or subtracting the contents of AR0, single-indirect addressing with no increment or decrement, and bit-reversed addressing (used in FFTs) with increment or decrement. All operations are performed on the current auxiliary register in the same cycle as the original instruction, following which the current auxiliary register and ARP can be modified.

addressing modes (continued)

In immediate addressing, the actual operand data is provided in a portion of the instruction word or words. There are two types of immediate addressing: long and short. In short-immediate addressing, the data is contained in a portion of the bits in a single-word instruction. In long-immediate addressing, the data is contained in the second word of a two-word instruction. The immediate-addressing mode is useful for data that does not need to be stored or used more than once during the course of program execution, such as initialization values, constants, etc.

The register-addressing mode uses operands in CPU registers either explicitly, such as with a direct reference to a specific register, or implicitly, with instructions that intrinsically reference certain registers. In either case, operand reference is simplified because 16-bit values can be used without specifying a full 16-bit operand address or immediate value.

Memory-mapped addressing provides the convenience of easy access to memory-mapped registers located on page zero of data memory. The flexibility of memory-mapped addressing results because accesses are made independently of actual DP value and without having to provide a complete address of the memory location being accessed. Commonly used on-board registers can be accessed with a simplified addressing scheme.

Circular addressing is the most sophisticated ’C5x addressing mode. This addressing mode allows specified buffers in memory to be accessed sequentially with a pointer that automatically wraps around to the beginning of the buffer when the last location is accessed. A total of two independent circular buffers can be allocated at any given time.

Five dedicated registers are allocated for implementation of circular addressing: a beginning-of-buffer and an end-of-buffer register for each of the two independent circular buffers and a control register. Additionally, one of the auxiliary registers is used as the pointer into the circular buffer. All registers used in circular addressing must be initialized properly prior to performing any circular buffer access.

The circular-addressing mode allows implementation of circular buffers, which facilitate data structures used in FIR filters, convolution and correlation algorithms, and waveform generators. Having the capability to access circular buffers automatically with no overhead allows these types of data structures to be implemented most efficiently.

repeat feature

The repeat function can be used with instructions such as multiply/accumulates (MAC and MACD), block moves (BLDD and BLPD), I/O transfers (IN/OUT), and table read/writes (TBLR/TBLW). These instructions, although normally multicycle, are pipelined when the repeat feature is used, and they effectively become single-cycle instructions. For example, the table-read instruction may take three or more cycles to execute, but when the instruction is repeated, a table location can be read every cycle.

The repeat counter (RPTC) is a 16-bit register that, when loaded with a number N, causes the next single instruction to be executed N + 1 times. The RPTC register is loaded by either the RPT or the RPTZ instruction, resulting in a maximum of 65,536 executions of a given instruction. RPTC is cleared by reset. The RPTZ instruction clears both ACC and PREG before the next instruction starts repeating. Once a repeat instruction (RPT or RPTZ) is decoded, all interrupts including $\overline{\text{NMI}}$ (except reset) are masked until the completion of the repeat loop. However, the device responds to the $\overline{\text{HOLD}}$ signal while executing an RPT/RPTZ loop.

repeat feature (continued)

The ’C5x implements a block-repeat feature that provides zero-overhead looping for implementation of FOR and DO loops. The function is controlled by three registers (PASR, PAER, and BRCR) and the BRAF bit in the PMST register. The block-repeat counter register (BRCR) is loaded with a loop count of 0 to 65,535. Then, execution of the RPTB (repeat block) instruction loads the program-address-start register (PASR) with the address of the instruction following the RPTB instruction and loads the program-address-end register (PAER)

with its long-immediate operand. The long-immediate operand is the address of the instruction following the last instruction in the loop minus one. (The repeat block must contain at least three instruction words.) Execution of the RPTB instruction automatically sets active the BRAF bit. With each PC update, the PAER contents are compared to the PC. If they are equal, the BRCR contents are compared to zero. If the BRCR contents are greater than zero, BRCR is decremented and the PASR is loaded into the PC, repeating the loop. If not, the BRAF bit is set low and the processor resumes execution past the end of the code's loop.

The equivalent of a WHILE loop can be implemented by setting the BRAF bit to zero if the exit condition is met. The program then completes the current pass through the loop but does not go back to the top. To exit, the bit must be reset at least four instruction words before the end of the loop. It is possible to exit block-repeat loops and return to them without stopping and restarting the loop. Branches, calls, and interrupts do not necessarily affect the loop. When program control is returned to the loop, loop execution is resumed.

instruction set summary

This section summarizes the operational codes (opcodes) of the instruction set for the 'C5x digital signal processors. The instruction set is a super set of the 'C1x and 'C2x instruction sets. The instructions are arranged according to function and are alphabetized by mnemonic within each category. The symbols in Table 6 are used in the instruction set opcode table (Table 7). The Texas Instruments 'C5x assembler accepts 'C2x instructions as well as 'C5x instructions.

The number of words that an instruction occupies in program memory is specified in column 4 of Table 7. In these cases, different forms of the instruction occupy a different number of words. For example, the ADD instruction occupies one word when the operand is a short immediate value or two words if the operand is a long immediate value.

The number of cycles that an instruction requires to execute is listed in column 5 of Table 7. All instructions are assumed to be executed from internal program memory and internal data dual-access memory. The cycle timings are for single-instruction execution, not for repeat mode.

A read or write access to any peripheral memory-mapped register in data memory locations 20h–4Fh adds one cycle to the cycle time shown because all peripherals perform these accesses over the internal peripheral bus.

instruction set summary (continued)

Table 6. Opcode Symbols

SYMBOL	DESCRIPTION
A	Address
ACC	Accumulator
ACCB	Accumulator buffer
ARX	Auxiliary register value (0–7)
BITX	4-bit field specifies which bit to test for the BIT instruction
BMAR	Block-move address register
DBMR	Dynamic bit-manipulation register
I	Addressing-mode bit
II...II	Immediate operand value
INTM	Interrupt-mode flag bit
INTR#	Interrupt vector number
N	Field for the XC instruction, indicating the number of instructions (one or two) to execute conditionally
PREG	Product register
PROG	Program memory
RPTC	Repeat counter
SHF, SHFT	3/4 bit shift value
TC	Test-control bit
T P	Two bits used by the conditional execution instructions to represent the conditions TC, NTC, and $\overline{\text{BIO}}$ T P Meaning 0 0 $\overline{\text{BIO}}$ low 0 1 TC = 1 1 0 TC = 0 1 1 None of the above conditions
TREGn	Temporary register n (n = 0, 1, or 2)
Z L V C	4-bit field representing the following conditions: Z: ACC = 0 L: ACC < 0 V: Overflow C: Carry A conditional instruction contains two of these 4-bit fields. The 4-LSB field of the instruction is a 4-bit mask field. A 1 in the corresponding mask bit indicates that the condition is being tested. The second 4-bit field (bits 4–7) indicates the state of the conditions designated by the mask bits as being tested. For example, to test for ACC ≥ 0, the Z and L fields are set while the V and C fields are not set. The next 4-bit field contains the state of the conditions to test. The Z field is set to indicate testing the condition ACC = 0, and the L field is reset to indicate testing the condition ACC ≥ 0. The conditions possible with these 8 bits are shown in the BCND, CC, and XC instructions. To determine if the conditions are met, the 4-LSB bit mask is ANDed with the conditions. If any bits are set, the conditions are met.

instruction set summary (continued)

Table 7. TMS320C5x Instruction Set Opcodes

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Absolute value of ACC	ABS	1011 1110 0000 0000	1	1
Add ACCB to ACC with carry	ADCB	1011 1110 0001 0001	1	1
Add to ACC with shift	ADD	0010 SHFT IAAA AAAA	1	1
Add to low ACC short immediate	ADD	1011 1000 IIII IIII	1	1
Add to ACC long immediate with shift	ADD	1011 1111 1001 SHFT	2	2
Add to ACC with shift of 16	ADD	0110 0001 IAAA AAAA	1	1
Add ACCB to ACC	ADDB	1011 1110 0001 0000	1	1
Add to ACC with carry	ADDC	0110 0000 IAAA AAAA	1	1
Add to low ACC with sign extension suppressed	ADDS	0110 0010 IAAA AAAA	1	1
Add to ACC with shift specified by TREG1 [3–0]	ADDT	0110 0011 IAAA AAAA	1	1
AND ACC with data value	AND	0110 1110 IAAA AAAA	1	1
AND with ACC long immediate with shift	AND	1011 1111 1011 SHFT	2	2
AND with ACC long immediate with shift of 16	AND	1011 1110 1000 0001	2	2
AND ACCB with ACC	ANDB	1011 1110 0001 0010	1	1
Barrel shift ACC right	BSAR	1011 1111 1110 SHFT	1	1
Complement ACC	CMPL	1011 1110 0000 0001	1	1
Store ACC in ACCB if ACC > ACCB	CRGT	1011 1110 0001 1011	1	1
Store ACC in ACCB if ACC< ACCB	CRLT	1011 1110 0001 1100	1	1
Exchange ACCB with ACC	EXAR	1011 1110 0001 1101	1	1
Load ACC with ACCB	LACB	1011 1110 0001 1111	1	1
Load ACC with shift	LACC	0001 SHFT IAAA AAAA	1	1
Load ACC long immediate with shift	LACC	1011 1111 1000 SHFT	2	2
Load ACC with shift of 16	LACC	0110 1010 IAAA AAAA	1	1
Load low word of ACC with immediate	LACL	1011 1001 IIII IIII	1	1
Load low word of ACC	LACL	0110 1001 IAAA AAAA	1	1
Load ACC with shift specified by TREG1 [3–0]	LACT	0110 1011 IAAA AAAA	1	1
Load ACCL with memory-mapped register	LAMM	0000 1000 IAAA AAAA	1	1 or 2
Negate ACC	NEG	1011 1110 0000 0010	1	1
Normalize ACC	NORM	1010 0000 IAAA AAAA	1	1
OR ACC with data value	OR	0110 1101 IAAA AAAA	1	1
OR with ACC long immediate with shift	OR	1011 1111 1100 SHFT	2	2
OR with ACC long immediate with shift of 16	OR	1011 1110 1000 0010	2	2
OR ACCB with ACC	ORB	1011 1110 0001 0011	1	1
Rotate ACC 1 bit left	ROL	1011 1110 0000 1100	1	1
Rotate ACCB and ACC left	ROLB	1011 1110 0001 0100	1	1
Rotate ACC 1 bit right	ROR	1011 1110 0000 1101	1	1
Rotate ACCB and ACC right	RORB	1011 1110 0001 0101	1	1
Store ACC in ACCB	SACB	1011 1110 0001 1110	1	1
Store high ACC with shift	SACH	1001 1SHF IAAA AAAA	1	1
Store low ACC with shift	SACL	1001 0SHF IAAA AAAA	1	1
Store ACCL to memory-mapped register	SAMM	1000 1000 IAAA AAAA	1	1 or 2
Shift ACC 16 bits right if TREG1 [4] = 0	SATH	1011 1110 0101 1010	1	1
Shift ACC0–ACC15 right as specified by TREG1 [3–0]	SATL	1011 1110 0101 1011	1	1
Subtract ACCB from ACC	SBB	1011 1110 0001 1000	1	1
Subtract ACCB from ACC with borrow	SBBB	1011 1110 0001 1001	1	1
Shift ACC 1 bit left	SFL	1011 1110 0000 1001	1	1
Shift ACCB and ACC left	SFLB	1011 1110 0001 0110	1	1
Shift ACC 1 bit right	SFR	1011 1110 0000 1010	1	1
Shift ACCB and ACC right	SFRB	1011 1110 0001 0111	1	1
Subtract from ACC with shift	SUB	0011 SHFT IAAA AAAA	1	1
Subtract from ACC with shift of 16	SUB	0110 0101 IAAA AAAA	1	1
Subtract from ACC short immediate	SUB	1011 1010 IIII IIII	1	1
Subtract from ACC long immediate with shift	SUB	1011 1111 1010 SHFT	2	2

instruction set summary (continued)

Table 7. TMS320C5x Instruction Set Opcodes (Continued)

ACCUMULATOR MEMORY REFERENCE INSTRUCTIONS (CONTINUED)				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Subtract from ACC with borrow	SUBB	0110 0100 IAAA AAAA	1	1
Conditional subtract	SUBC	0000 1010 IAAA AAAA	1	1
Subtract from ACC with sign extension suppressed	SUBS	0110 0110 IAAA AAAA	1	1
Subtract from ACC, shift specified by TREG1 [3–0]	SUBT	0110 0111 IAAA AAAA	1	1
XOR ACC with data value	XOR	0110 1100 IAAA AAAA	1	1
XOR with ACC long immediate with shift	XOR	1011 1111 1101 SHFT	2	2
XOR with ACC long immediate with shift of 16	XOR	1011 1110 1000 0011	2	2
XOR ACCB with ACC	XORB	1011 1110 0001 1010	1	1
Zero ACC, load high ACC with rounding	ZALR	0110 1000 IAAA AAAA	1	1
Zero ACC and product register	ZAP	1011 1110 0101 1001	1	1
AUXILIARY REGISTERS AND DATA PAGE POINTER INSTRUCTIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Add to AR short immediate	ADRK	0111 1000 IIII IIII	1	1
Compare AR with CMPR	CMPR	1011 1111 0100 01CM	1	1
Load AR from addressed data	LAR	0000 0ARX IAAA AAAA	1	1
Load AR short immediate	LAR	1011 0ARX IIII IIII	1	1
Load AR long immediate	LAR	1011 1111 0000 1ARX	2	2
Load data page pointer with addressed data	LDP	0000 1101 IAAA AAAA	1	2
Load data page immediate	LDP	1011 110I IIII IIII	1	2
Modify auxiliary register	MAR	1000 1011 IAAA AAAA	1	1
Store AR to addressed data	SAR	1000 0ARX IAAA AAAA	1	1
Subtract from AR short immediate	SBRK	0111 1100 IIII IIII	1	1
BRANCH INSTRUCTIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Branch unconditional with AR update	B	0111 1001 1AAA AAAA	2	4
Branch addressed by ACC	BACC	1011 1110 0010 0000	1	4
Branch addressed by ACC delayed	BACCD	1011 1110 0010 0001	1	2
Branch AR ≠ 0 with AR update	BANZ	0111 1011 1AAA AAAA	2	2 or 4
Branch AR ≠ 0 with AR update delayed	BANZD	0111 1111 1AAA AAAA	2	2
Branch conditional	BCND	1110 00TP ZLVC ZLVC	2	2 or 4
Branch conditional delayed	BCNDD	1111 00TP ZLVC ZLVC	2	2
Branch unconditional with AR update delayed	BD	0111 1101 1AAA AAAA	2	2
Call subroutine addressed by ACC	CALA	1011 1110 0011 0000	1	4
Call subroutine addressed by ACC delayed	CALAD	1011 1110 0011 1101	1	2
Call unconditional with AR update	CALL	0111 1010 1AAA AAAA	2	4
Call unconditional with AR update delayed	CALLD	0111 1110 1AAA AAAA	2	2
Call conditional	CC	1110 10TP ZLVC ZLVC	2	2 or 4
Call conditional delayed	CCD	1111 10TP ZLVC ZLVC	2	2
Software interrupt	INTR	1011 1110 011I NTR#	1	4
Nonmaskable interrupt	NMI	1011 1110 0101 0010	1	4
Return	RET	1110 1111 0000 0000	1	4
Return conditional	RETC	1110 11TP ZLVC ZLVC	1	2 or 4
Return conditionally, delayed	RETCD	1111 11TP ZLVC ZLVC	1	2
Return, delayed	RETD	1111 1111 0000 0000	1	2
Return from interrupt with enable	RETE	1011 1110 0011 1010	1	4
Return from interrupt	RETI	1011 1110 0011 1000	1	4
Trap	TRAP	1011 1110 0101 0001	1	4
Execute next one or two INST on condition	XC	111N 01TP ZLVC ZLVC	1	1

instruction set summary (continued)

Table 7. TMS320C5x Instruction Set Opcodes (Continued)

I/O AND DATA MEMORY OPERATIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Block move from data to data memory	BLDD	1010 1000 IAAA AAAA	2	3
Block move data to data DEST long immediate	BLDD	1010 1001 IAAA AAAA	2	3
Block move data to data with source in BMAR	BLDD	1010 1100 IAAA AAAA	1	2
Block move data to data with DEST in BMAR	BLDD	1010 1101 IAAA AAAA	1	2
Block move data to PROG with DEST in BMAR	BLDP	0101 0111 IAAA AAAA	1	2
Block move from program to data memory	BLPD	1010 0101 IAAA AAAA	2	3
Block move PROG to data with source in BMAR	BLPD	1010 0100 IAAA AAAA	1	2
Data move in data memory	DMOV	0111 0111 IAAA AAAA	1	1
Input external access	IN	1010 1111 IAAA AAAA	2	2
Load memory-mapped register	LMMR	1000 1001 IAAA AAAA	2	2 or 3
Out external access	OUT	0000 1100 IAAA AAAA	2	3
Store memory-mapped register	SMMR	0000 1001 IAAA AAAA	2	2 or 3
Table read	TBLR	1010 0110 IAAA AAAA	1	3
Table write	TBLW	1010 0111 IAAA AAAA	1	3
PARALLEL LOGIC UNIT INSTRUCTIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
AND DBMR with data value	APL	0101 1010 IAAA AAAA	1	1
AND long immediate with data value	APL	0101 1110 IAAA AAAA	2	2
Compare DBMR to data value	CPL	0101 1011 IAAA AAAA	1	1
Compare data with long immediate	CPL	0101 1111 IAAA AAAA	2	2
OR DBMR to data value	OPL	0101 1001 IAAA AAAA	1	1
OR long immediate with data value	OPL	0101 1101 IAAA AAAA	2	2
Store long immediate to data	SPLK	1010 1110 IAAA AAAA	2	2
XOR DBMR to data value	XPL	0101 1000 IAAA AAAA	1	1
XOR long immediate with data value	XPL	0101 1100 IAAA AAAA	2	2
T REGISTER, P REGISTER, AND MULTIPLY INSTRUCTIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Add PREG to ACC	APAC	1011 1110 0000 0100	1	1
Load high PREG	LPH	0111 0101 IAAA AAAA	1	1
Load TREG0	LT	0111 0011 IAAA AAAA	1	1
Load TREG0 and accumulate previous product	LTA	0111 0000 IAAA AAAA	1	1
Load TREG0, accumulate previous product, and move data	LTD	0111 0010 IAAA AAAA	1 1	1 1
Load TREG0 and load ACC with PREG	LTP	0111 0001 IAAA AAAA	1	1
Load TREG0 and subtract previous product	LTS	0111 0100 IAAA AAAA	1	1
Multiply/accumulate	MAC	1010 0010 IAAA AAAA	2	3
Multiply/accumulate with data shift	MACD	1010 0011 IAAA AAAA	2	3
Mult/ACC w/source ADRS in BMAR and DMOV	MADD	1010 1011 IAAA AAAA	1	3
Mult/ACC with source address in BMAR	MADS	1010 1010 IAAA AAAA	1	3
Multiply data value times TREG0	MPY	0101 0100 IAAA AAAA	1	1
Multiply TREG0 by 13-bit immediate	MPY	110I IIII IIII IIII	1	1
Multiply TREG0 by long immediate	MPY	1011 1110 1000 0000	2	2
Multiply TREG0 by data, add previous product	MPYA	0101 0000 IAAA AAAA	1	1
Multiply TREG0 by data, ACC – PREG	MPYS	0101 0001 IAAA AAAA	1	1
Multiply unsigned data value times TREG0	MPYU	0101 0101 IAAA AAAA	1	1
Load ACC with product register	PAC	1011 1110 0000 0011	1	1
Subtract product from ACC	SPAC	1011 1110 0000 0101	1	1
Store high product register	SPH	1000 1101 IAAA AAAA	1	1
Store low product register	SPL	1000 1100 IAAA AAAA	1	1
Set PREG shift count	SPM	1011 1111 0000 00PM	1	1
Data to TREG0, square it, add PREG to ACC	SQRA	0101 0010 IAAA AAAA	1	1
Data to TREG0, square it, ACC – PREG	SQRS	0101 0011 IAAA AAAA	1	1
Zero product register	ZPR	1011 1110 0101 1000	1	1

instruction set summary (continued)

Table 7. TMS320C5x Instruction Set Opcodes (Continued)

CONTROL INSTRUCTIONS				
INSTRUCTION	**MNEMONIC**	**OPCODE**	**WORDS**	**CYCLES**
Test bit specified immediate	BIT	0100 BITX IAAA AAAA	1	1
Test bit in data value as specified by TREG2 [3–0]	BITT	0110 1111 IAAA AAAA	1	1
Reset overflow mode	CLRC	1011 1110 0100 0010	1	1
Reset sign extension mode	CLRC	1011 1110 0100 0110	1	1
Reset hold mode	CLRC	1011 1110 0100 1000	1	1
Reset TC bit	CLRC	1011 1110 0100 1010	1	1
Reset carry	CLRC	1011 1110 0100 1110	1	1
Reset CNF bit	CLRC	1011 1110 0100 0100	1	1
Reset INTM bit	CLRC	1011 1110 0100 0000	1	1
Reset XF pin	CLRC	1011 1110 0100 1100	1	1
Idle	IDLE	1011 1110 0010 0010	1	1
Idle until interrupt — low-power mode	IDLE2	1011 1110 0010 0011	1	1
Load status register 0	LST	0000 1110 IAAA AAAA	1	2
Load status register 1	LST	0000 1111 IAAA AAAA	1	2
No operation	NOP	1000 1011 0000 0000	1	1
Pop PC stack to low ACC	POP	1011 1110 0011 0010	1	1
Pop stack to data memory	POPD	1000 1010 IAAA AAAA	1	1
Push data memory value onto PC stack	PSHD	0111 0110 IAAA AAAA	1	1
Push low ACC to PC stack	PUSH	1011 1110 0011 1100	1	1
Repeat instruction as specified by data	RPT	0000 1011 IAAA AAAA	1	2
Repeat next INST specified by long immediate	RPT	1011 1110 1100 0100	2	2
Repeat INST specified by short immediate	RPT	1011 1011 IIII IIII	1	2
Block repeat	RPTB	1011 1110 1100 0110	2	2
Clear ACC/PREG and repeat next INST long immediate	RPTZ	1011 1110 1100 0101	2	2
Set overflow mode	SETC	1011 1110 0100 0011	1	1
Set sign extension mode	SETC	1011 1110 0100 0111	1	1
Set hold mode	SETC	1011 1110 0100 1001	1	1
Set TC bit	SETC	1011 1110 0100 1011	1	1
Set carry	SETC	1011 1110 0100 1111	1	1
Set XF pin high	SETC	1011 1110 0100 1101	1	1
Set CNF bit	SETC	1011 1110 0100 0101	1	1
Set INTM bit	SETC	1011 1110 0100 0001	1	1
Store status register 0	SST	1000 1110 IAAA AAAA	1	1
Store status register 1	SST	1000 1111 IAAA AAAA	1	1

development support

Texas Instruments offers an extensive line of development tools for the ’C5x generation of DSPs, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules.

The following products support development of ’C5x-based applications:

Software Development Tools:
Assembler/Linker
Simulator
Optimizing ANSI C compiler
Application algorithms
C/Assembly debugger and code profiler

development support (continued)

Hardware Development Tools:
Extended development set (XDS™) emulator (supports ’C5x multiprocessor system debug)
’C5x EVM (Evaluation Module)
’C5x DSK (DSP Starter Kit)

The *TMS320 Family Development Support Reference Guide* (SPRU011) contains information about development support products for all TMS320 family member devices, including documentation. Refer to this document for further information about TMS320 documentation or any other TMS320 support products from Texas Instruments. There is an additional document, the *TMS320 Third Party Support Reference Guide* (SPRU052), which contains information about TMS320-related products from other companies in the industry. To receive copies of TMS320 literature, contact the Literature Response Center at 800/477-8924.

See Table 8 for complete listings of development support tools for the ’C5x. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

Table 8. TMS320C5x, TMS320LC5x Development Support Tools

DEVELOPMENT TOOL	PLATFORM	PART NUMBER
Software		
Compiler/Assembler/Linker	PC-DOS™, OS/2™	TMDS3242855-02
Compiler/Assembler/Linker	SPARC™, HP™	TMDS3242555-08
Assembler/Linker	PC-DOS, OS/2™	TMDS3242850-02
Simulator	PC-DOS, WIN™	TMDS3245851-02
Simulator	SPARC	TMDS3245551-09
Digital Filter Design Package	PC-DOS	DFDP
Debugger/Emulation Software	PC-DOS, OS/2, WIN	TMDS3240150
Debugger/Emulation Software	SPARC™	TMDS3240650
Hardware		
XDS-510 XL Emulator	PC-DOS, OS/2	TMD000510
XDS-510 WS Emulator	SPARC	TMDS000510WS
EVM Evaluation Module	PC-DOS, WIN	TMDS3260050
DSK DSP Starter Kit	PC-DOS	TMDS3200051

device and development support tool nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all TMS320 devices and support tools. Each TMS320 member has one of three prefixes: TMX, TMP, or TMS. Texas Instruments recommends two of three possible prefix designators for its support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (TMX/TMDX) through fully qualified production devices/tools (TMS/TMDS). This development flow is defined below.

Device development evolutionary flow:

TMX Experimental device that is not necessarily representative of the final device’s electrical specifications

TMP Final silicon die that conforms to the device’s electrical specifications but has not completed quality and reliability verification

PC-DOS and OS/2 are trademarks of International Business Machines Corp.
SPARC is a trademark of SPARC International, Inc.
WIN is a trademark of Microsoft Corporation.
HP is a trademark of Hewlett-Packard Company.
XDS is a trademark of Texas Instruments Incorporated.

device and development support tool nomenclature (continued)

TMS Fully-qualified production device

Support tool development evolutionary flow:

TMDX Development support product that has not yet completed Texas Instruments internal qualification testing.

TMDS Fully qualified development support product

TMX and TMP devices and TMDX development support tools are shipped against the following disclaimer:

“Developmental product is intended for internal evaluation purposes.”

TMS devices and TMDS development support tools have been characterized fully, and the quality and reliability of the device has been demonstrated fully. TI’s standard warranty applies.

Predictions show that prototype devices (TMX or TMP) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, N, FN, or GB) and temperature range (for example, L). Figure 8 provides a legend for reading the complete device name for any TMS320 family member.

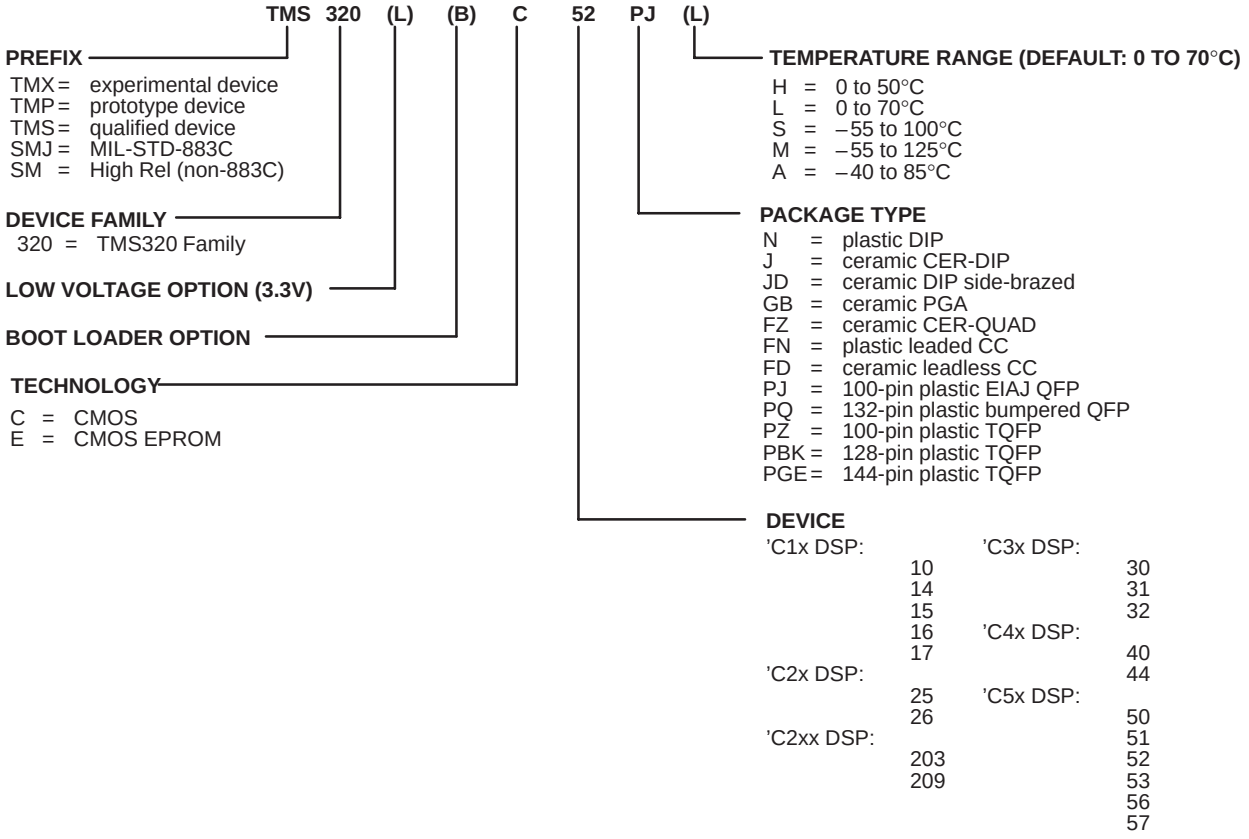


Figure 8. TMS320 Device Nomenclature

documentation support

Extensive documentation supports all TMS320 family generations of devices from product announcement through applications development. The types of documentation available include data sheets, such as this document, with design specifications, complete user's guides for all devices, development support tools, and three volumes of the publication *Digital Signal Processing Applications with the TMS320 Family* (literature numbers SPRA012, SPRA016, and SPRA017).

The application book series describes hardware and software applications, including algorithms, for fixed and floating point TMS320 family devices. The *TMS320C5x User's Guide* (literature number SPRU056), which describes in detail the fifth-generation TMS320 products, is currently available.

A series of DSP textbooks is published by Prentice-Hall and John Wiley & Sons to support digital signal processing research and education. The TMS320 newsletter, *Details on Signal Processing*, is published quarterly and distributed to update TMS320 customers on product information. The TMS320 DSP bulletin board service (BBS) provides access to information pertaining to the TMS320 family, including documentation, source code and object code for many DSP algorithms and utilities. The BBS can be reached at 713/274-2323.

Information regarding TI DSP products is also available on the Worldwide Web at http:/www.ti.com uniform resource locator (URL).

absolute maximum ratings over operating ambient-air temperature range (unless otherwise noted) ('320C5x only)†

Supply voltage range, V_{DD} (see Note 3) . . . . . – 0.3 V to 7 V
Input voltage range, V_I . . . . . – 0.3 V to 7 V
Output voltage range, V_O . . . . . – 0.3 V to 7 V
Operating ambient temperature range, T_A . . . . . –40°C to 85°C
Operating case temperature, T_C . . . . . 0°C to 85°C
Storage temperature range, T_{stg} . . . . . – 55°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 3: All voltage values are with respect to V_{SS}.

recommended operating conditions ('320C5x only)

			MIN	NOM	MAX	UNIT
V_{DD}	Supply voltage		4.75	5	5.25	V
V_{SS}	Supply voltage			0		V
V_{IH}	High-level input voltage	X2/CLKIN, CLKIN2	3		$V_{DD}+0.3$	V
		CLKX, CLKR, TCLKX, TCLKR	2.5		$V_{DD}+0.3$	
		All other inputs	2		$V_{DD}+0.3$	
V_{IL}	Low-level input voltage	X2/CLKIN, CLKIN2, CLKX, CLKR, TCLKX, TCLKR	– 0.3		0.7	V
		All other inputs	– 0.3		0.8	
I_{OH}	High-level output current (see Note 4)				– 300‡	µA
I_{OL}	Low-level output current				2	mA
T_C	Operating case temperature		0		85	°C
T_A	Operating ambient temperature		–40		85	°C

‡ This I_{OH} can be exceeded when using a 1-kΩ pulldown resistor on the TDM serial port TADD output; however, this output still meets V_{OH} specifications under these conditions.

NOTE 4: Figure 9 shows the test load circuit and Figure 10 and Figure 11 show the voltage reference levels.

electrical characteristics over recommended ranges of supply voltage and operating ambient-air temperature (unless otherwise noted) ('320C5x only)

PARAMETER		TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
V_{OH}	High-level output voltage (see Note 4)	I_{OH} = 300 μA	2.4	3		V
V_{OL}	Low-level output voltage (see Note 4)	I_{OL} = 2 mA		0.3	0.6	V
I_{OZ}	High-impedance output current (V_{DD} = 5.25 V)	$\overline{BR}$ (with internal pullup)	– 500		20	μA
		All other 3-state outputs	– 20		20	
I_I	Input current (V_I = V_{SS} to V_{DD})	$\overline{TRST}$ (with internal pulldown)	– 10		800	μA
		TMS, TCK, TDI (with internal pullups)	– 500		10	
		X2/CLKIN	– 50		50	
		All other inputs	– 10		10	
$I_{DD(core)}$	Supply current, core CPU	f_X = 40 MHz, V_{DD} = 5.25 V		60		mA
		f_X = 57 MHz, V_{DD} = 5.25 V		67		
		f_X = 80 MHz, V_{DD} = 5.25 V		94		
		f_X = 100 MHz, V_{DD} = 5.25 V		110		
$I_{DD(pins)}$	Supply current, pins	f_X = 40 MHz, V_{DD} = 5.25 V		40		mA
		f_X = 57 MHz, V_{DD} = 5.25 V		45		
		f_X = 80 MHz, V_{DD} = 5.25 V		63		
		f_X = 100 MHz, V_{DD} = 5.25 V		75		
$I_{DD(standby)}$	Supply current, standby	IDLE2, divide-by-two clock mode, clocks shut off		5		μA
C_i	Input capacitance			15		pF
C_o	Output capacitance			15		pF

† Typical values are at V_{DD} = 5 V, T_A = 25°C, unless otherwise specified.
NOTE 4: Figure 9 shows the test load circuit and Figure 10 and Figure 11 show the voltage reference levels.

absolute maximum ratings over specified temperature range (unless otherwise noted) ('320LC5x only)†

Supply voltage range, V_{DD} (see Note 3) . . . . . –0.3 V to 5 V
Input voltage range, V_I . . . . . –0.3 V to 5 V
Output voltage range, V_O . . . . . –0.3 V to 5 V
Operating ambient temperature range, T_A . . . . . –40° to 85°C
Operating case temperature, T_C . . . . . 0°C to 85°C
Storage temperature range, T_{stg} . . . . . –55° to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 3: All voltage values are with respect to V_{SS}.

recommended operating conditions ('320LC5x only)

			MIN	NOM	MAX	UNIT
V_{DD}	Supply voltage		3.13	3.3	3.47	V
V_{SS}	Supply voltage			0		V
V_{IH}	High-level input voltage	X2/CLKIN, CLKIN2	2.5		V_{DD} + 0.3	V
		CLKX, CLKR, TCLKX, TCLKR	2.0		V_{DD} + 0.3	
		All other inputs	1.8		V_{DD} + 0.3	
V_{IL}	Low-level input voltage	X2/CLKIN, CLKIN2, CLKX, CLKR, TCLKX, TCLKR	–0.3		0.5	V
		All other inputs	–0.3		0.6	V
I_{OH}	High-level output current				– 300‡	µA
I_{OL}	Low-level output current				2	mA
T_C	Operating case temperature		0		85	°C
T_A	Operating ambient temperature		–40		85	°C

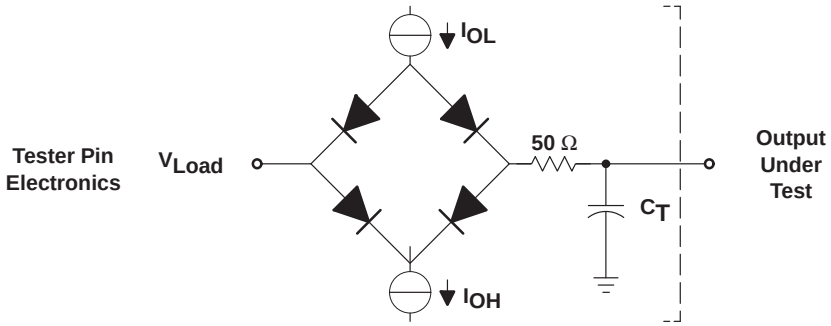
‡ This I_{OH} may be exceeded when using a 1-kΩ pulldown resistor on the TDM serial port TADD output; however, this output still meets V_{OH} specifications under these conditions.

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) ('320LC5x only)

PARAMETER		TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V_{OH}	High-level output voltage (see Note 4)	I_{OH} = 300 µA	2.0			V
		I_{OH} = 20 µA		V_{DD} – 0.3‡		
V_{OL}	Low-level output voltage (see Note 4)	I_{OL} = 2 mA			0.4	V
		I_{OL} = 20 µA		0.3‡		
I_{OZ}	High-impedance output current (V_{DD} = 3.47 V)	$\overline{BR}$ (with internal pullup)	–500		20	µA
		All other 3-state outputs	–20		20	
I_I	Input current (V_I = V_{SS} to V_{DD})	$\overline{TRST}$ (with internal pulldown)	–10		800	µA
		TMS, TCK, TDI pins (with internal pullups)	–500		10	
		X2/CLKIN (oscillator enabled)	–50		50	
		X2/CLKIN (oscillator disabled)	–10		10	
		All other inputs	–10		10	
$I_{DD(core)}$	Supply current, core CPU	f_x = 40 MHz, V_{DD} = 3.47 V		26		mA
		f_x = 50 MHz, V_{DD} = 3.47 V		33		
		f_x = 80 MHz, V_{DD} = 3.47 V		53		
$I_{DD(pins)}$	Supply current, pins	f_x = 40 MHz, V_{DD} = 3.47 V		18		mA
		f_x = 50 MHz, V_{DD} = 3.47 V		22		
		f_x = 80 MHz, V_{DD} = 3.47 V		35		
$I_{DD(standby)}$	Supply current, standby	IDLE2, divide-by-two clock mode, clocks shut off		5		µA
C_i	Input capacitance			15		pF
C_o	Output capacitance			15		pF

† All typical values are at V_{DD} = 3.3 V, T_A = 25°C.
‡ Values derived from characterization data and not tested
NOTE 4: Figure 9 shows the test load circuit and Figure 10 and Figure 11 show the voltage reference levels.

PARAMETER MEASUREMENT INFORMATION



Where:

I_{OL}	=	2 mA (all outputs) minimum
I_{OH}	=	300 µA (all outputs) minimum
V_{LOAD}	=	1.5 V
C_T	=	80-pF typical load circuit capacitance

Figure 9. Test Load Circuit

signal transition levels

The data in this section is shown for both the 5-V version (’C5x) and the 3.3-V version (’LC5x). In each case, the 5-V data is shown followed by the 3.3-V data in parentheses. TTL-output levels are driven to a minimum logic-high level of 2.4 V (2 V) and to a maximum logic-low level of 0.6 V (0.4 V). Figure 10 shows the TTL-level outputs.

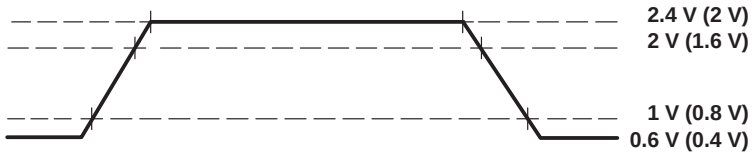


Figure 10. TTL-Level Outputs

TTL-output transition times are specified as follows:

- For a *high-to-low transition*, the level at which the output is said to be no longer high is 2 V (1.6 V), and the level at which the output is said to be low is 1 V (0.8 V).
- For a *low-to-high transition*, the level at which the output is said to be no longer low is 1 V (0.8 V), and the level at which the output is said to be high is 2 V (1.6 V).

Figure 11 shows the TTL-level inputs.



Figure 11. TTL-Level Inputs

TTL-compatible input transition times are specified as follows:

- For a *high-to-low transition* on an input signal, the level at which the input is said to be no longer high is 2 V (1.8 V), and the level at which the input is said to be low is 0.8 V (0.6 V).
- For a *low-to-high transition* on an input signal, the level at which the input is said to be no longer low is 0.8 V (0.6 V), and the level at which the input is said to be high is 2 V (1.8 V).

PARAMETER MEASUREMENT INFORMATION

timing parameter symbology

Timing parameter symbols used are created in accordance with JEDEC Standard 100-A. To shorten the symbols, some of the pin names and other related terminology have been abbreviated as follows:

Lowercase subscripts and their meanings:

a	access time
c	cycle time (period)
d	delay time
dis	disable time
en	enable time
f	fall time
h	hold time
r	rise time
su	setup time
t	transition time
v	valid time
w	pulse duration (width)
X	Unknown, changing, or don't care level

Letters and symbols and their meanings:

H	High
L	Low
V	Valid
Z	High impedance

CLOCK CHARACTERISTICS AND TIMING

The ’C5x can use either its internal oscillator or an external frequency source for a clock. The clock mode is determined by the clock mode pins (CLKMD1, CLKMD2, and CLKMD3). Table 9 shows the standard clock options available on the ’C50, ’LC50, ’C51, ’LC51, ’C52, ’LC52, ’C53, ’LC53, ’C53S, and ’LC53S. For these devices, the CLKIN2 pin functions as the external frequency input when using the PLL options. An expanded set of clock options is shown in Table 10 and is available on the ’LC56, ’C57S, and ’LC57 devices. For these devices, X2/CLKIN functions as the external frequency input when using the PLL options.

Table 9. Standard Clock Options

CLKMD1	CLKMD2	CLOCK SOURCE
1	0	PLL clock generator option†
0	1	Reserved for test purposes
1	1	External divide-by-two option or internal divide-by-two clock option with an external crystal
0	0	External divide-by-two option with the internal oscillator disabled

† PLL multiply-by-one option on ’C50, ’C51, ’C53, ’C53S devices, PLL multiply-by-two option on ’C52 device

Table 10. PLL Clock Option for ’LC56, ’C57S, and ’LC57

CLKMD1	CLKMD2	CLKMD3	CLOCK SOURCE
0	0	0	PLL multiply-by-three
0	1	0	PLL multiply-by-four
1	0	0	PLL multiply-by-five
1	1	0	PLL multiply-by-nine
0	0	1	External divide-by-two option with oscillator disabled
0	1	1	PLL multiply-by-two
1	0	1	PLL multiply-by-one
1	1	1	External/Internal divide-by-two with oscillator enabled

internal divide-by-two clock option with external crystal

The internal oscillator is enabled by connecting a crystal across X1 and X2/CLKIN. The frequency of CLKOUT1 is one-half of the crystal's oscillating frequency. The crystal should be in either fundamental or overtone operation and parallel resonant, with an effective series resistance of 30 Ω and a power dissipation of 1 mW; it should be specified at a load capacitance of 20 pF. Overtone crystals require an additional tuned-LC circuit. Figure 12 shows an external crystal (fundamental frequency) connected to the on-chip oscillator.

recommended operating conditions for internal divide-by-two clock option

			MIN	NOM	MAX	UNIT
f_{clk}	Input clock frequency	TMS320C5x-40	0†		40.96	MHz
		TMS320C5x-57	0†		57.14	
		TMS320C5x-80	0†		80	
		TMS320C5x-100‡	0†		100	
		TMS320LC5x-40	0†		40	MHz
		TMS320LC5x-50	0†		50	
		TMS320LC5x-80	0†		80	
C1, C2	Load capacitance			10		pF

† This device utilizes a fully static design and, therefore, can operate with input clock cycle time ($t_{c(CI)}$) approaching ∞. The device is characterized at frequencies approaching 0 Hz, but is tested at f_{clk} = 6.7 MHz to meet device test time requirements.
‡ '320C51, '320C52 currently available at this clock speed

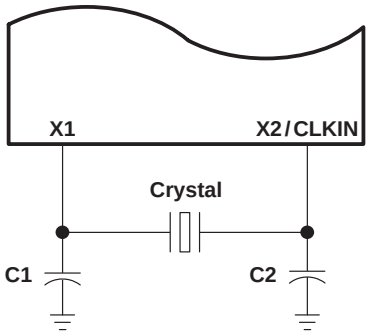


Figure 12. Internal Clock Option

external divide-by-two clock option

An external frequency source can be used by injecting the frequency directly into X2/CLKIN with X1 left unconnected. Refer to Table 9 and Table 10 for appropriate configuration of the CLKMD1, CLKMD2 and CLKMD3 pins to generate the external divide-by-2 clock option. The external frequency injected must conform to the specifications listed in the timing requirements table.

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$] ('320C5x only) (see Figure 13)

PARAMETER		'320C5x-40			'320C5x-57			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	48.8	$2t_{c(CI)}$	†	35	$2t_{c(CI)}$	†	ns
$t_{d(CIH-COH/L)}$	Delay time, X2/CLKIN high to CLKOUT1 high/low	3	11	20	3	11	20	ns
$t_{f(CO)}$	Fall time, CLKOUT1		5			5		ns
$t_{r(CO)}$	Rise time, CLKOUT1		5			5		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 3	H	H + 2	H – 3	H	H + 2	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3	H	H + 2	H – 3	H	H + 2	ns

PARAMETER		'320C5x-80			'320C5x-100			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	25	$2t_{c(CI)}$	†	20	$2t_{c(CI)}$	†	ns
$t_{d(CIH-COH/L)}$	Delay time, X2/CLKIN high to CLKOUT1 high/low	1	9	18	1	9	18	ns
$t_{f(CO)}$	Fall time, CLKOUT1		4			4		ns
$t_{r(CO)}$	Rise time, CLKOUT1		4			4		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 3	H	H + 2	H – 3	H	H + 2	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3	H	H + 2	H – 3	H	H + 2	ns

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$] ('320LC5x only) (see Figure 13)

PARAMETER		'320LC5x-40			'320LC5x-50			'320LC5x-80			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	UNIT
$t_{c(CO)}$	Cycle time, CLKOUT1	50	$2t_{c(CI)}$	†	40	$2t_{c(CI)}$	†	25	$2t_{c(CI)}$	†	ns
$t_{d(CIH-COH/L)}$	Delay time, X2/CLKIN high to CLKOUT1 high/low	3	11	20	3	11	20	1	9	18	ns
$t_{f(CO)}$	Fall time, CLKOUT1		5			5			4		ns
$t_{r(CO)}$	Rise time, CLKOUT1		5			5			4		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 3	H	H + 2	H – 3	H	H + 2	H – 3	H	H + 2	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3	H	H + 2	H – 3	H	H + 2	H – 3	H	H + 2	ns

† This device utilizes a fully static design and, therefore, can operate with $t_{c(CI)}$ approaching infinity. The device is characterized at frequencies approaching 0 Hz but is tested at $t_{c(CO)}$ = 300 ns to meet device test time requirements.

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature (’320C5x only) (see Figure 13)

		’320C5x-40		’320C5x-57		’320C5x-80		’320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{c(CI)}$	Cycle time, X2/CLKIN	24.4	†	17.5	†	12.5	†	10	†	ns
$t_{f(CI)}$	Fall time, X2/CLKIN‡		5		5		4		4	ns
$t_{r(CI)}$	Rise time, X2/CLKIN‡		5		5		4		4	ns
$t_{w(CIL)}$	Pulse duration, X2/CLKIN low	11	†	8	†	5	†	5	†	ns
$t_{w(CIH)}$	Pulse duration, X2/CLKIN high	11	†	8	†	5	†	5	†	ns

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature (’320LC5x only) (see Figure 13)

		’320LC5x-40		’320LC5x-50		’320LC5x-80		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{c(CI)}$	Cycle time, X2/CLKIN	25	†	20	†	12.5	†	ns
$t_{f(CI)}$	Fall time, X2/CLKIN‡		5		5		4	ns
$t_{r(CI)}$	Rise time, X2/CLKIN‡		5		5		4	ns
$t_{w(CIL)}$	Pulse duration, X2/CLKIN low	11	†	9	†	5	†	ns
$t_{w(CIH)}$	Pulse duration, X2/CLKIN high	11	†	9	†	5	†	ns

† This device utilizes a fully static design and, therefore, can operate with $t_{c(CI)}$ approaching ∞. The device is characterized at frequencies approaching 0 Hz, but is tested at a minimum of $t_{c(CI)}$ = 150 ns to meet device test time requirements.
‡ Values derived from characterization data and not tested

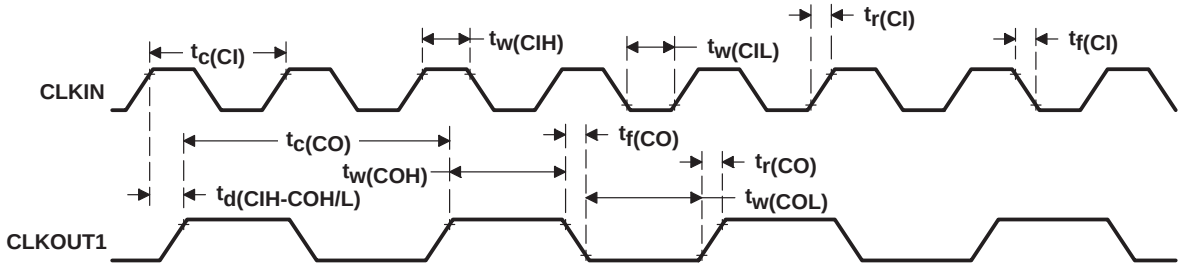


Figure 13. External Divide-by-Two Clock Timing

PLL clock generator option

An external frequency source can be used by injecting the frequency directly into CLKIN2‡ with X1 left unconnected and X2 connected to V_{DD}. This external frequency is multiplied by the factors shown in Table 9 and Table 10 to generate the internal machine cycle. The multiply-by-one option is available on the ’C50, ’LC50, ’C51, ’LC51, ’C53, ’LC53, ’C53S and ’LC53S. The multiply-by-two option is available on the ’C52 and ’LC52. Multiplication factors of 1, 2, 3, 4, 5, and 9 are available on the ’LC56, ’LC57, ’C57S and ’LC57S. Refer to Table 9 and Table 10 for appropriate configuration of the CLKMD1, CLKMD2 and CLKMD3 pins to generate the desired PLL multiplication factor. The external frequency injected must conform to the specifications listed in the timing requirements table.

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$] (’320C5x only) (see Figure 14)

PARAMETER		’320C5x-40			’320C5x-57			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	48.8		75	35		75	ns
$t_{f(CO)}$	Fall time, CLKOUT1		5			5		ns
$t_{r(CO)}$	Rise time, CLKOUT1		5			5		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 3†	H	H + 2†	H – 3†	H	H + 2†	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3†	H	H + 2†	H – 3†	H	H + 2†	ns
$t_{d(C2H-COH)}$	Delay time, CLKIN2 high to CLKOUT1 high	2	9	16	2	9	16	ns
$t_{d(TP)}$	Delay time, transitory phase—PLL synchronized after CLKIN2 supplied†			$1000t_{c(C2)}$‡			$1000t_{c(C2)}$‡	ns

PARAMETER		’320C5x-80			’320C5x-100			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	25		55	20		45	ns
$t_{f(CO)}$	Fall time, CLKOUT1		4			4		ns
$t_{r(CO)}$	Rise time, CLKOUT1		4			4		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 3†	H	H + 2†	H – 3†	H	H + 2†	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3†	H	H + 2†	H – 3†	H	H + 2†	ns
$t_{d(C2H-COH)}$	Delay time, CLKIN2 high to CLKOUT1 high	1	8	15	1	8	15	ns
$t_{d(TP)}$	Delay time, transitory phase—PLL synchronized after CLKIN2 supplied†			$1000t_{c(C2)}$‡			$1000t_{c(C2)}$‡	ns

† Values assured by design and not tested
‡ On the TMS320C57S devices, CLKIN2 functions as the PLL clock input.

switching characteristics over recommended operating conditions [H = 0.5 $t_{c(CO)}$] ('320LC5x only) (see Figure 14)

PARAMETER		'320LC5x-40			'320LC5x-50			'320LC5x-80			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(CO)}$	Cycle time, CLKOUT1	50		75†	40		75†	25		55†	ns
$t_{d(C2H-COH)}$	Delay time, CLKIN2 high to CLKOUT1 high	2	9	16	2	9	16	1	8	15	ns
$t_{f(CO)}$	Fall time, CLKOUT1		5			5			4		ns
$t_{r(CO)}$	Rise time, CLKOUT1		5			5			4		ns
$t_{w(COL)}$	Pulse duration, CLKOUT1 low	H – 3‡	H	H + 2‡	H – 3‡	H	H + 2‡	H – 3‡	H	H + 2‡	ns
$t_{w(COH)}$	Pulse duration, CLKOUT1 high	H – 3‡	H	H + 2‡	H – 3‡	H	H + 2‡	H – 3‡	H	H + 2‡	ns
$t_{d(TP)}$	Delay time, transitory phase—PLL synchronized after CLKIN2 supplied			$1000t_{c(C2)}$			$1000t_{c(C2)}$			$1000t_{c(C2)}$	ns

† Clocks can only be stopped while executing IDLE2 when using the PLL clock generator option.
‡ Values assured by design and not tested
§ On the 'LC56, 'LC57, and 'LC57S devices, CLKIN2 functions as the PLL clock input.

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature (’320C5x only) (see Figure 14)

			’320C5x-40		’320C5x-57		UNIT
			MIN	MAX	MIN	MAX	
$t_{c(C2)}$	Cycle time, CLKIN2	Multiply-by-one†	48.8	75‡	35	75‡	ns
		Multiply-by-two§	97.6	150‡	70	150‡	ns
$t_{f(C2)}$	Fall time, CLKIN2¶			5		5	ns
$t_{r(C2)}$	Rise time, CLKIN2¶			5		5	ns
$t_{w(C2L)}$	Pulse duration, CLKIN2 low		15	$t_{c(C2)}-15$	11	$t_{c(C2)}-11$	ns
$t_{w(C2H)}$	Pulse duration, CLKIN2 high		15	$t_{c(C2)}-15$	11	$t_{c(C2)}-11$	ns

			’320C5x-80		’320C5x-100		UNIT
			MIN	MAX	MIN	MAX	
$t_{c(C2)}$	Cycle time, CLKIN2	Multiply-by-one†	25	75‡	20	75‡	ns
		Multiply-by-two§	50	150‡	40	110‡	ns
$t_{f(C2)}$	Fall time, CLKIN2¶			4		4	ns
$t_{r(C2)}$	Rise time, CLKIN2¶			4		4	ns
$t_{w(C2L)}$	Pulse duration, CLKIN2 low		8	$t_{c(C2)}-8$	7	$t_{c(C2)}-7$	ns
$t_{w(C2H)}$	Pulse duration, CLKIN2 high		8	$t_{c(C2)}-8$	7	$t_{c(C2)}-7$	ns

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature (’320LC5x only) (see Figure 14)

			’320LC5x-40		’320LC5x-50		’320LC5x-80		UNIT
			MIN	MAX	MIN	MAX	MIN	MAX	
$t_{c(C2)}$	Cycle time, CLKIN2	Multiply-by-one†	50	75‡	40	75‡	25	37.5‡	ns
		Multiply-by-two§	100	150‡	80	150‡	50	110‡	ns
$t_{f(C2)}$	Fall time, CLKIN2¶			5		5		4	ns
$t_{r(C2)}$	Rise time, CLKIN2¶			5		5		4	ns
$t_{w(C2L)}$	Pulse duration, CLKIN2 low		15	$t_{c(C2)}-15$	13	$t_{c(C2)}-13$	8	$t_{c(C2)}-8$	ns
$t_{w(C2H)}$	Pulse duration, CLKIN2 high		15	$t_{c(C2)}-15$	13	$t_{c(C2)}-13$	8	$t_{c(C2)}-8$	ns

† Not available on ’C52, ’LC52

‡ Clocks can be stopped only while executing IDLE2 when using the PLL clock generator option. The $t_{d(TP)}$ (the transitory phase) occurs when restarting clock from IDLE2 in this mode.

§ Available on ’C52, ’LC52, ’LC56, ’C57S, ’LC57, and ’LC57S

¶ Values derived from characterization data and not tested

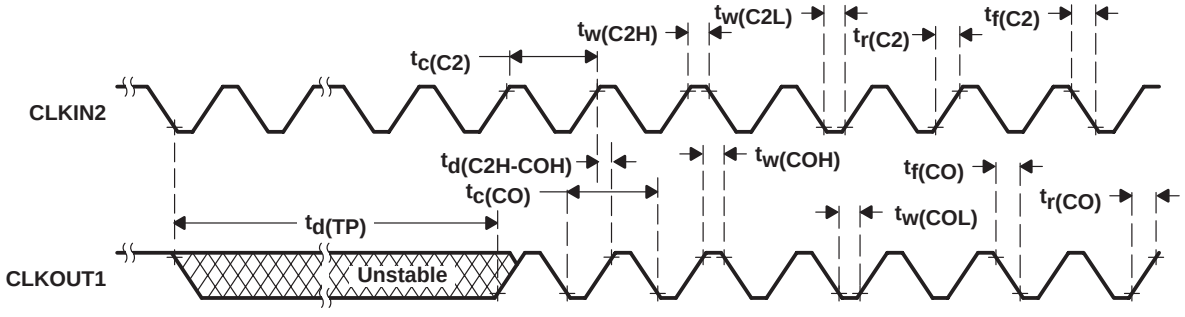


Figure 14. PLL Clock Generator Timing

MEMORY AND PARALLEL I/O INTERFACE READ

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] ('320C5x only) (see Figure 15)

PARAMETER		'320C5x-40		'320C5x-57		'320C5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(AV-RDL)}$	Setup time, address valid before $\overline{RD}$ low†	H – 10‡		H – 10‡		H – 7‡		H – 6‡		ns
$t_{h(RDH-AV)}$	Hold time, address valid after $\overline{RD}$ high†	0‡		0‡		0‡		0‡		ns
$t_{w(RDL)}$	Pulse duration, $\overline{RD}$ low§¶#	H – 2	H + 2	H – 2	H + 2	H – 2	H + 2	H – 2	H + 2	ns
$t_{w(RDH)}$	Pulse duration, $\overline{RD}$ high§¶#	H – 2		H – 2		H – 2		H – 2		ns
$t_{d(CO-ST)}$	Delay time, CLKOUT1 to $\overline{STRB}$ rising or falling edge§¶	– 1	3	– 2	2	– 2	2	– 2	2	ns
$t_{d(CO-RD)}$	Delay time, CLKOUT1 to $\overline{RD}$ rising or falling edge§¶	– 3	1	– 3	1	– 3	1	– 3	1	ns
$t_{d(RDH-WEL)}$	Delay time, $\overline{RD}$ high to $\overline{WE}$ low	2H – 5		2H – 5		2H – 4		2H – 4		ns

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] ('320LC5x only) (see Figure 15)

PARAMETER		'320LC5x-40 '320LC5x-50		'320LC5x-80		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(AV-RDL)}$	Setup time, address valid before $\overline{RD}$ low†	H – 10‡		H – 7‡		ns
$t_{h(RDH-AV)}$	Hold time, address valid after $\overline{RD}$ high†	0‡		0‡		ns
$t_{w(RDL)}$	Pulse duration, $\overline{RD}$ low§¶#	H – 2	H + 2	H – 2	H + 2	ns
$t_{w(RDH)}$	Pulse duration, $\overline{RD}$ high§¶#	H – 2		H – 2		ns
$t_{d(RDH-WEL)}$	Delay time, $\overline{RD}$ high to $\overline{WE}$ low	2H – 5		2H – 4		ns
$t_{d(CO-RD)}$	Delay time, CLKOUT1 to $\overline{RD}$ rising or falling edge§¶	– 2	2	– 3	1	ns
$t_{d(CO-ST)}$	Delay time, CLKOUT1 to $\overline{STRB}$ rising or falling edge§¶	0	4	– 2	2	ns

† A0–A15, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, R/$\overline{W}$, and $\overline{BR}$ timings all are included in timings referenced as address.
‡ See Figure 16 for address bus timing variation with load capacitance.
§ These timings are for the cycles following the first cycle after reset, which is always seven wait states.
¶ Values are derived from characterization data and not tested.
# Timings are valid for zero wait-state cycles only.

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] ('320C5x only) (see Figure 15)

		'320C5x-40		'320C5x-57		'320C5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{a(RDAV)}$	Access time, read data from address valid		2H – 18†		2H – 15†		2H – 10†		2H – 10†	ns
$t_{a(RDL-RD)}$	Access time, read data after $\overline{RD}$ low		H – 10		H – 10		H – 7		H – 6	ns
$t_{su(RD-RDH)}$	Setup time, read data before $\overline{RD}$ high	10		10		7		6		ns
$t_{h(RDH-RD)}$	Hold time, read data after $\overline{RD}$ high	0		0		0		0		ns

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] ('320LC5x only) (see Figure 15)

		'320LC5x-40 '320LC5x-50		'320LC5x-80		UNIT
		MIN	MAX	MIN	MAX	
$t_{a(RDAV)}$	Access time, read data from address valid		2H – 17†		2H – 10†	ns
$t_{su(RD-RDH)}$	Setup time, read data before $\overline{RD}$ high	10		7		ns
$t_{h(RDH-RD)}$	Hold time, read data after $\overline{RD}$ high	0		0		ns
$t_{a(RDL-RD)}$	Access time, read data after $\overline{RD}$ low		H – 10		H – 7	ns

† See Figure 16 for address bus timing variation with load capacitance.

MEMORY AND PARALLEL I/O INTERFACE WRITE

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] ('320C5x only) (see Figure 15)

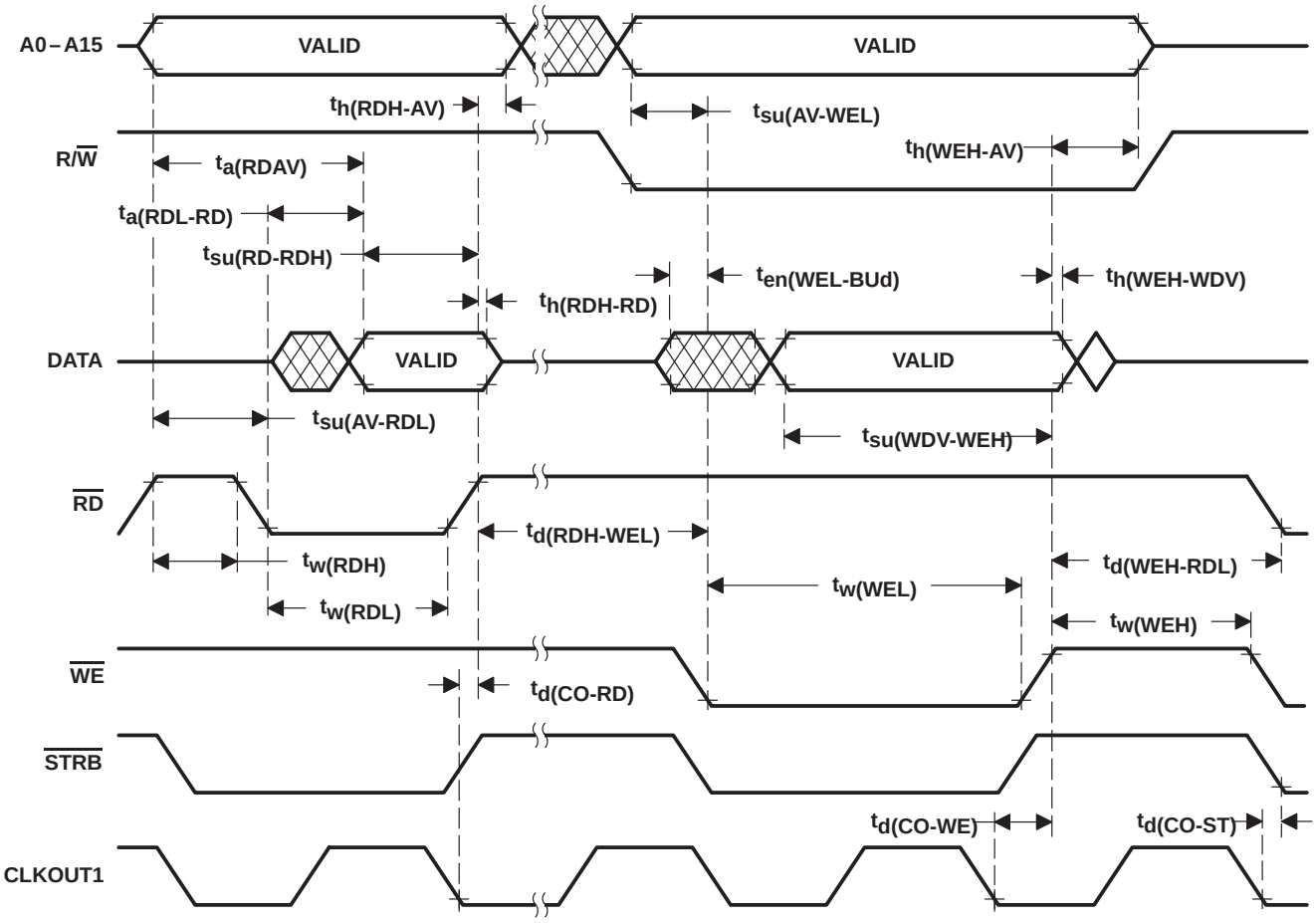
PARAMETER		'320C5x-40		'320C5x-57		'320C5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(AV-WEL)}$	Setup time, address valid before $\overline{WE}$ low†	H – 5‡		H – 5‡		H – 4‡		H – 3‡		ns
$t_{su(WDV-WEH)}$	Setup time, write data valid before $\overline{WE}$ high	2H – 20	2H§¶	2H – 20	2H§¶	2H – 14	2H§¶	2H – 14	2H§¶	ns
$t_{h(WEH-AV)}$	Hold time, address valid after $\overline{WE}$ high†	H – 10‡		H – 10‡		H – 7‡		H – 7‡		ns
$t_{h(WEH-WDV)}$	Hold time, write data valid after $\overline{WE}$ high	H – 5	H + 10§	H – 5	H + 10§	H – 4	H + 7§	H – 4	H + 7§	ns
$t_{w(WEL)}$	Pulse duration, $\overline{WE}$ low§¶	2H – 2	2H + 2§	2H – 2	2H + 2§	2H – 2	2H + 2	2H – 2	2H + 2	ns
$t_{w(WEH)}$	Pulse duration, $\overline{WE}$ high§	2H – 2		2H – 2		2H – 2		2H – 2		ns
$t_{d(CO-ST)}$	Delay time, CLKOUT1 to $\overline{STRB}$ rising or falling edge§	– 1	3	– 2	2	– 2	2	– 2	2	ns
$t_{d(CO-WE)}$	Delay time, CLKOUT1 to $\overline{WE}$ rising or falling edge§	0	4	– 1	3	– 1	3	– 1	3	ns
$t_{d(WEH-RDL)}$	Delay time, $\overline{WE}$ high to $\overline{RD}$ low	3H – 10		3H – 10		3H – 7		3H – 7		ns
$t_{en(WEL-BUd)}$	Enable time, $\overline{WE}$ low to data bus driven	– 5§		– 5§		– 4§		– 4§		ns

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] ('320LC5x only) (see Figure 15)

PARAMETER		'320LC5x-40 '320LC5x-50		'320LC5x-80		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(AV-WEL)}$	Setup time, address valid before $\overline{WE}$ low†	H – 7‡		H – 4‡		ns
$t_{su(WDV-WEH)}$	Setup time, write data valid before $\overline{WE}$ high#	2H – 20	2H§¶	2H – 14	2H§¶	ns
$t_{h(WEH-AV)}$	Hold time, address valid after $\overline{WE}$ high†	H – 10‡		H – 7‡		ns
$t_{h(WEH-WDV)}$	Hold time, write data valid after $\overline{WE}$ high	H – 5	H + 10§	H – 4	H + 7§	ns
$t_{w(WEL)}$	Pulse duration, $\overline{WE}$ low¶§	2H – 4	2H + 2	2H – 4	2H + 2	ns
$t_{w(WEH)}$	Pulse duration, $\overline{WE}$ high¶	2H – 2		2H – 2		ns
$t_{d(WEH-RDL)}$	Delay time, $\overline{WE}$ high to $\overline{RD}$ low	3H – 10		3H – 7		ns
$t_{d(CO-ST)}$	Delay time, CLKOUT1 to $\overline{STRB}$ rising or falling edge¶	0	4	– 2	2	ns
$t_{d(CO-WE)}$	Delay time, CLKOUT1 to $\overline{WE}$ rising or falling edge¶	0	4	– 1	3	ns
$t_{en(WE-BUd)}$	Enable time, $\overline{WE}$ to data bus driven	– 5§		– 4§		ns

† A0–A15, $\overline{PS}$, $\overline{DS}$, $\overline{IS}$, R/$\overline{W}$, and $\overline{BR}$ timings are all included in timings referenced as address.
‡ See Figure 16 for address bus timing variation with load capacitance.
§ Values derived from characterization data and not tested
¶ This value holds true for zero wait states or one software wait state only.
# $\overline{STRB}$ and $\overline{WE}$ edges are 0–4 ns from CLKOUT1 edges on writes. Rising and falling edges of these signals track each other; tolerance of resulting pulsewidths is ±2 ns, not ±4 ns.

MEMORY AND PARALLEL I/O INTERFACE WRITE (CONTINUED)



NOTES: A. All timings are for 0 wait states. However, external writes always require two cycles to prevent external bus conflicts. The diagram illustrates a one-cycle read and a two-cycle write and is not drawn to scale. All external writes immediately preceded by an external read or immediately followed by an external read require three machine cycles.

B. Refer to Appendix B of *TMS320C5x User's Guide* (literature number SPRU056) for logical timings of external interface.

Figure 15. Memory and Parallel I/O Interface Read and Write Timing

MEMORY AND PARALLEL I/O INTERFACE WRITE (CONTINUED)

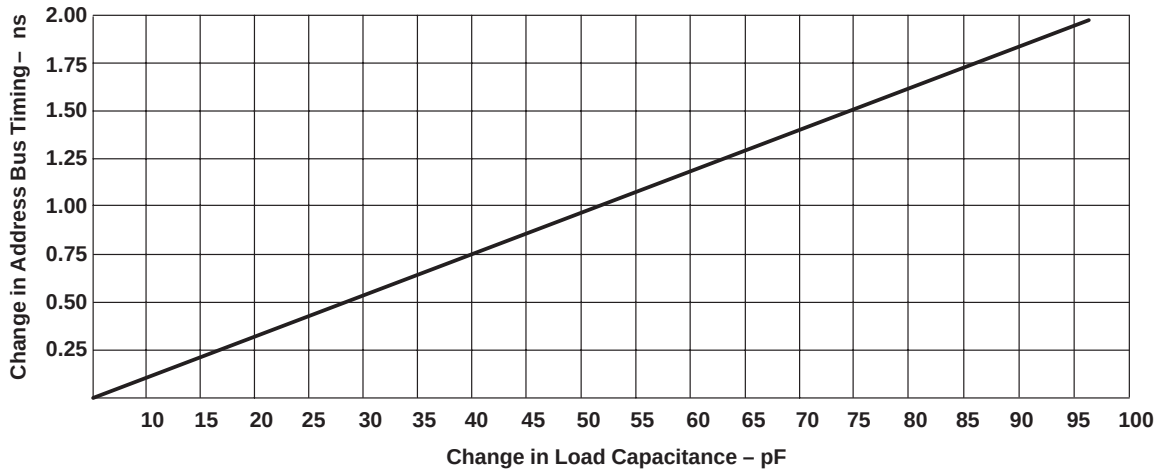


Figure 16. Address Bus Timing Variation With Load Capacitance

READY TIMING FOR EXTERNALLY-GENERATED WAIT STATES

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature (see Note 5) (see Figure 17 and Figure 18)

		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{su(RY-COH)}$	Setup time, READY before CLKOUT1 rising edge	10		7		6		ns
$t_{su(RY-RDL)}$	Setup time, READY before $\overline{RD}$ falling edge	10		7		6		ns
$t_{h(COH-RYH)}$	Hold time, READY after CLKOUT1 rising edge	0		0		0		ns
$t_{h(RDL-RY)}$	Hold time, READY after $\overline{RD}$ falling edge	0		0		0		ns
$t_{h(WEL-RY)}$	Hold time, READY after $\overline{WE}$ falling edge	H + 5		H + 4		H + 3		ns
$t_{v(WEL-RY)}$	Valid time, READY after $\overline{WE}$ falling edge		H – 15		H – 10		H – 8	ns

NOTE 5: The external READY input is sampled only after the internal software wait states are completed.

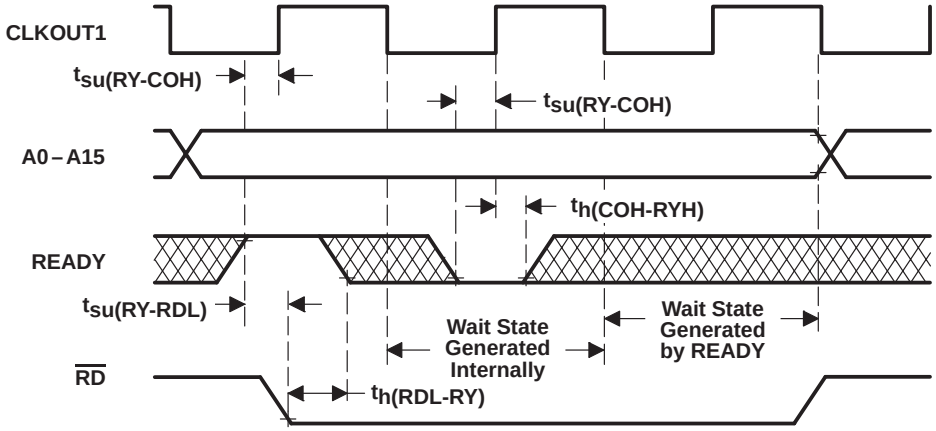


Figure 17. Ready Timing for Externally-Generated Wait States During an External Read Cycle

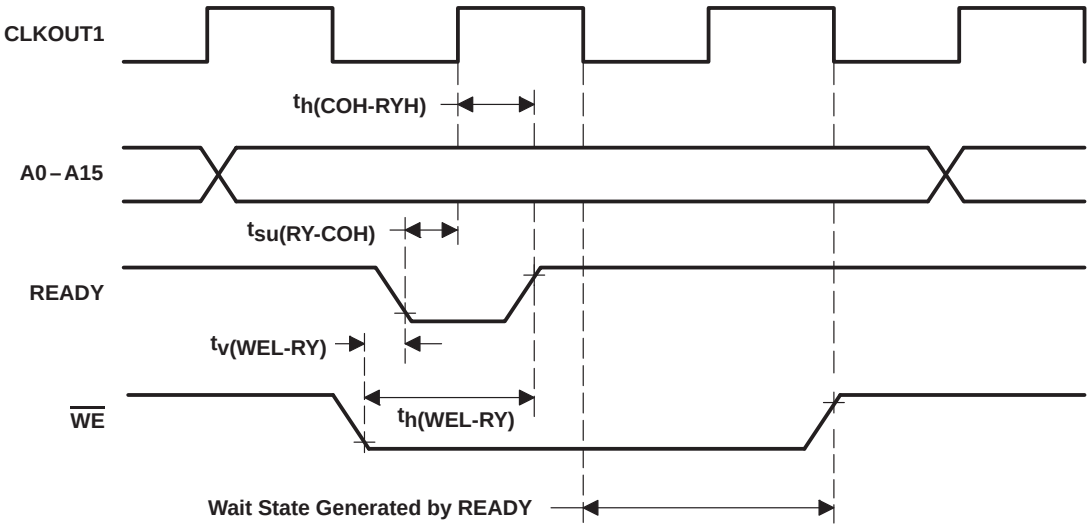


Figure 18. Ready Timing for Externally-Generated Wait States During an External Write Cycle

RESET, INTERRUPT, AND $\overline{\text{BIO}}$

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] (see Figure 19)

		’320C5x-40 ’320C5x-57 ’320LC5x-40 ’320LC5x-50		’320C5x-80 ’320C5x-100 ’320LC5x-80		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(IN-COL)}$	Setup time, $\overline{\text{INT1}}$–$\overline{\text{INT4}}$, $\overline{\text{NMI}}$ before CLKOUT1 low †	15		10		ns
$t_{su(RS-COL)}$	Setup time, $\overline{\text{RS}}$ before CLKOUT1 low	15	2H – 5‡	10	2H – 5‡	ns
$t_{su(RS-CIL)}$	Setup time, $\overline{\text{RS}}$ before X2/CLKIN low	10		7		ns
$t_{su(BI-COL)}$	Setup time, $\overline{\text{BIO}}$ before CLKOUT1 low	15		10		ns
$t_{h(COL-IN)}$	Hold time, $\overline{\text{INT1}}$–$\overline{\text{INT4}}$, $\overline{\text{NMI}}$ after CLKOUT1 low †	0		0		ns
$t_{h(COL-BI)}$	Hold time, $\overline{\text{BIO}}$ after CLKOUT1 low	0		0		ns
$t_{w(INL)SYN}$	Pulse duration, $\overline{\text{INT1}}$–$\overline{\text{INT4}}$, $\overline{\text{NMI}}$ low, synchronous	4H + 15§		4H + 10§		ns
$t_{w(INH)SYN}$	Pulse duration, $\overline{\text{INT1}}$–$\overline{\text{INT4}}$, $\overline{\text{NMI}}$ high, synchronous	2H + 15§		2H + 10§		ns
$t_{w(INL)ASY}$	Pulse duration, $\overline{\text{INT1}}$–$\overline{\text{INT4}}$, $\overline{\text{NMI}}$ low, asynchronous ‡	6H + 15§		6H + 10§		ns
$t_{w(INH)ASY}$	Pulse duration, $\overline{\text{INT1}}$–$\overline{\text{INT4}}$, $\overline{\text{NMI}}$ high, asynchronous ‡	4H + 15§		4H + 10§		ns
$t_{w(RSL)}$	Pulse duration, $\overline{\text{RS}}$ low	12H		12H		ns
$t_{w(BIL)SYN}$	Pulse duration, $\overline{\text{BIO}}$ low, synchronous	15		10		ns
$t_{w(BIL)ASY}$	Pulse duration, $\overline{\text{BIO}}$ low, asynchronous ‡	H + 15		H + 10		ns
$t_{d(RSH)}$	Delay time, $\overline{\text{RS}}$ high to reset vector fetch	34H		34H		ns

† These parameters must be met to use the synchronous timings. Both reset and the interrupts can operate asynchronously. The pulse durations require an extra half-cycle to ensure internal synchronization.
‡ Values derived from characterization data and not tested
§ If in IDLE2, add 4H to these timings.

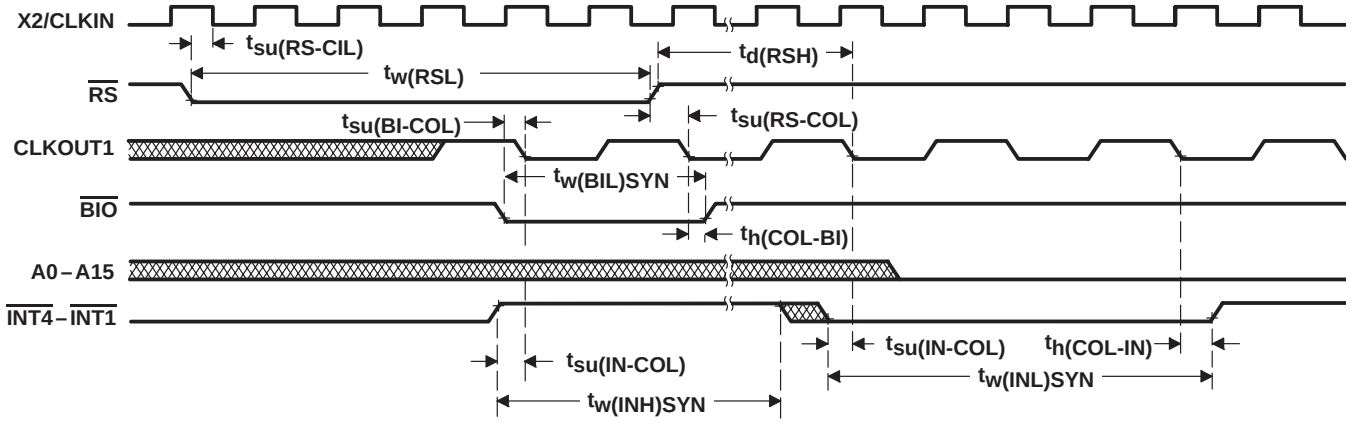


Figure 19. Reset, Interrupt, and $\overline{\text{BIO}}$ Timings

INSTRUCTION ACQUISITION (IAQ), INTERRUPT ACKNOWLEDGE (IACK), EXTERNAL FLAG (XF), AND TOUT (SEE NOTE 6)

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Figure 20)

PARAMETER		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320C5x-100 '320LC5x-80		UNIT
		MIN	MAX	MIN	MAX	
$t_{su(AV-IQL)}$	Setup time, address valid before $\overline{IAQ}$ low †	H – 12‡		H – 9‡		ns
$t_{h(IQL-AV)}$	Hold time, address valid after $\overline{IAQ}$ low	H – 10‡		H – 7‡		ns
$t_{w(IQL)}$	Pulse duration, $\overline{IAQ}$ low	H – 10‡		H – 7‡		ns
$t_{d(CO-TU)}$	Delay time, CLKOUT1 falling edge to TOUT	– 6	6	– 6	6	ns
$t_{su(AV-IKL)}$	Setup time, address valid before $\overline{IACK}$ low §	H – 12‡		H – 9‡		ns
$t_{h(IKL-AV)}$	Hold time, address valid after $\overline{IACK}$ low	H – 10‡		H – 7‡		ns
$t_{w(IKL)}$	Pulse duration, $\overline{IACK}$ low	H – 10‡		H – 7‡		ns
$t_{w(TUH)}$	Pulse duration, TOUT high	2H – 12		2H – 9		ns
$t_{d(CO-XFV)}$	Delay time, XF valid after CLKOUT1	0	12	0	9	ns

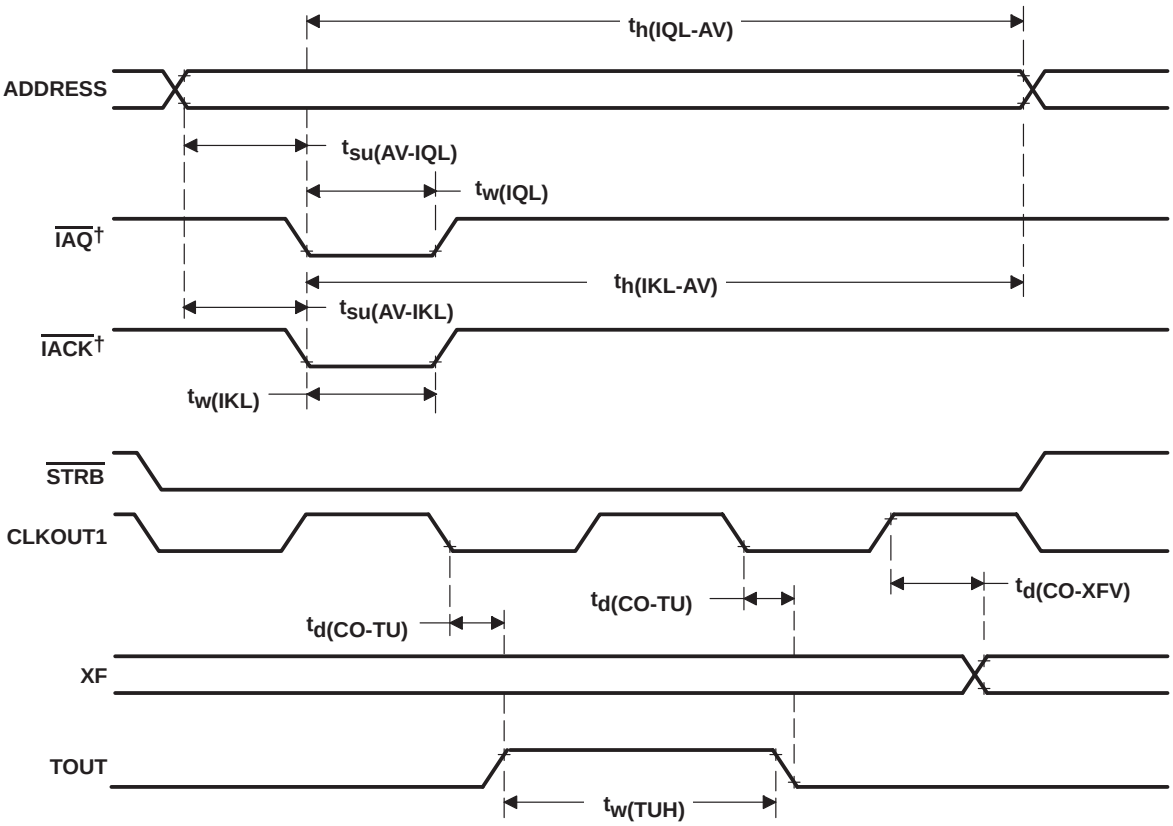
† $\overline{IAQ}$ goes low during an instruction acquisition. It goes low only on the first cycle of the read when wait states are used. The falling edge should be used to latch the valid address. The AVIS bit in the PMST register must be set to zero for the address to be valid when the instruction being addressed resides in on-chip memory.

‡ Valid only if the external address reflects the current instruction activity (that is, code is executing on chip with no external bus cycles and AVIS is on or code is executing off chip)

§ $\overline{IACK}$ goes low during the fetch of the first word of the interrupt vector. It goes low only on the first cycle of the read when wait states are used. Address pins A1 – A4 can be decoded at the falling edge to identify the interrupt being acknowledged. The AVIS bit in the PMST register must be set to zero for the address to be valid when the vectors reside in on-chip memory.

NOTE 6: $\overline{IAQ}$ pin is not present on 100-pin packages.
$\overline{IACK}$ pin is not present on 100-pin and 128-pin packages.

INSTRUCTION ACQUISITION (IAQ), INTERRUPT ACKNOWLEDGE (IACK), EXTERNAL FLAG (XF), AND TOUT (SEE NOTE 6) (CONTINUED)



† $\overline{\text{IAQ}}$ and $\overline{\text{IACK}}$ are not affected by wait states.

Figure 20. $\overline{\text{IAQ}}$, $\overline{\text{IACK}}$, and XF Timings Example With Two External Wait States

EXTERNAL DMA

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Note 7) (see Figure 21)

PARAMETER		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{d(HOL-HAL)}$	Delay time, $\overline{HOLD}$ low to $\overline{HOLDA}$ low	4H	†	4H	†	4H	†	ns
$t_{d(HOH-HAH)}$	Delay time, $\overline{HOLD}$ high before $\overline{HOLDA}$ high	2H		2H		2H		ns
$t_{h(AZ-HAL)}$	Address high-impedance before $\overline{HOLDA}$ low‡	H – 15§		H – 10§		H – 8§		ns
$t_{en(HAH-Ad)}$	Enable time, $\overline{HOLDA}$ high to address driven	H – 5§		H – 4§		H – 3§		ns
$t_{d(XBL-IQL)}$	Delay time, $\overline{XBR}$ low to $\overline{IAQ}$ low	4H§	6H§	4H§	6H§	4H§	6H§	ns
$t_{d(XBH-IQH)}$	Delay time, $\overline{XBR}$ high to $\overline{IAQ}$ high	2H§	4H§	2H§	4H§	2H§	4H§	ns
$t_{d(XSL-RDV)}$	Delay time, read data valid after $\overline{XSTRB}$ low		40		29		25	ns
$t_{h(XSH-RD)}$	Hold time, read data valid after $\overline{XSTRB}$ high	0		0		0		ns
$t_{en(IQL-RDd)}$	Enable time, $\overline{IAQ}$ low to read data driven¶	0§	2H§	0§	2H§	0§	2H§	ns
$t_{h(XRL-DZ)}$	Hold time, XR/$\overline{W}$ low to data high impedance	0§	15§	0§	10§	0§	8	ns
$t_{h(IQH-DZ)}$	Hold time, $\overline{IAQ}$ high to data high impedance		H§		H§		H§	ns
$t_{en(D-XRH)}$	Enable time, data from XR/$\overline{W}$ going high		4§		3§		2§	ns

† $\overline{HOLD}$ is not acknowledged until current external access request is complete.
‡ This parameter includes all memory control lines.
§ Values derived from characterization data and not tested
¶ This parameter refers to the delay between the time the condition ($\overline{IAQ}$ = 0 and XR/$\overline{W}$ = 1) is satisfied and the time that the 'C5x data lines become valid.
NOTE 7: X preceding a name refers to external drive of the signal.

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature (see Note 7) (see Figure 21)

		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{d(HAL-XBL)}$	Delay time, $\overline{HOLDA}$ low to $\overline{XBR}$ low#	0§		0§		0§		ns
$t_{d(IQL-XSL)}$	Delay time, $\overline{IAQ}$ low to $\overline{XSTRB}$ low#	0§		0§		0§		ns
$t_{su(AV-XSL)}$	Setup time, Xaddress valid before $\overline{XSTRB}$ low	15		12		10		ns
$t_{su(DV-XSL)}$	Setup time, Xdata valid before $\overline{XSTRB}$ low	15		12		10		ns
$t_{h(XSL-D)}$	Hold time, Xdata hold after $\overline{XSTRB}$ low	15		12		10		ns
$t_{h(XSL-WA)}$	Hold time, write Xaddress hold after $\overline{XSTRB}$ low	15		12		10		ns
$t_{w(XSL)}$	Pulse duration, $\overline{XSTRB}$ low	45		40		35		ns
$t_{w(XSH)}$	Pulse duration, $\overline{XSTRB}$ high	45		40		35		ns
$t_{su(RW-XSL)}$	Setup time, R/$\overline{W}$ valid before $\overline{XSTRB}$ low	20		20		18		ns
$t_{h(XSH-RA)}$	Hold time, read Xaddress after $\overline{XSTRB}$ high	0		0		0		ns

§ Values derived from characterization data and not tested
# $\overline{XBR}$, XR/$\overline{W}$, and $\overline{XSTRB}$ lines must be pulled up with a 10-kΩ resistor to be certain that they are in an inactive high state during the transition period between the 'C5x driving them and the external circuit driving them.
NOTE 7: X preceding a name refers to external drive of the signal.

EXTERNAL DMA (CONTINUED)

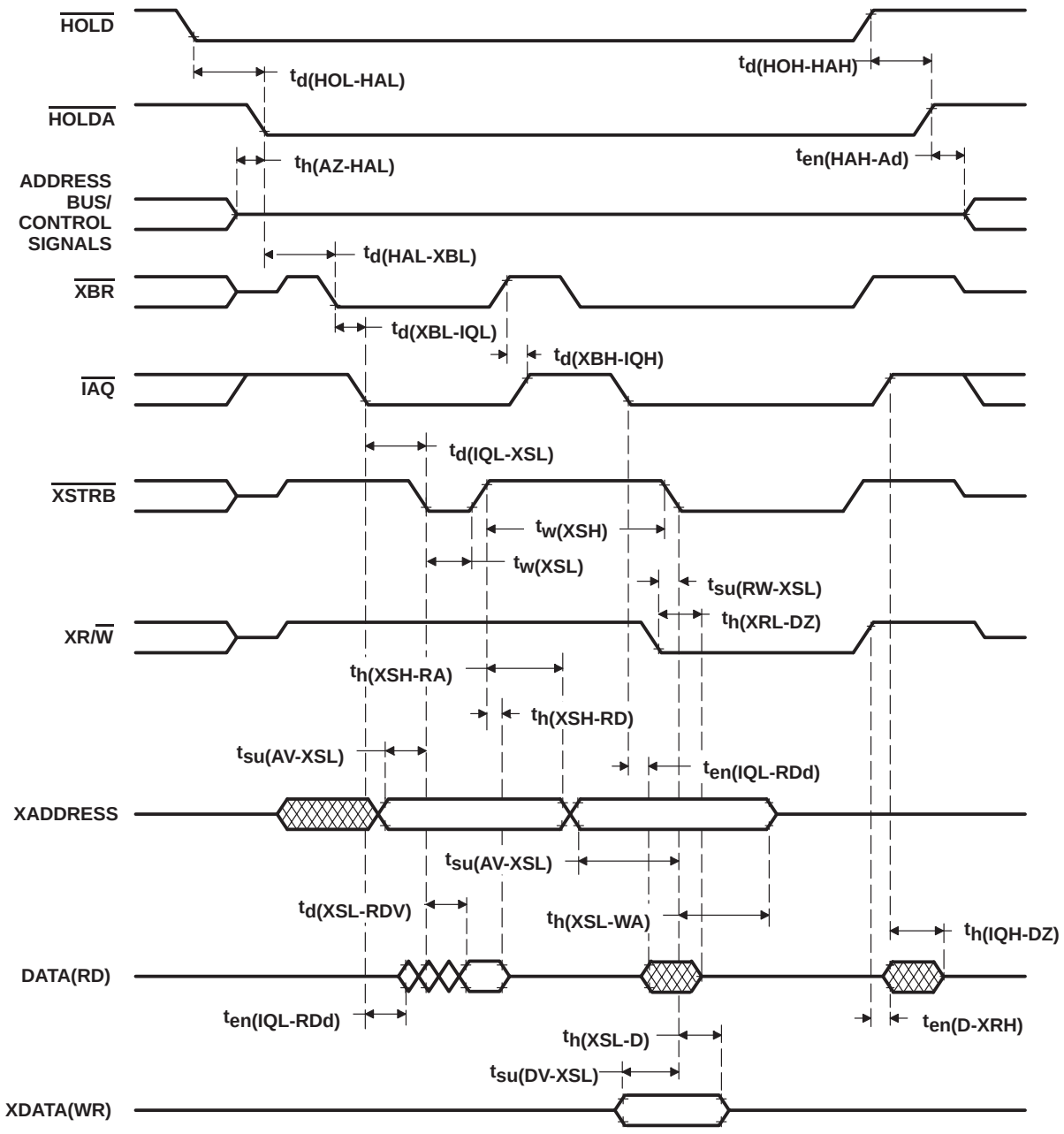


Figure 21. External DMA Timing

SERIAL-PORT RECEIVE TIMING

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] (see Figure 22)

		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{c(SCK)}$	Cycle time, serial-port clock	5.2H†	‡	5.2H†	‡	5.2H†	‡	ns
$t_{f(SCK)}$	Fall time, serial-port clock		8§		6§		6§	ns
$t_{r(SCK)}$	Rise time, serial-port clock		8§		6§		6§	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	2.1H†		2.1H†		2.1H†		ns
$t_{su(FS-CK)}$	Setup time, FSR before CLKR falling edge	10		7		6		ns
$t_{su(DR-CK)}$	Setup time, DR before CLKR falling edge	10		7		6		ns
$t_{h(CK-FS)}$	Hold time, FSR after CLKR falling edge	10		7		6		ns
$t_{h(CK-DR)}$	Hold time, DR valid after CLKR falling edge	10		7		6		ns

† Values ensured by design but not tested
‡ The serial-port design is fully static and, therefore, can operate with $t_{c(SCK)}$ approaching ∞. It is characterized approaching an input frequency of 0 Hz but tested at a much higher frequency to minimize test time.
§ Values derived from characterization data and not tested

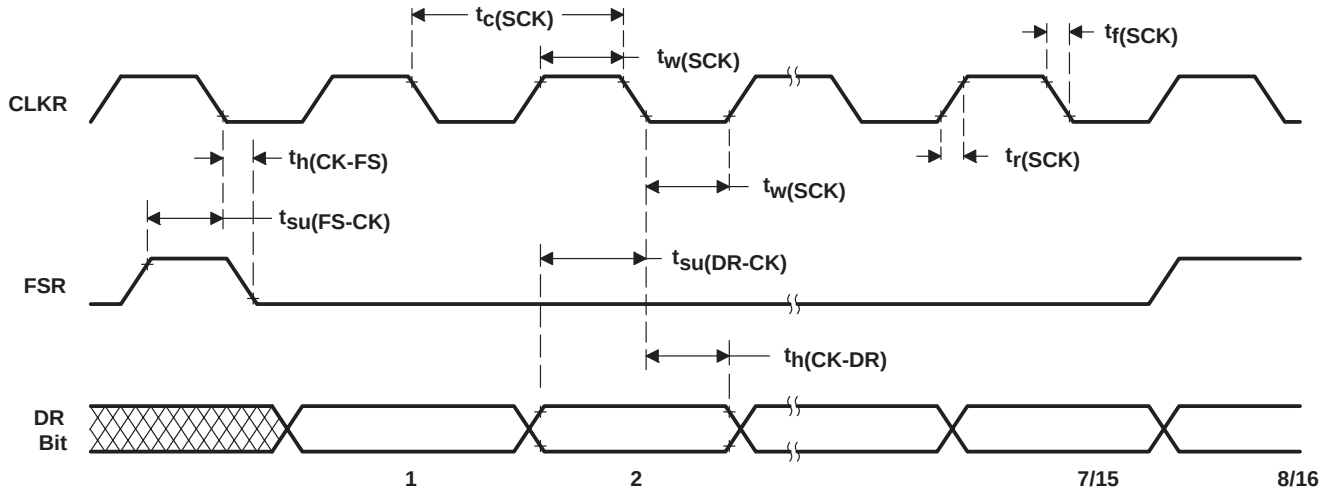


Figure 22. Serial-Port Receive Timing

SERIAL-PORT TRANSMIT TIMING, EXTERNAL CLOCKS, AND EXTERNAL FRAMES

switching characteristics over recommended operating conditions (see Note 8) (see Figure 23)

PARAMETER		MIN	MAX	UNIT
$t_{d(CXH-DXV)}$	Delay time, DX valid after CLKX high		25	ns
$t_{dis(CXH-DX)}$	Disable time, DX invalid after CLKX high		40†	ns
$t_{h(CXH-DXV)}$	Hold time, DX valid after CLKX high	– 5		ns

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = $0.5t_{c(CO)}$] (see Note 8) (see Figure 23)

		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{c(SCK)}$	Cycle time, serial-port clock	5.2H‡	§	5.2H‡	§	5.2H‡	§	ns
$t_{f(SCK)}$	Fall time, serial-port clock		8†		6†		6†	ns
$t_{r(SCK)}$	Rise time, serial-port clock		8†		6†		6†	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	2.1H‡		2.1H‡		2.1H‡		ns
$t_{d(CXH-FXH)}$	Delay time, FSX high after CLKX high		2H – 8		2H – 8		2H – 5	ns
$t_{h(CXL-FXL)}$	Hold time, FSX low after CLKX low	10		7		6		ns
$t_{h(CXH-FXL)}$	Hold time, FSX low after CLKX high		2H – 8¶		2H – 8¶		2H – 5¶	ns

† Values derived from characterization data and not tested
‡ Values ensured by design but not tested
§ The serial-port design is fully static and, therefore, can operate with $t_{c(SCK)}$ approaching ∞. It is characterized approaching an input frequency of 0 Hz but tested at a much higher frequency to minimize test time.
¶ If the FSX pulse does not meet this specification, the first bit of serial data is driven on the DX pin until the falling edge of FSX. After the falling edge of FSX, data is shifted out on the DX pin. The transmit buffer empty interrupt is generated when the $t_{h(CXL-FXL)}$ and $t_{h(CXH-FXL)}$ specification is met.

NOTE 8: Internal clock with external FSX and vice versa are also allowable. However, FSX timings to CLKX always are defined depending on the source of FSX, and CLKX timings always are dependent on the source of CLKX. Specifically, the relationship of FSX to CLKX is independent of the source of CLKX.

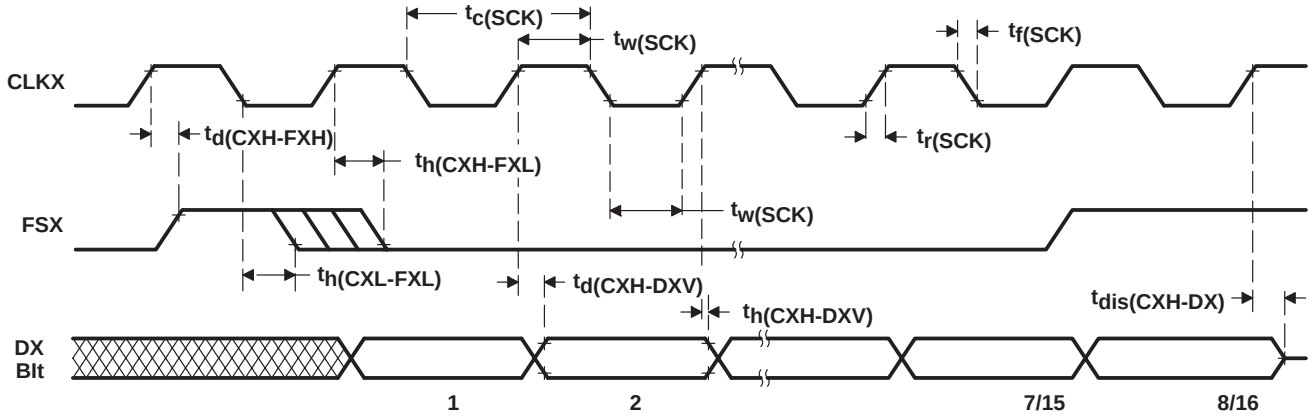


Figure 23. Serial-Port Transmit Timing of External Clocks and External Frames

SERIAL-PORT TRANSMIT TIMING, INTERNAL CLOCKS, AND INTERNAL FRAMES (SEE NOTE 8)

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Figure 24)

PARAMETER		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50			'320C5x-80 '320C5x-100 '320LC5x-80			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	
$t_{d(CX-FX)}$	Delay time, CLKX rising edge to FSX	– 5		25	– 4		18	ns
$t_{d(CX-DX)}$	Delay time, CLKX rising edge to DX			25			18	ns
$t_{dis(CX-DX)}$	Disable time, CLKX rising edge to DX			40†			29†	ns
$t_{c(SCK)}$	Cycle time, serial-port clock		8H			8H		ns
$t_{f(SCK)}$	Fall time, serial-port clock		5			4		ns
$t_{r(SCK)}$	Rise time, serial-port clock		5			4		ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	4H – 20			4H – 14			ns
$t_{h(CXH-DXV)}$	Hold time, DX valid after CLKX high	– 5			– 4			ns

† Values derived from characterization data and not tested

NOTE 8: Internal clock with external FSX and vice versa are also allowable. However, FSX timings to CLKX always are defined depending on the source of FSX, and CLKX timings always are dependent on the source of CLKX. Specifically, the relationship of FSX to CLKX is independent of the source of CLKX.

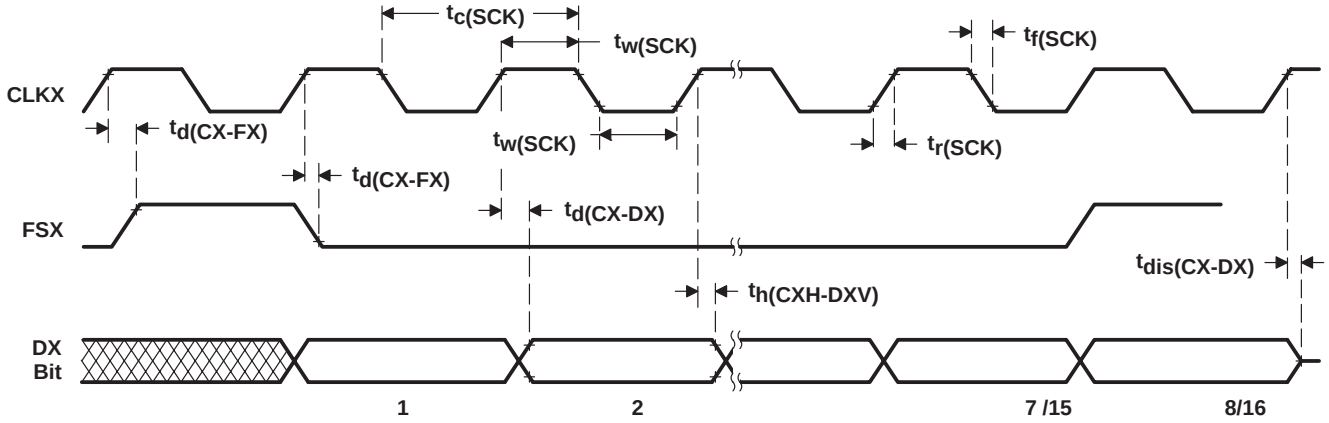


Figure 24. Serial-Port Transmit Timing of Internal Clocks and Internal Frames

SERIAL-PORT RECEIVE TIMING IN TDM MODE

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] (see Figure 25)

		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{c(SCK)}$	Cycle time, serial-port clock	5.2H†	‡	5.2H†	‡	5.2H§	‡	ns
$t_{f(SCK)}$	Fall time, serial-port clock		8¶		8¶		8¶	ns
$t_{r(SCK)}$	Rise time, serial-port clock		8¶		8¶		8¶	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	2.1H†		2.1H†		2.1H†		ns
$t_{su(TD-TCH)}$	Setup time, TDAT before TCLK rising edge	30		21		18		ns
$t_{h(TCH-TD)}$	Hold time, TDAT after TCLK rising edge	– 3		– 2		– 2		ns
$t_{su(TA-TCH)}$	Setup time, TADD before TCLK rising edge#	20		12		10		ns
$t_{h(TCH-TA)}$	Hold time, TADD after TCLK rising edge#	– 3		– 2		– 2		ns
$t_{su(TF-TCH)}$	Setup time, TFRM before TCLK rising edge§	10		10		10		ns
$t_{h(TCH-TF)}$	Hold time, TFRM after TCLK rising edge§	10		10		10		ns

† Values ensured by design and are not tested
‡ The serial-port design is fully static and, therefore, can operate with $t_{c(SCK)}$ approaching ∞. It is characterized approaching an input frequency of 0 Hz but tested at a much higher frequency to minimize test time.
§ TFRM timing and waveforms shown in Figure 25 are for external TFRM. TFRM also can be configured as internal. The TFRM internal case is illustrated in the transmit timing diagram in Figure 26.
¶ Values derived from characterization data and not tested
# These parameters apply only to the first bits in the serial bit string.

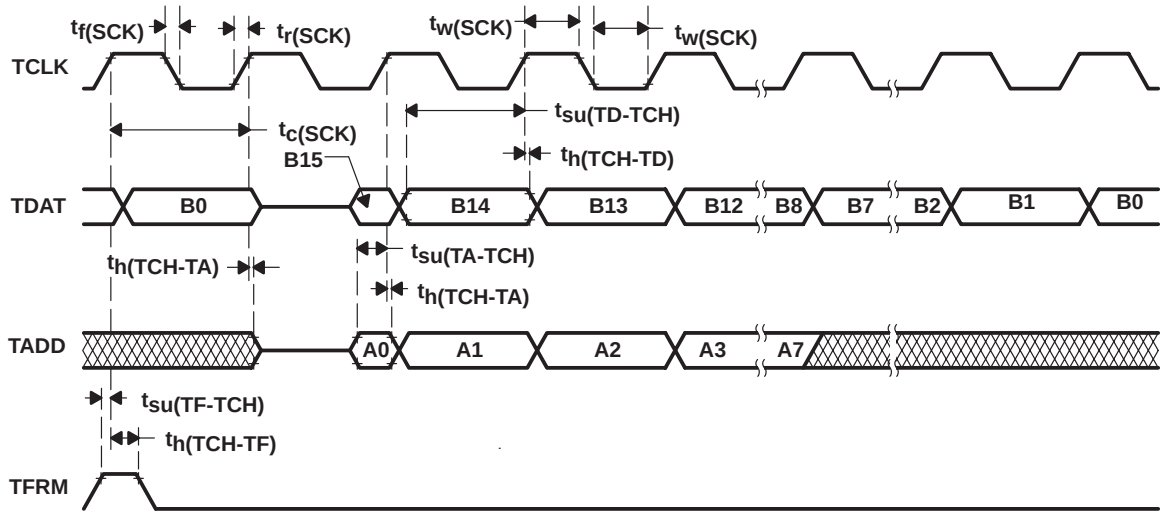


Figure 25. Serial-Port Receive Timing in TDM Mode

SERIAL-PORT TRANSMIT TIMING IN TDM MODE

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Figure 26)

PARAMETER		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50		'320C5x-80 '320LC5x-80		'320C5x-100		UNIT
		MIN	MAX	MIN	MAX	MIN	MAX	
$t_{h(TCH-TDV)}$	Hold time, TDAT/TADD valid after TCLK rising edge	0		0		0		ns
$t_{d(TCH-TFV)}$	Delay time, TFRM valid after TCLK rising edge†	H	3H + 10	H	3H + 7		3H + 5	ns
$t_{d(TC-TDV)}$	Delay time, TCLK to valid TDAT/TADD		20		15		12	ns

† TFRM timing and waveforms shown in Figure 28 are for internal TFRM. TFRM can also be configured as external. The TFRM external case is illustrated in the receive timing diagram in Figure 27.

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] (see Figure 26)

		'320C5x-40 '320C5x-57 '320LC5x-40 '320LC5x-50			'320C5x-80 '320LC5x-80			'320C5x-100			UNIT
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX	
$t_{c(SCK)}$	Cycle time, serial-port clock	5.2H‡	8H§	¶	5.2H‡	8H§	¶	5.2H‡	8H§	¶	ns
$t_{f(SCK)}$	Fall time, serial-port clock			8#			6#			5#	ns
$t_{r(SCK)}$	Rise time, serial-port clock			8#			6#			5#	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/ high	2.1H‡			2.1H‡			2.1H‡			ns

‡ Values ensured by design and are not tested
§ When SCK is generated internally
¶ The serial-port design is fully static and, therefore, can operate with $t_{c(SCK)}$ approaching ∞. It is characterized approaching an input frequency of 0 Hz but tested as a much higher frequency to minimize test time.
# Values derived from characterization data and not tested

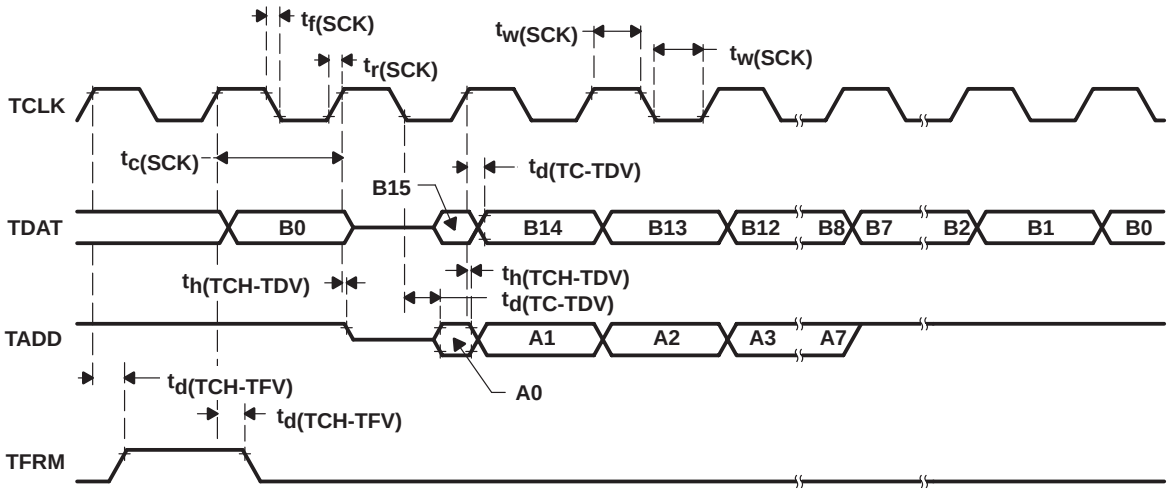


Figure 26. Serial-Port Transmit Timing in TDM Mode

BUFFERED SERIAL-PORT RECEIVE TIMING

timing requirements over recommended ranges of supply voltage and operating ambient-air temperature [H = 0.5$t_{c(CO)}$] (see Figure 27)

		MIN	MAX	UNIT
$t_{c(SCK)}$	Cycle time, serial-port clock	25	†	ns
$t_{f(SCK)}$	Fall time, serial-port clock		6‡	ns
$t_{r(SCK)}$	Rise time, serial-port clock		6‡	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	12		ns
$t_{su(FS-CK)}$	Setup time, FSR before CLKR falling edge	2		ns
$t_{su(DR-CK)}$	Setup time, DR before CLKR falling edge	0		ns
$t_{h(CK-FS)}$	Hold time, FSR after CLKR falling edge	12	$t_{c(SCK)}$§	ns
$t_{h(CK-DR)}$	Hold time, DR after CLKR falling edge	15		ns

† The serial-port design is fully static and, therefore, can operate with $t_{c(SCK)}$ approaching ∞. It is characterized approaching an input frequency of 0 Hz but tested at a much higher frequency to minimize test time.
‡ Values derived from characterization data and not tested
§ First bit is read when FSR is sampled low by CLKR clock.

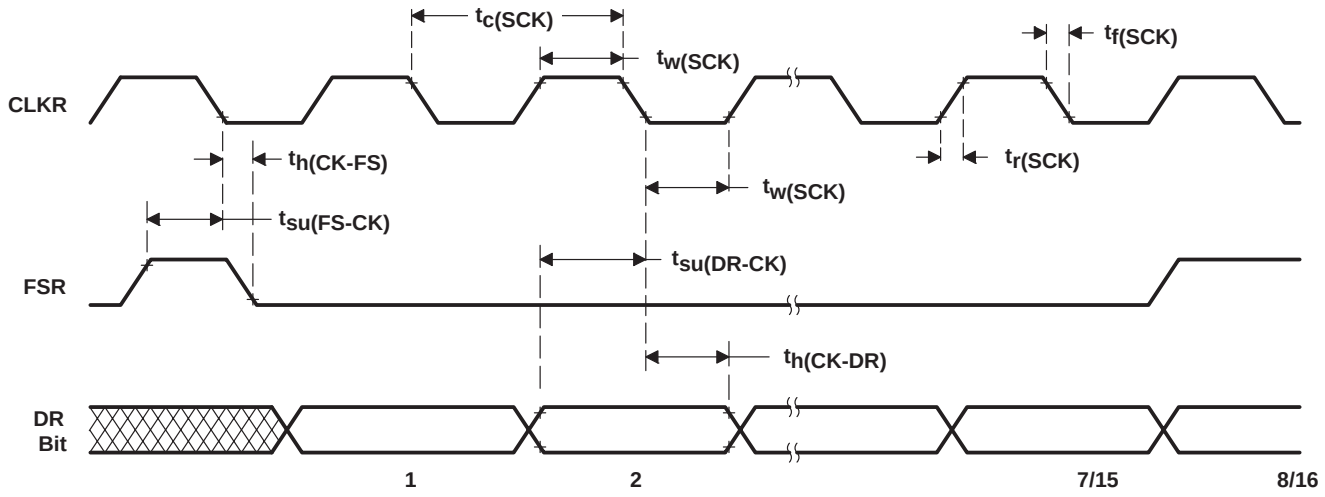


Figure 27. Buffered Serial-Port Receive Timing

BUFFERED SERIAL-PORT TRANSMIT TIMING OF EXTERNAL FRAMES (SEE NOTES 9 AND 10)

switching characteristics over recommended operating conditions (see Figure 28)

PARAMETER		MIN	MAX	UNIT
$t_{d(CXH-DXV)}$	Delay time, DX valid after CLKX rising edge	5	21	ns
$t_{dis(CXH-DX)}$	Disable time, DX invalid after CLKX rising edge	5	15	ns
$t_{dis(CXH-DX)PCM}$	Disable time in PCM mode, DX invalid after CLKX rising edge		15	ns
$t_{en(CXH-DX)PCM}$	Enable time in PCM mode, DX valid after CLKX rising edge	21		ns
$t_{h(CXH-DXV)}$	Hold time, DX valid after CLKX rising edge	5	20	ns

timing requirements over recommended operating conditions (see Figure 28)

		MIN	MAX	UNIT
$t_{c(SCK)}$	Cycle time, serial-port clock	25	†	ns
$t_{f(SCK)}$	Fall time, serial-port clock		4‡	ns
$t_{r(SCK)}$	Rise time, serial-port clock		4‡	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	8.5		ns
$t_{su(FX-CXL)}$	Setup time, FSX before CLKX falling edge	5		ns
$t_{h(CXL-FX)}$	Hold time, FSX after CLKX falling edge	5	$t_{c(SCK)}-5$§	ns

† The serial-port design is fully static and, therefore, can operate with $t_{c(SCK)}$ approaching ∞. It is characterized approaching an input frequency of 0 Hz but tested at a much higher frequency to minimize test time.

‡ Values derived from characterization data and not tested

§ If the FSX pulse does not meet this specification, the first bit of the serial data is driven on the DX pin until FSX goes low (sampled on falling edge of CLKX). After falling edge of the FSX, data is shifted out on the DX pin.

NOTE 9: Internal clock with external FSX and vice versa are also allowable. However, FSX timings to CLKX always are defined depending on the source of FSX, and CLKX timings always are dependent upon the source of CLKX. Specifically, the relationship of FSX to CLKX is independent of the source of CLKX. External FSX timings are obtained from the “timing requirements over recommended operating conditions” table listed in the “Buffered Serial-Port Transmit Timing of External Frames” section and internal FSX timings are obtained from the “switching characteristics over recommended operating conditions” table listed under the “Buffered Serial-Port Transmit Timing of Internal Frame and Internal Clock” section. Internal CLKX timings are obtained from the “switching characteristics over recommended operating conditions” table listed under the “Buffered Serial-Port Transmit Timing of Internal Frame and Internal Clock” section and external CLKX timings are obtained from the “timing requirements over recommended operating conditions” table in the “Buffered Serial-Port Transmit Timing of External Frames” section.

NOTE 10: Timings for CLKX and FSX are given with polarity bits (CLKP and FSP) set to 0

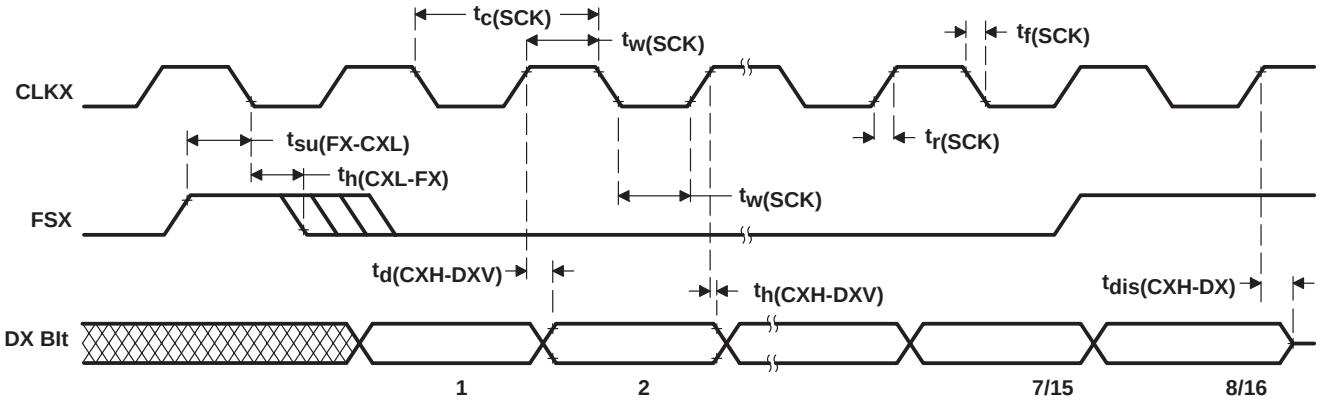


Figure 28. Buffered Serial-Port Transmit Timing of External Clocks and External Frames

BUFFERED SERIAL-PORT TRANSMIT TIMING OF INTERNAL FRAME AND INTERNAL CLOCK (SEE NOTES 9 AND 10)

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (see Figure 29)

PARAMETER		MIN	MAX	UNIT
$t_{d(CXH-FXH)}$	Delay time, FSX high after CLKX rising edge		10	ns
$t_{d(CXH-FXL)}$	Delay time, FSX low after CLKX rising edge		10	ns
$t_{d(CXH-DXV)}$	Delay time, DX valid after CLKX rising edge	5	10	ns
$t_{dis(CXH-DX)}$	Disable time, DX invalid after CLKX rising edge	4	8	ns
$t_{dis(CXH-DX)PCM}$	Disable time in PCM mode, DX invalid after CLKX rising edge		10	ns
$t_{en(CXH-DX)PCM}$	Enable time in PCM mode, DX valid after CLKX rising edge	16		ns
$t_{c(SCK)}$	Cycle time, serial-port clock	2H	62H	ns
$t_{f(SCK)}$	Fall time, serial-port clock		4†	ns
$t_{r(SCK)}$	Rise time, serial-port clock		4†	ns
$t_{w(SCK)}$	Pulse duration, serial-port clock low/high	H–4		ns
$t_{h(CXH-DXV)}$	Hold time, DX valid after CLKX rising edge	4	8	ns

† Values derived from characterization data and not tested

NOTES: 9. Internal clock with external FSX and vice versa are also allowable. However, FSX timings to CLKX always are defined depending on the source of FSX, and CLKX timings always are dependent upon the source of CLKX. Specifically, the relationship of FSX to CLKX is independent of the source of CLKX. External FSX timings are obtained from the “timing requirements over recommended operating conditions” table listed in the “Buffered Serial-Port Transmit Timing of External Frames” section and internal FSX timings are obtained from the “switching characteristics over recommended operating conditions” table listed under the “Buffered Serial-Port Transmit Timing of Internal Frame and Internal Clock” section. Internal CLKX timings are obtained from the “switching characteristics over recommended operating conditions” table listed under the “Buffered Serial-Port Transmit Timing of Internal Frame and Internal Clock” section and external CLKX timings are obtained from the “timing requirements over recommended operating conditions” table in the “Buffered Serial-Port Transmit Timing of External Frames” section.

10. Timings for CLKX and FSX are given with polarity bits (CLKP and FSP) set to 0.

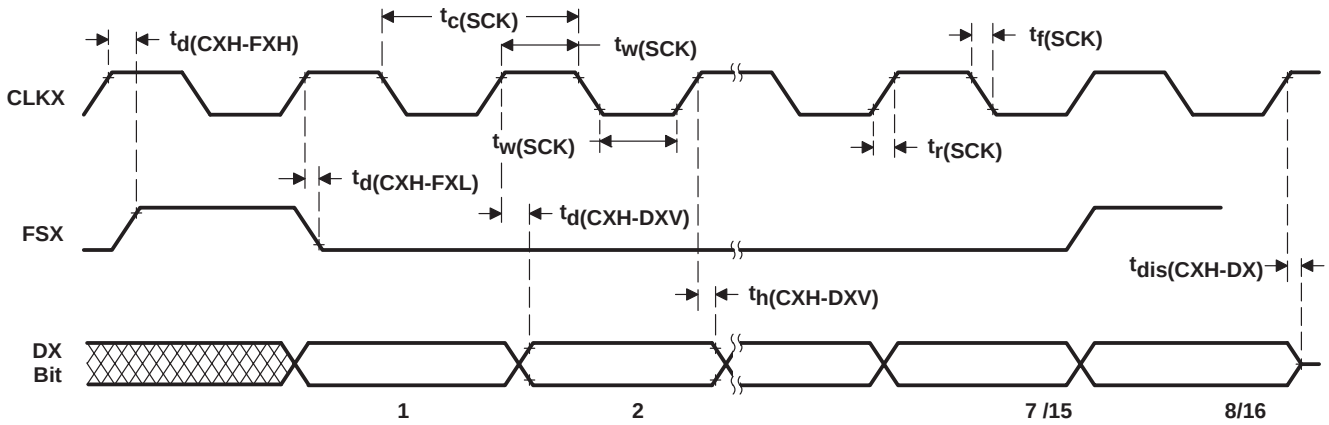


Figure 29. Buffered Serial-Port Transmit Timing of Internal Clocks and Internal Frames

HOST PORT INTERFACE (TMS320C57S, TMS320LC57 ONLY)

switching characteristics over recommended operating conditions [H = 0.5$t_{c(CO)}$] (See Notes 11 and 12) (see Figure 30 through Figure 33)

PARAMETER		MIN	MAX	UNIT
$t_{d(DSL-HDV)}$	Delay time, $\overline{DS}$ low to HD valid	5		ns
$t_{d(HEL-HDV1)}$	Delay time, $\overline{HDS}$ falling to HD valid for first byte of a subsequent read: Case 1: Shared-access mode if $t_{w(HDS)h}$ < 7H † ‡ Case 2: Shared-access mode if $t_{w(HDS)h}$ > 7H Case 3: Host-only mode if $t_{w(HDS)h}$ < 7H Case 4: Host-only mode if $t_{w(HDS)h}$ > 7H		 7H+20–$t_{w(DSH)}$ 20 40–$t_{w(DSH)}$ 20	ns
$t_{d(DSL-HDV2)}$	Delay time, $\overline{DS}$ low to HD valid, second byte		20	ns
$t_{d(DSH-HYH)}$	Delay time, $\overline{DS}$ high to HRDY high			ns
$t_{su(HDV-HYH)}$	Setup time, HD valid before HRDY rising edge	3H–10		ns
$t_{h(DSH-HDV)}$	Hold time, HD valid after $\overline{DS}$ rising edge	0	12§	ns
$t_{d(COH-HYH)}$	Delay time, CLKOUT rising edge to HRDY high		10	ns
$t_{d(DSH-HYL)}$	Delay time, $\overline{HDS}$ or $\overline{HCS}$ high to HRDY low		12	ns
$t_{d(COH-HTX)}$	Delay time, CLKOUT rising edge to $\overline{HINT}$ change		10	ns

† Host-only mode timings apply for read accesses to HPIC or HPIA, write accesses to BOB, and resetting DSPINT or HINT to 0 in shared-access mode. HRDY does not go low for these accesses.
‡ Shared-access mode timings are met automatically if HRDY is used.
§ HD release
NOTES: 11. SAM = shared-access mode, HOM = host-only mode
HAD stands for HCNTRL0, HCNTRL1, and HR/$\overline{W}$.
HDS refers to either $\overline{HDS1}$ or $\overline{HDS2}$.
$\overline{DS}$ refers to the logical OR of $\overline{HCS}$ and $\overline{HDS}$.
12. On host-read accesses to the HPI, the setup time of HD before $\overline{DS}$ rising edge depends on the host waveforms and cannot be specified here.

timing requirements over recommended operating conditions [H = 0.5$t_{c(CO)}$] (See Note 11) (see Figure 30 through Figure 33)

		MIN	MAX	UNIT
$t_{su(HBV-DSL)}$	Setup time, HAD/HBIL valid before $\overline{HAS}$ or $\overline{DS}$ falling edge#	10		ns
$t_{h(DSL-HBV)}$	Hold time, HAD/HBIL valid after $\overline{HAS}$ or $\overline{DS}$ falling edge#	10		ns
$t_{su(HSL-DSL)}$	Setup time, $\overline{HAS}$ low before $\overline{DS}$ falling edge	10		ns
$t_{w(DSL)}$	Pulse duration, $\overline{DS}$ low	25		ns
$t_{w(DSH)}$	Pulse duration, $\overline{DS}$ high	10		ns
$t_{c(DSH-DSH)}$	Cycle time, $\overline{DS}$ rising edge to next $\overline{DS}$ rising edge: Case 1: When using HRDY (see Figure 32) Case 2a: SAM accesses and HOM active writes to DSPINT or HINT without using HRDY (see Figure 30 and Figure 31) Case 2b: When not using HRDY for other HOM accesses	 50 10H¶ 50		ns
$t_{su(HDV-DSH)}$	Setup time, HD valid before $\overline{DS}$ rising edge	10		ns
$t_{h(DSH-HDV)}$	Hold time, HD valid after $\overline{DS}$ rising edge	0		ns

¶ A host not using HRDY must meet the 10 H requirement all the time unless a software handshake is used to change the access rate according to the HPI mode.
# When $\overline{HAS}$ is tied to V_{DD}, timing is referenced to $\overline{DS}$.
NOTE 11: SAM = shared-access mode, HOM = host-only mode
HAD stands for HCNTRL0, HCNTRL1, and HR/$\overline{W}$.
HDS refers to either $\overline{HDS1}$ or $\overline{HDS2}$.
$\overline{DS}$ refers to the logical OR of $\overline{HCS}$ and $\overline{HDS}$.

HOST PORT INTERFACE (TMS320C57S, TMS320LC57 ONLY) (CONTINUED)

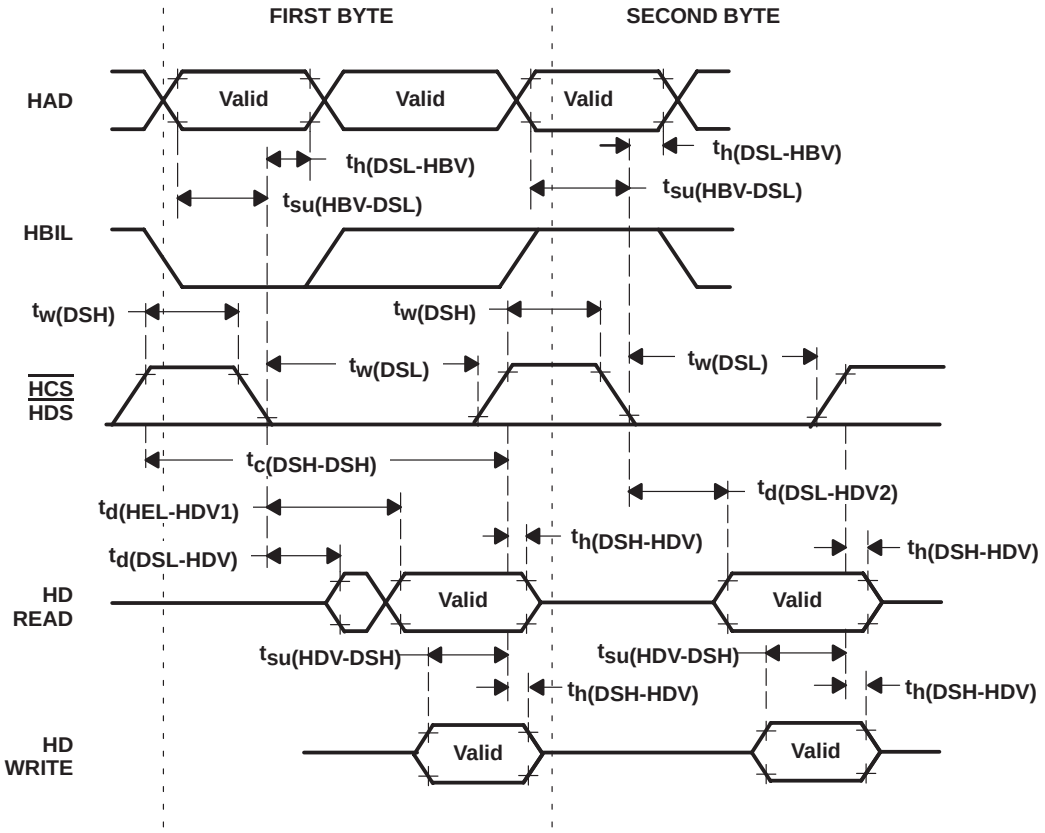


Figure 30. Read/Write Access Timings Without HRDY or $\overline{\text{HAS}}$

HOST PORT INTERFACE (TMS320C57S, TMS320LC57 ONLY) (CONTINUED)

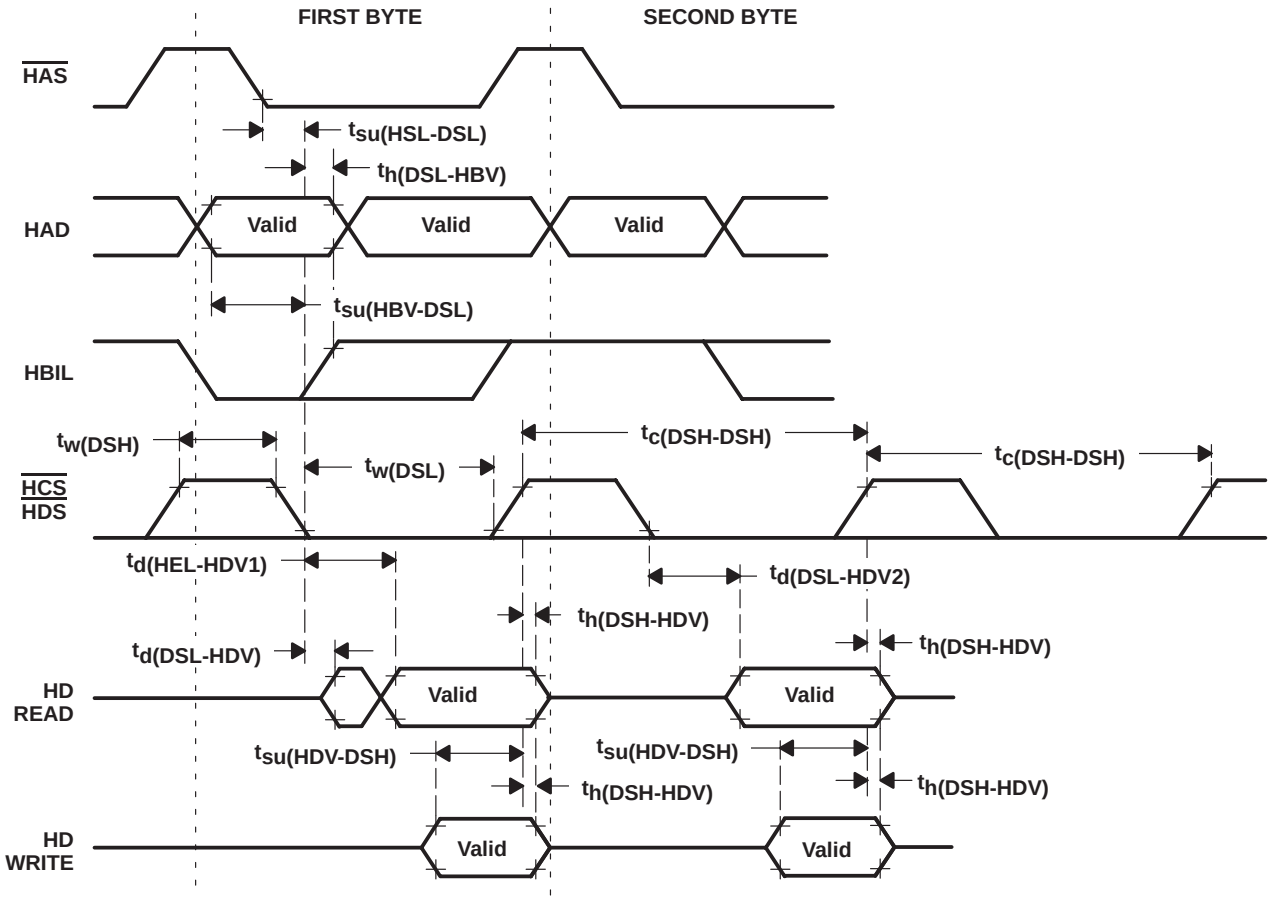
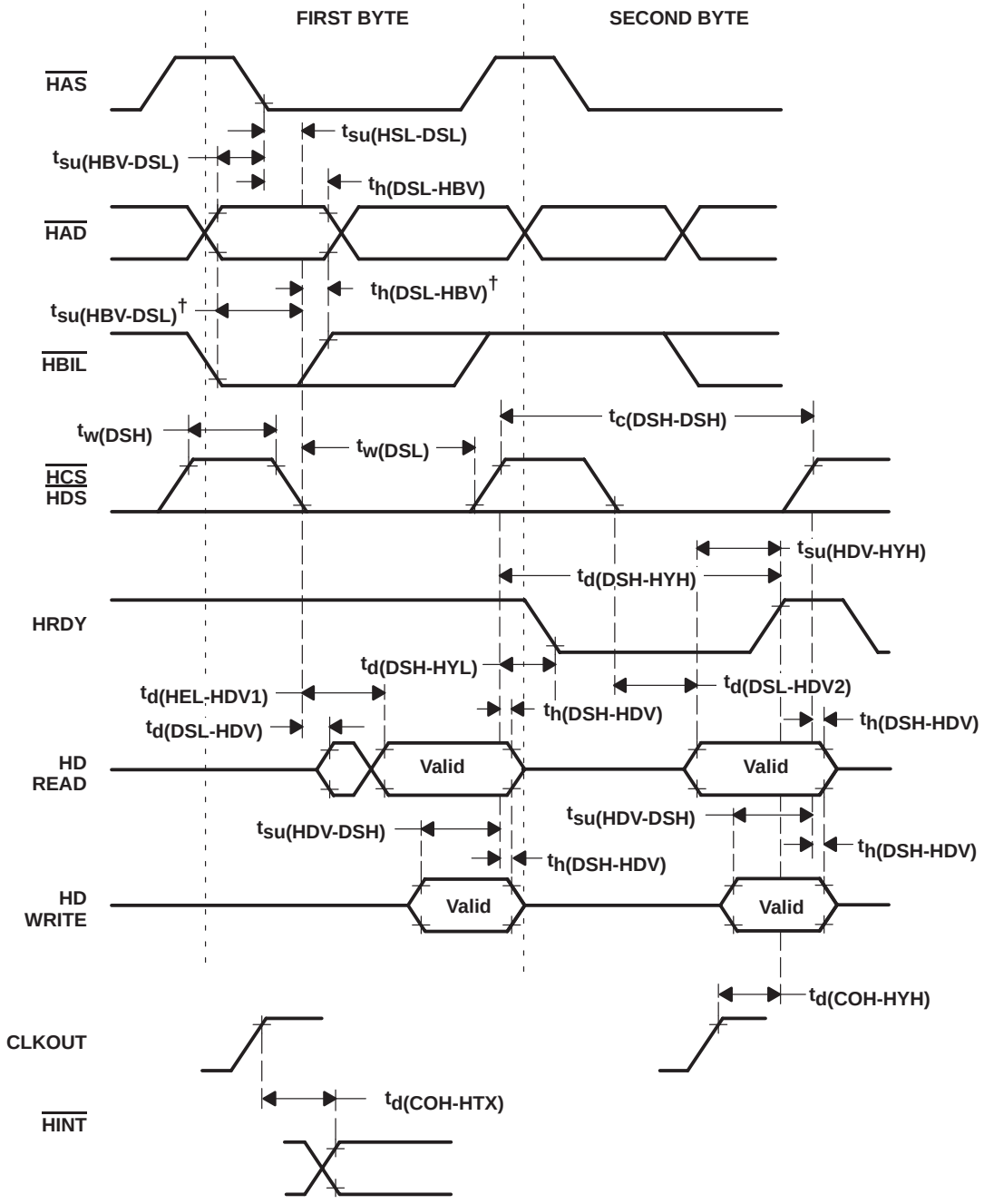


Figure 31. Read/Write Access Timings Using $\overline{\text{HAS}}$ Without HRDY

HOST PORT INTERFACE (TMS320C57S, TMS320LC57 ONLY) (CONTINUED)



† When $\overline{\text{HAS}}$ is tied to V_{DD}

Figure 32. Read/Write Access Timing With HRDY

HOST PORT INTERFACE (TMS320C57S, TMS320LC57 ONLY) (CONTINUED)

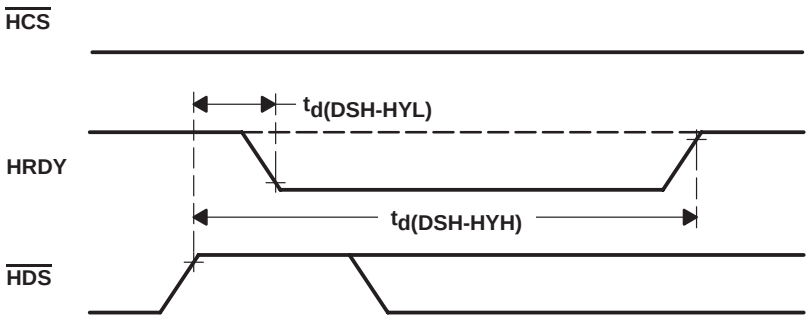
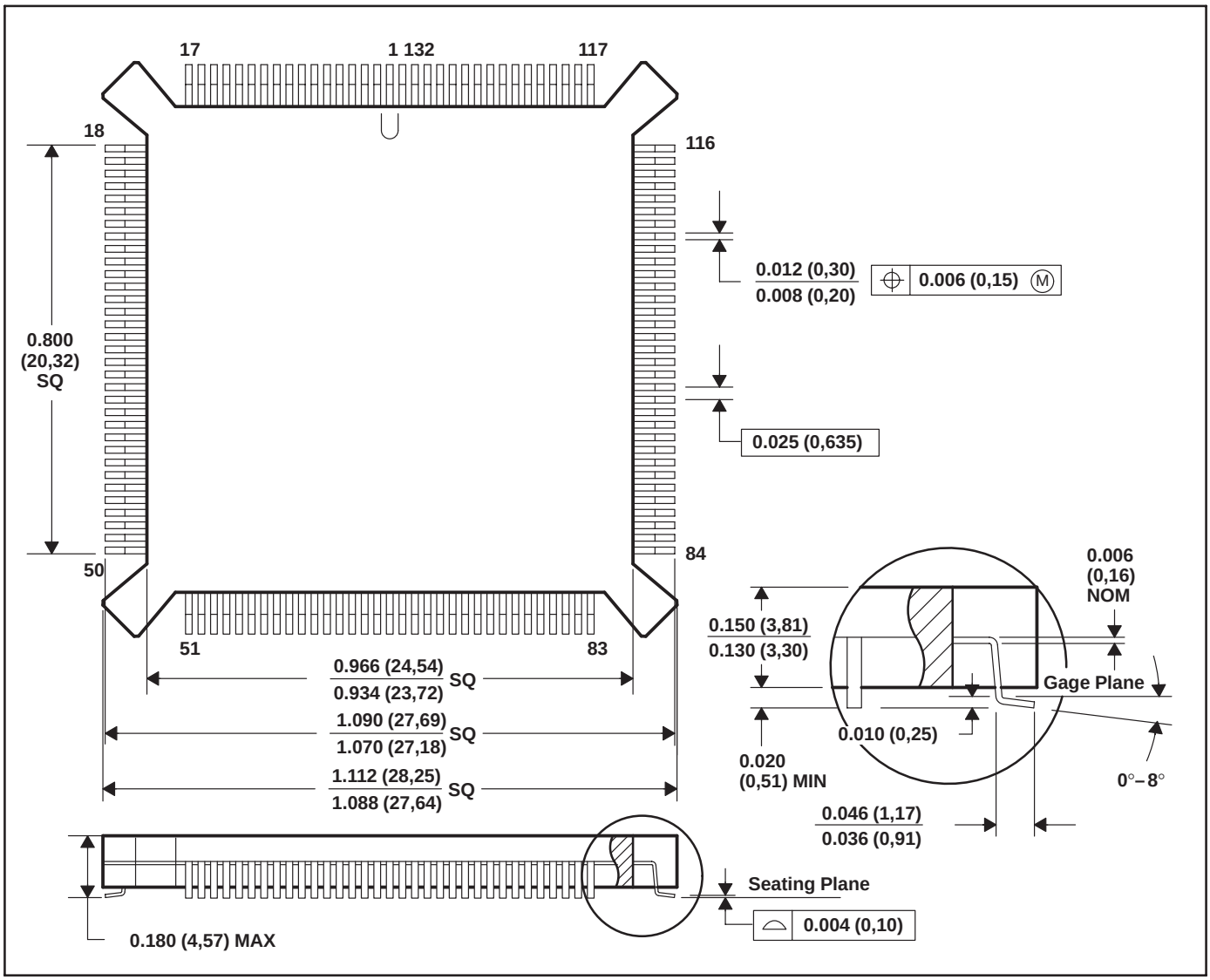


Figure 33. HRDY Signal When $\overline{\text{HCS}}$ Is Always Low

MECHANICAL DATA

PQ (S-PQFP-G132) **PLASTIC QUAD FLATPACK**



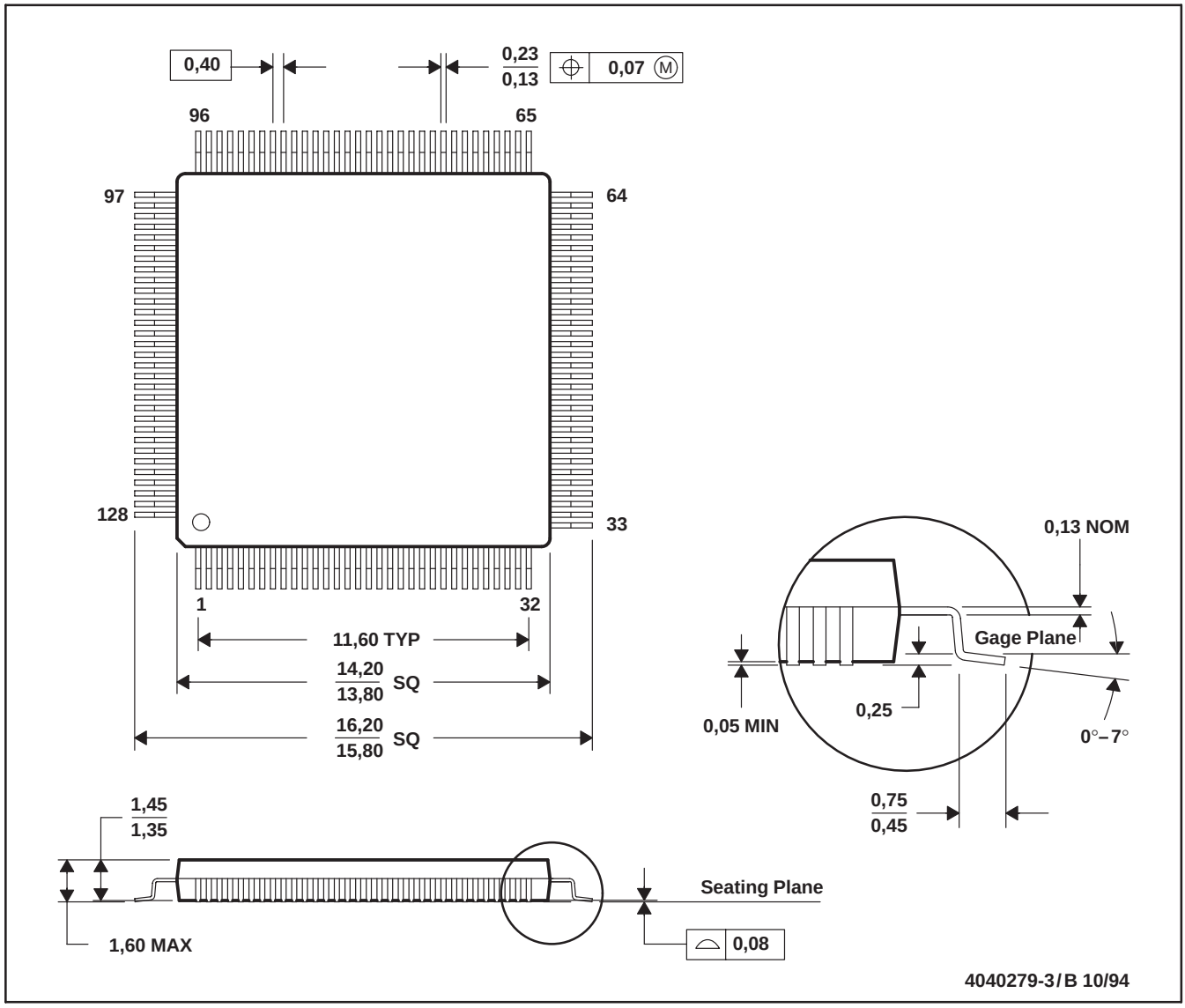
NOTES: A. All linear dimensions are in inches (millimeters).
B. This drawing is subject to change without notice.
C. Falls within JEDEC MO-069

Thermal Resistance Characteristics

PARAMETER	°C/W
$R_{\Theta JA}$	35
$R_{\Theta JC}$	8.5

MECHANICAL DATA

PBK/S-PQFP-G128 **PLASTIC QUAD FLATPACK**



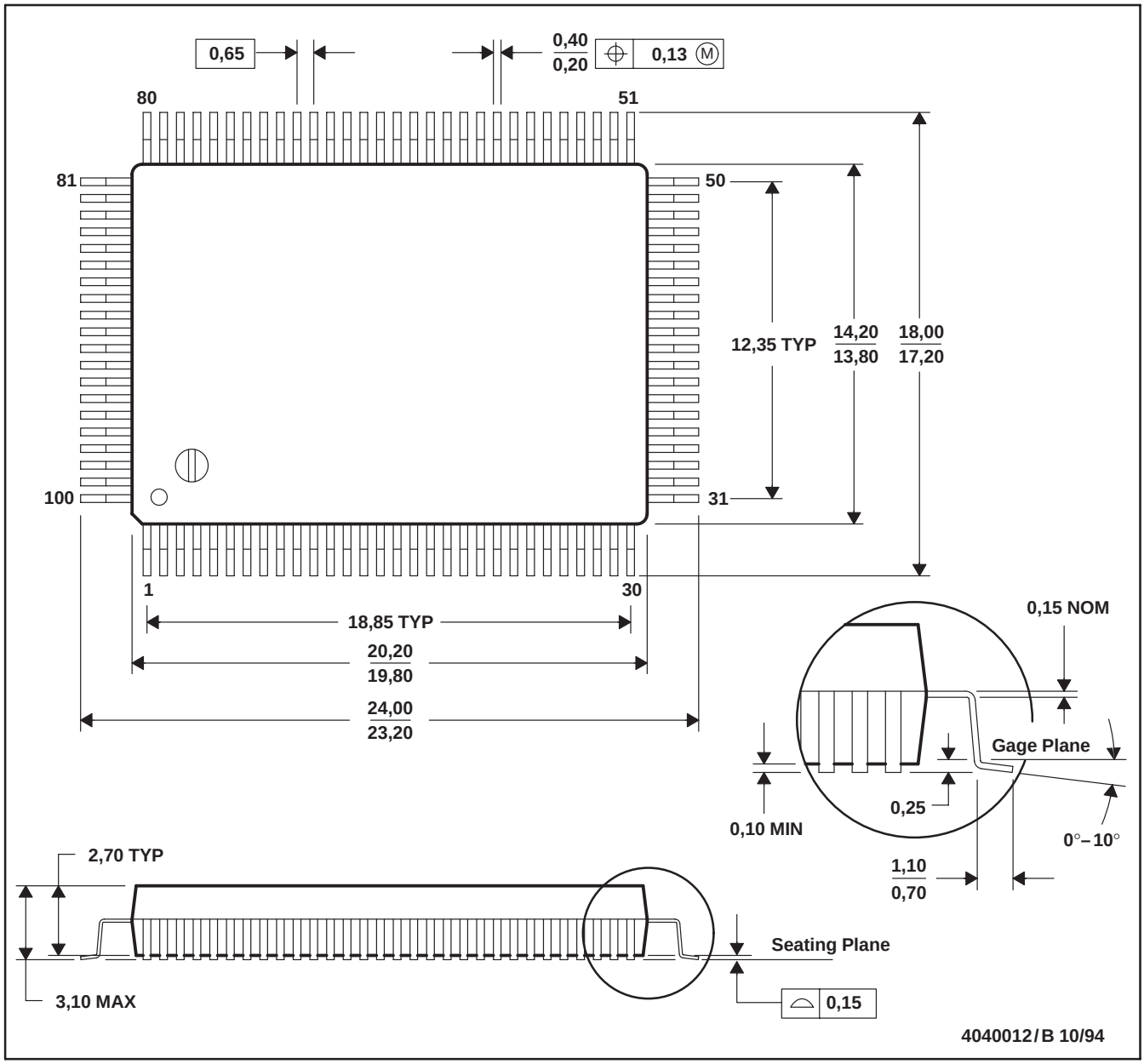
NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.

Thermal Resistance Characteristics

PARAMETER	°C/W
$R_{\Theta JA}$	58
$R_{\Theta JC}$	10

MECHANICAL DATA

PJ (R-PQFP-G100) **PLASTIC QUAD FLATPACK**



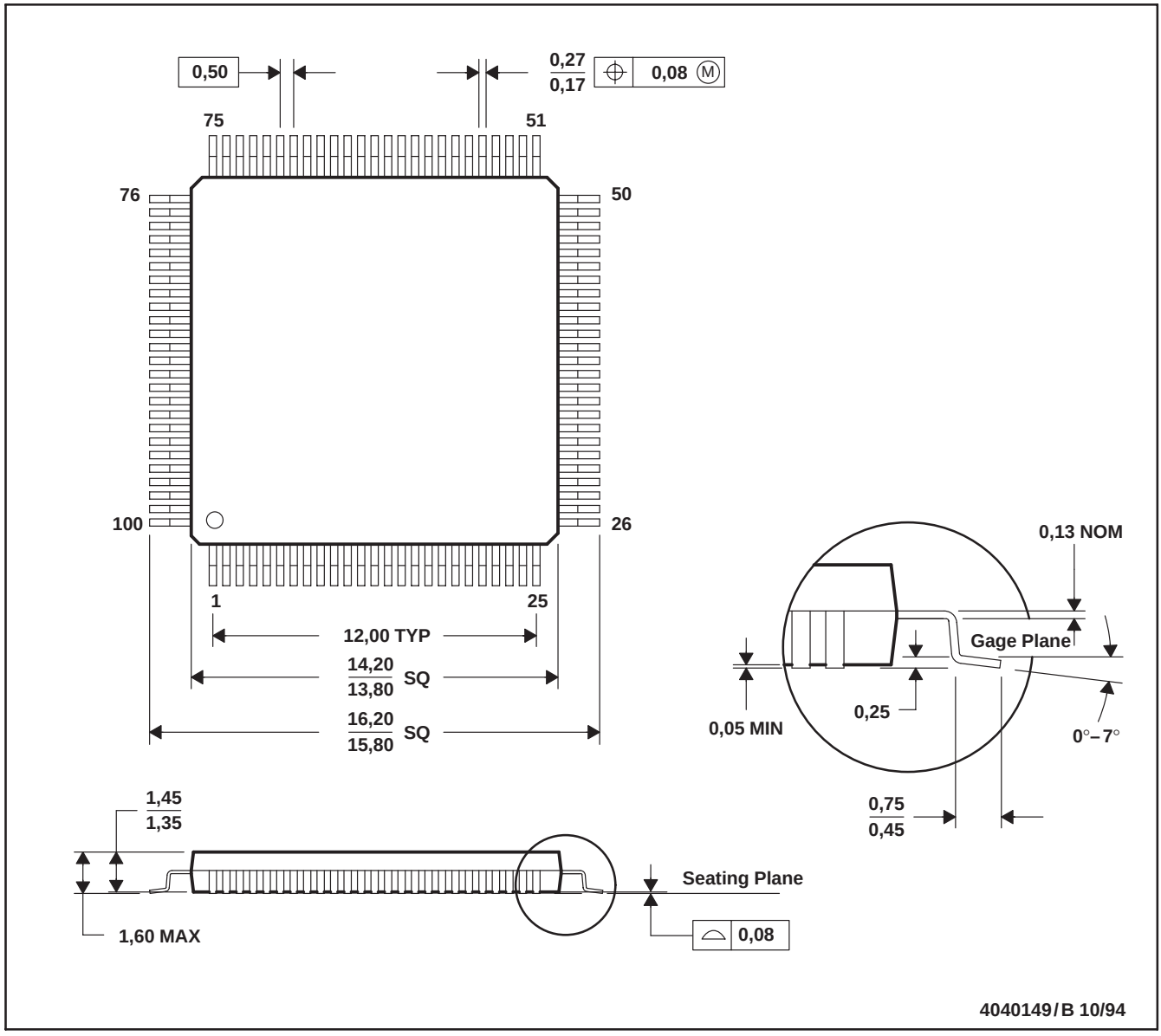
NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Contact field sales office to determine if a tighter coplanarity requirement is available for this package.

Thermal Resistance Characteristics

PARAMETER	°C/W
$R_{\Theta JA}$	78
$R_{\Theta JC}$	13

MECHANICAL DATA

PZ (S-PQFP-G100) **PLASTIC QUAD FLATPACK**



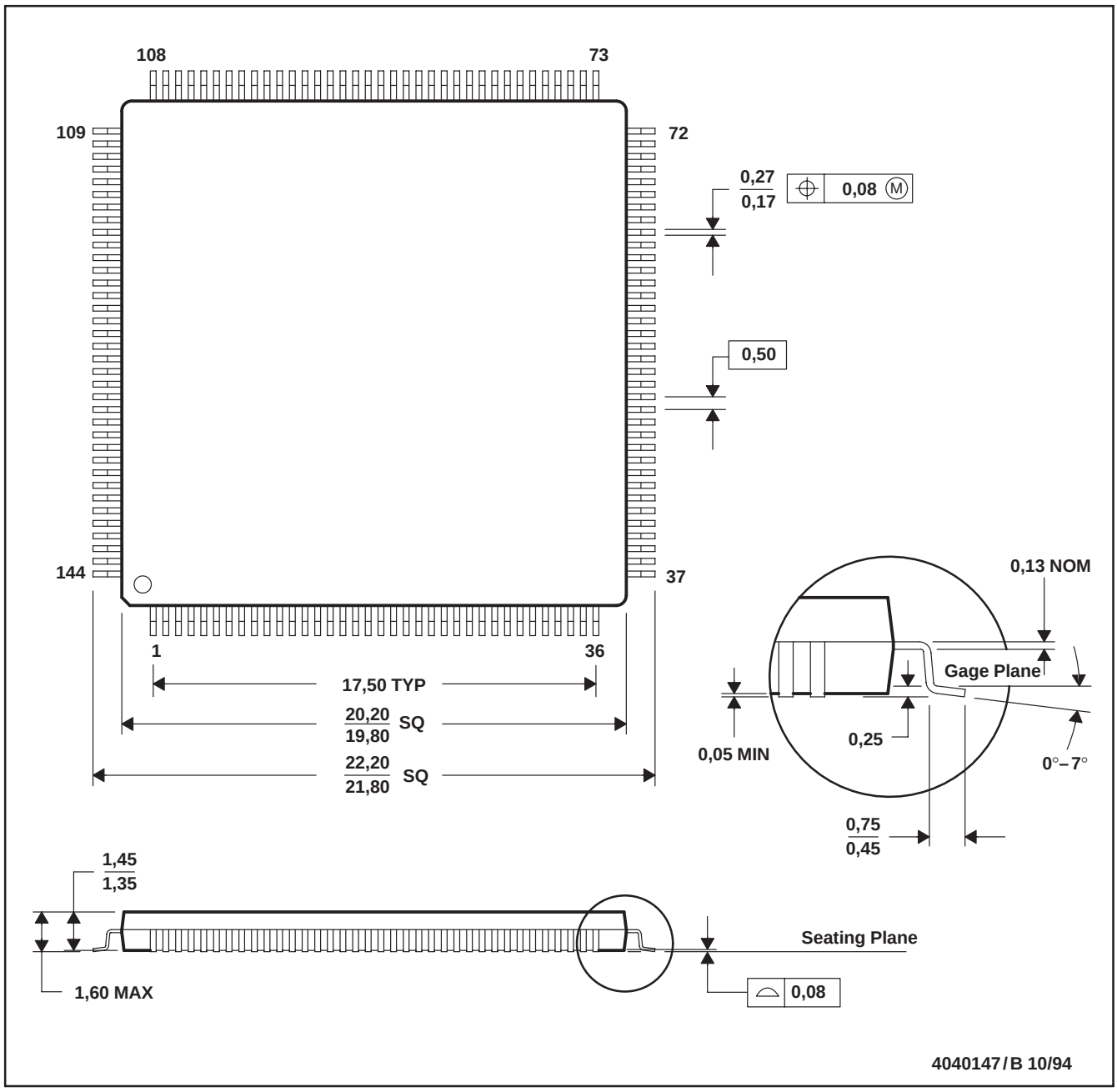
NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MO-136

Thermal Resistance Characteristics

PARAMETER	°C/W
$R_{\Theta JA}$	58
$R_{\Theta JC}$	10

MECHANICAL DATA

PGE (S-PQFP-G144) **PLASTIC QUAD FLATPACK**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Falls within JEDEC MO-136

Thermal Resistance Characteristics

PARAMETER	°C/W
$R_{\Theta JA}$	40
$R_{\Theta JC}$	9.9

IMPORTANT NOTICE

Texas Instruments and its subsidiaries (TI) reserve the right to make changes to their products or to discontinue any product or service without notice, and advise customers to obtain the latest version of relevant information to verify, before placing orders, that information being relied on is current and complete. All products are sold subject to the terms and conditions of sale supplied at the time of order acknowledgement, including those pertaining to warranty, patent infringement, and limitation of liability.

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

CERTAIN APPLICATIONS USING SEMICONDUCTOR PRODUCTS MAY INVOLVE POTENTIAL RISKS OF DEATH, PERSONAL INJURY, OR SEVERE PROPERTY OR ENVIRONMENTAL DAMAGE ("CRITICAL APPLICATIONS"). TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS. INCLUSION OF TI PRODUCTS IN SUCH APPLICATIONS IS UNDERSTOOD TO BE FULLY AT THE CUSTOMER'S RISK.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards must be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance or customer product design. TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used. TI's publication of information regarding any third party's products or services does not constitute TI's approval, warranty or endorsement thereof.

Copyright © 1998, Texas Instruments Incorporated