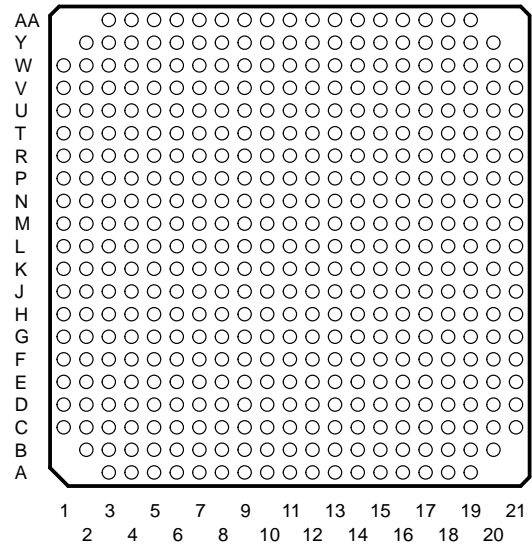


- **Highest Performance Fixed-Point Digital Signal Processor (DSP) SM/SMJ320C6201B**
  - 5-, 6.7-ns Instruction Cycle Time
  - 150 and 200-MHz Clock Rate
  - Eight 32-Bit Instructions/Cycle
  - 1200 and 1600 MIPS
- **VelociTI™ Advanced Very Long Instruction Word (VLIW) C62x™ CPU Core**
  - Eight Independent Functional Units:
    - Six ALUs (32-/40-Bit)
    - Two 16-Bit Multipliers (32-Bit Results)
  - Load-Store Architecture With 32 32-Bit General-Purpose Registers
  - Instruction Packing Reduces Code Size
  - All Instructions Conditional
- **Instruction Set Features**
  - Byte-Addressable (8-, 16-, 32-Bit Data)
  - 32-Bit Address Range
  - 8-Bit Overflow Protection
  - Saturation
  - Bit-Field Extract, Set, Clear
  - Bit-Counting
  - Normalization
- **1M-Bit On-Chip SRAM**
  - 512K-Bit Internal Program/Cache (16K 32-Bit Instructions)
  - 512K-Bit Dual-Access Internal Data (64K Bytes) Organized as Two Blocks for Improved Concurrency
- **32-Bit External Memory Interface (EMIF)**
  - Glueless Interface to Synchronous Memories: SDRAM and SBSRAM
  - Glueless Interface to Asynchronous Memories: SRAM and EPROM
- **Four-Channel Bootloading Direct-Memory-Access (DMA) Controller with an Auxiliary Channel**
- **16-Bit Host-Port Interface (HPI)**
  - Access to Entire Memory Map

**GLP**  
**429-PIN BALL GRID ARRAY (BGA) PACKAGE**  
**(BOTTOM VIEW)**



- **Two Multichannel Buffered Serial Ports (McBSPs)**
  - Direct Interface to T1/E1, MVIP, SCSA Framers
  - ST-Bus-Switching Compatible
  - Up to 256 Channels Each
  - AC97-Compatible
  - Serial Peripheral Interface (SPI) Compatible (Motorola™)
- **Two 32-Bit General-Purpose Timers**
- **Flexible Phase-Locked Loop (PLL) Clock Generator**
- **IEEE-1149.1 (JTAG†) Boundary-Scan Compatible**
- **429-Pin BGA Package (GLP Suffix)**
- **CMOS Technology**
  - 0.18-μm/5-Level Metal Process
- **3.3-V I/Os, 1.8-V Internal**



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.

C62x and VelociTI are trademarks of Texas Instruments Incorporated.

Motorola is a trademark of Motorola, Inc.

† IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

UNLESS OTHERWISE NOTED this document contains PRODUCTION DATA information 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.



POST OFFICE BOX 1443 • HOUSTON, TEXAS 77251-1443

Copyright © 2001, Texas Instruments Incorporated

On products compliant to MIL-PRF-38535, all parameters are tested unless otherwise noted. On all other products, production processing does not necessarily include testing of all parameters.

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## Table of Contents

description .....	2	parameter measurement information .....	30
device characteristics .....	3	signal transition levels .....	30
functional block diagram .....	3	input and output clocks .....	31
CPU description .....	4	asynchronous memory timing .....	34
signal groups description .....	7	synchronous-burst memory timing .....	36
signal descriptions .....	9	synchronous DRAM timing .....	40
development support .....	22	HOLD/HOLDA timing .....	44
device and development support nomenclature .....	23	reset timing .....	45
documentation support .....	24	external interrupt timing .....	47
clock PLL .....	26	host-port interface timing .....	48
power-supply sequencing .....	27	multichannel buffered serial port timing .....	51
absolute maximum ratings over operating case temperature ranges .....	28	DMAC, timer, power-down timing .....	59
recommended operating conditions .....	28	JTAG test-port timing .....	60
electrical characteristics over recommended ranges of supply voltage and operating case temperature .....	29	mechanical data .....	61

## description

The 320C6201B DSP is a member of the fixed-point DSP family in the 320C6000 platform. The SM/SMJ320C6201B (C6201B) device is based on the high-performance, advanced VelocityTI very-long-instruction-word (VLIW) architecture developed by Texas Instruments (TI™), making this DSP an excellent choice for multichannel and multifunction applications. With performance of up to 1600 million instructions per second (MIPS) at a clock rate of 200 MHz, the C6201B offers cost-effective solutions to high-performance DSP programming challenges. The C6201B is a newer revision of the C6201. The C6201B DSP possesses the operational flexibility of high-speed controllers and the numerical capability of array processors. This processor has 32 general-purpose registers of 32-bit word length and eight highly independent functional units. The eight functional units provide six arithmetic logic units (ALUs) for a high degree of parallelism and two 16-bit multipliers for a 32-bit result. The C6201B can produce two multiply-accumulates (MACs) per cycle—for a total of 400 million MACs per second (MMACS). The C6201B DSP also has application-specific hardware logic, on-chip memory, and additional on-chip peripherals.

The C6201B includes a large bank of on-chip memory and has a powerful and diverse set of peripherals. Program memory consists of a 64K-byte block that is user-configurable as cache or memory-mapped program space. Data memory of the C6201B consists of two 32K-byte blocks of RAM for improved concurrency. The peripheral set includes two multichannel buffered serial ports (McBSPs), two general-purpose timers, a host-port interface (HPI), and a glueless external memory interface (EMIF) capable of interfacing to SDRAM or SBSRAM and asynchronous peripherals.

The C6201B has a complete set of development tools which includes: a new C compiler, a third-party Ada 95 compiler, an assembly optimizer to simplify programming and scheduling, and a Windows™ debugger interface for visibility into source code execution.

TI is a trademark of Texas Instruments Incorporated.

Windows is a registered trademark of the Microsoft Corporation.



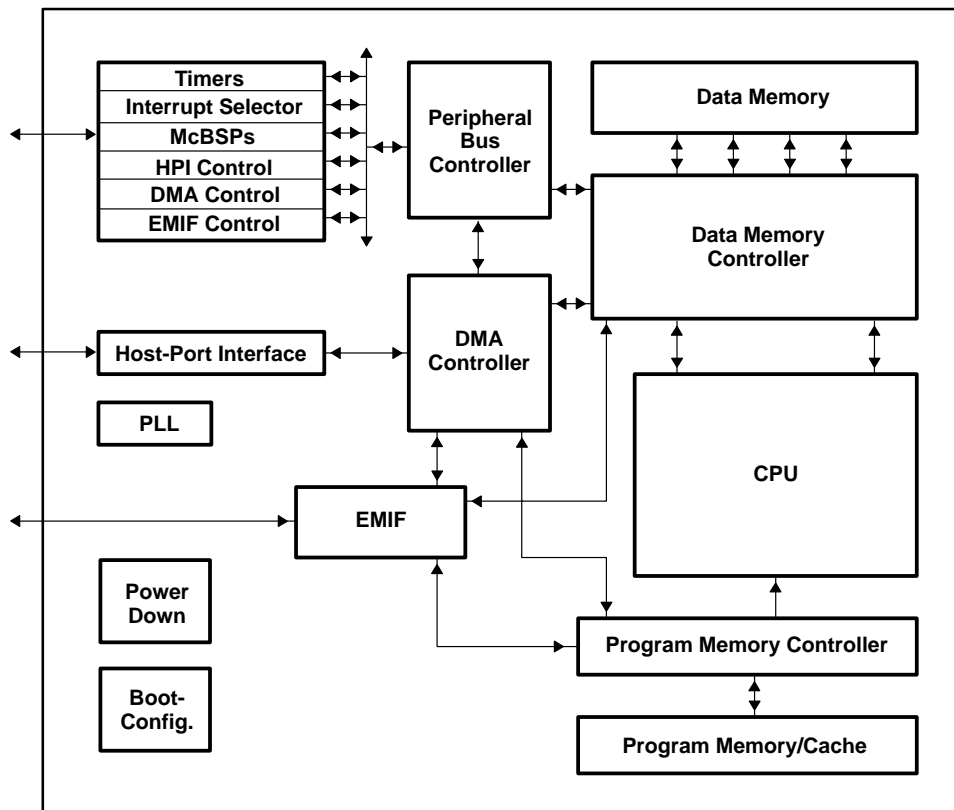
## device characteristics

Table 1 provides an overview of the C62x DSP. The table shows significant features of each device, including the capacity of on-chip RAM, the peripherals, the execution time, and the package type with pin count.

**Table 1. Characteristics of the C6201B Processor**

CHARACTERISTICS	DESCRIPTION
Device Number	320C6201B
On-Chip Memory	512-Kbit Program Memory 512-Kbit Data Memory (organized as two blocks)
Peripherals	2 Multichannel Buffered Serial Ports (McBSPs) 2 General-Purpose Timers Host-Port Interface (HPI) External Memory Interface (EMIF)
Cycle Time	6.7 ns (320C6201B 150 MHz), 5 ns (320C6201B 200 MHz)
Package Type	27 mm × 27 mm, 429-Pin Ceramic D-BGA (GLP)
Nominal Voltage	1.8 V Core 3.3 V I/O

## functional block diagram



### CPU description

The CPU fetches VelociTI advanced very-long instruction words (VLIW) (256 bits wide) to supply up to eight 32-bit instructions to the eight functional units during every clock cycle. The VelociTI VLIW architecture features controls by which all eight units do not have to be supplied with instructions if they are not ready to execute. The first bit of every 32-bit instruction determines if the next instruction belongs to the same execute packet as the previous instruction, or whether it should be executed in the following clock as a part of the next execute packet. Fetch packets are always 256 bits wide; however, the execute packets can vary in size. The variable-length execute packets are a key memory-saving feature, distinguishing the C62x CPU from other VLIW architectures.

The CPU features two sets of functional units. Each set contains four units and a register file. One set contains functional units .L1, .S1, .M1, and .D1; the other set contains units .D2, .M2, .S2, and .L2. The two register files each contain 16 32-bit registers for a total of 32 general-purpose registers. The two sets of functional units, along with two register files, compose sides A and B of the CPU (see Figure 1 and Figure 2). The four functional units on each side of the CPU can freely share the 16 registers belonging to that side. Additionally, each side features a single data bus connected to all the registers on the other side, by which the two sets of functional units can access data from the register files on the opposite side. While register access by functional units on the same side of the CPU as the register file can service all the units in a single clock cycle, register access using the register file across the CPU supports one read and one write per cycle.

Another key feature of the C62x CPU is the load/store architecture, where all instructions operate on registers (as opposed to data in memory). Two sets of data-addressing units (.D1 and .D2) are responsible for all data transfers between the register files and the memory. The data address driven by the .D units allows data addresses generated from one register file to be used to load or store data to or from the other register file. The C62x CPU supports a variety of indirect addressing modes using either linear- or circular-addressing modes with 5- or 15-bit offsets. All instructions are conditional, and most can access any one of the 32 registers. Some registers, however, are singled out to support specific addressing or to hold the condition for conditional instructions (if the condition is not automatically “true”). The two .M functional units are dedicated for multiplies. The two .S and .L functional units perform a general set of arithmetic, logical, and branch functions with results available every clock cycle.

The processing flow begins when a 256-bit-wide instruction fetch packet is fetched from a program memory. The 32-bit instructions destined for the individual functional units are “linked” together by “1” bits in the least significant bit (LSB) position of the instructions. The instructions that are “chained” together for simultaneous execution (up to eight in total) compose an execute packet. A “0” in the LSB of an instruction breaks the chain, effectively placing the instructions that follow it in the next execute packet. If an execute packet crosses the fetch packet boundary (256 bits wide), the assembler places it in the next fetch packet, while the remainder of the current fetch packet is padded with NOP instructions. The number of execute packets within a fetch packet can vary from one to eight. Execute packets are dispatched to their respective functional units at the rate of one per clock cycle and the next 256-bit fetch packet is not fetched until all the execute packets from the current fetch packet have been dispatched. After decoding, the instructions simultaneously drive all active functional units for a maximum execution rate of eight instructions every clock cycle. While most results are stored in 32-bit registers, they can be subsequently moved to memory as bytes or half-words as well. All load and store instructions are byte-, half-word, or word-addressable.

CPU description (continued)

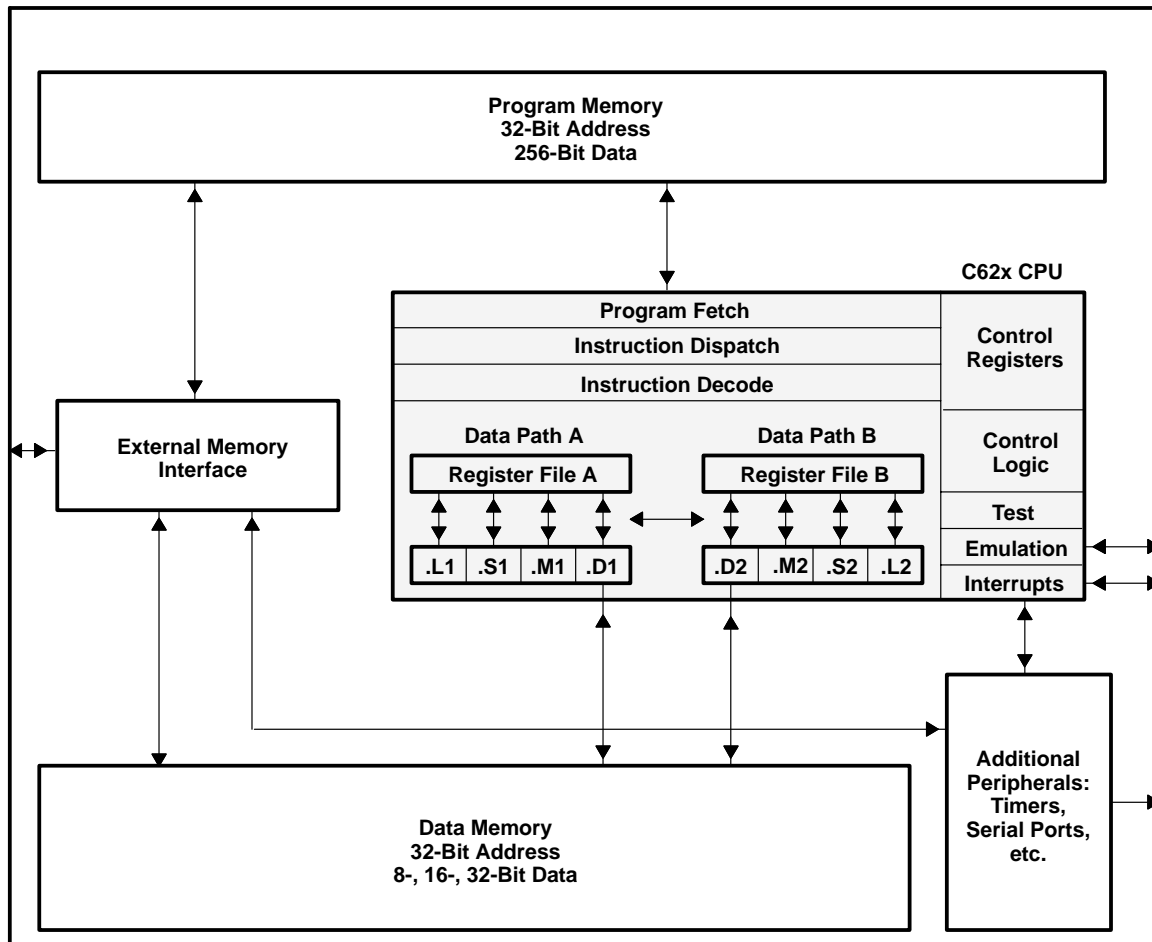


Figure 1. 320C62x CPU Block Diagram

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## CPU description (continued)

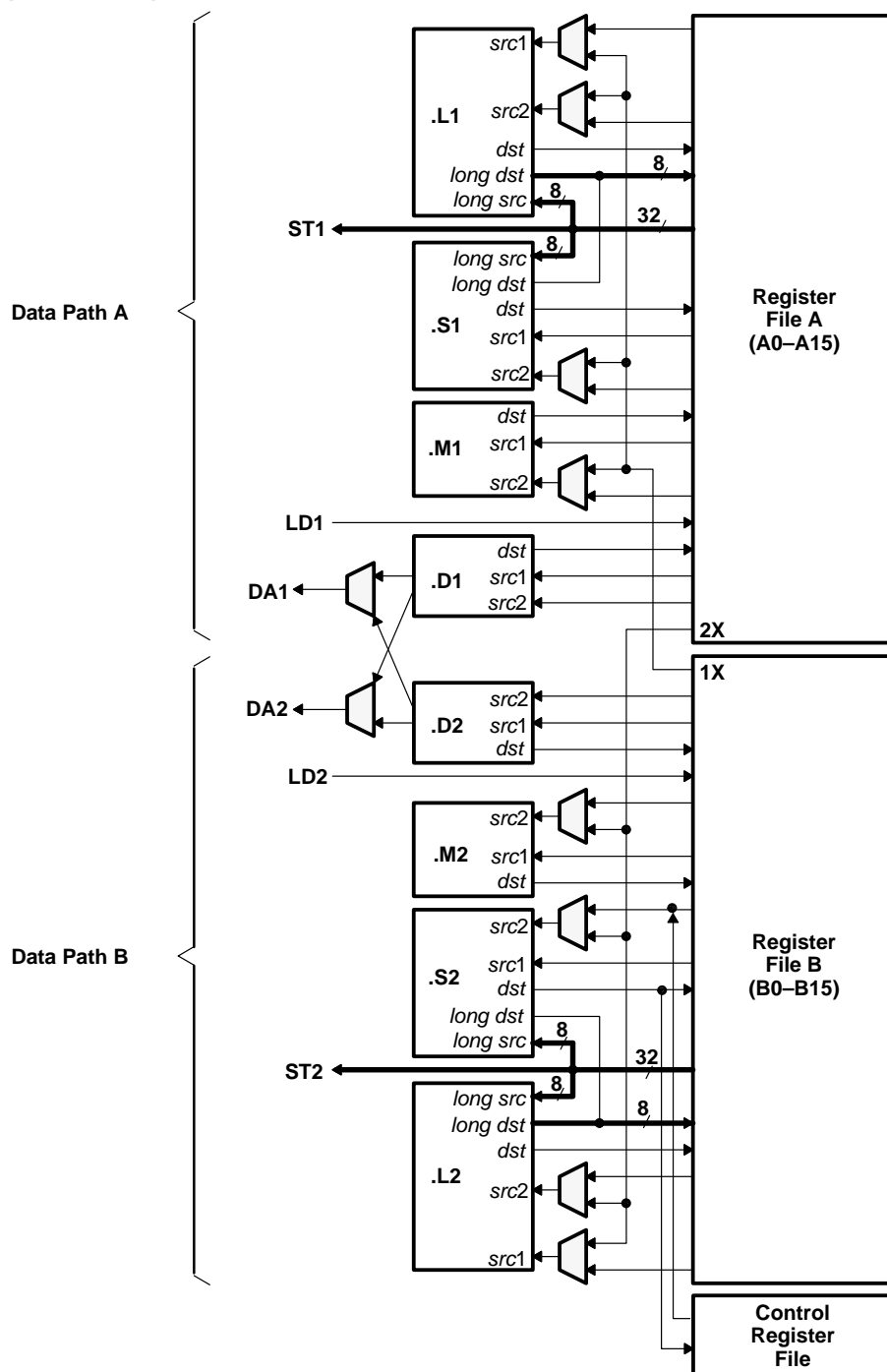


Figure 2. 320C62x CPU Data Paths

## signal groups description

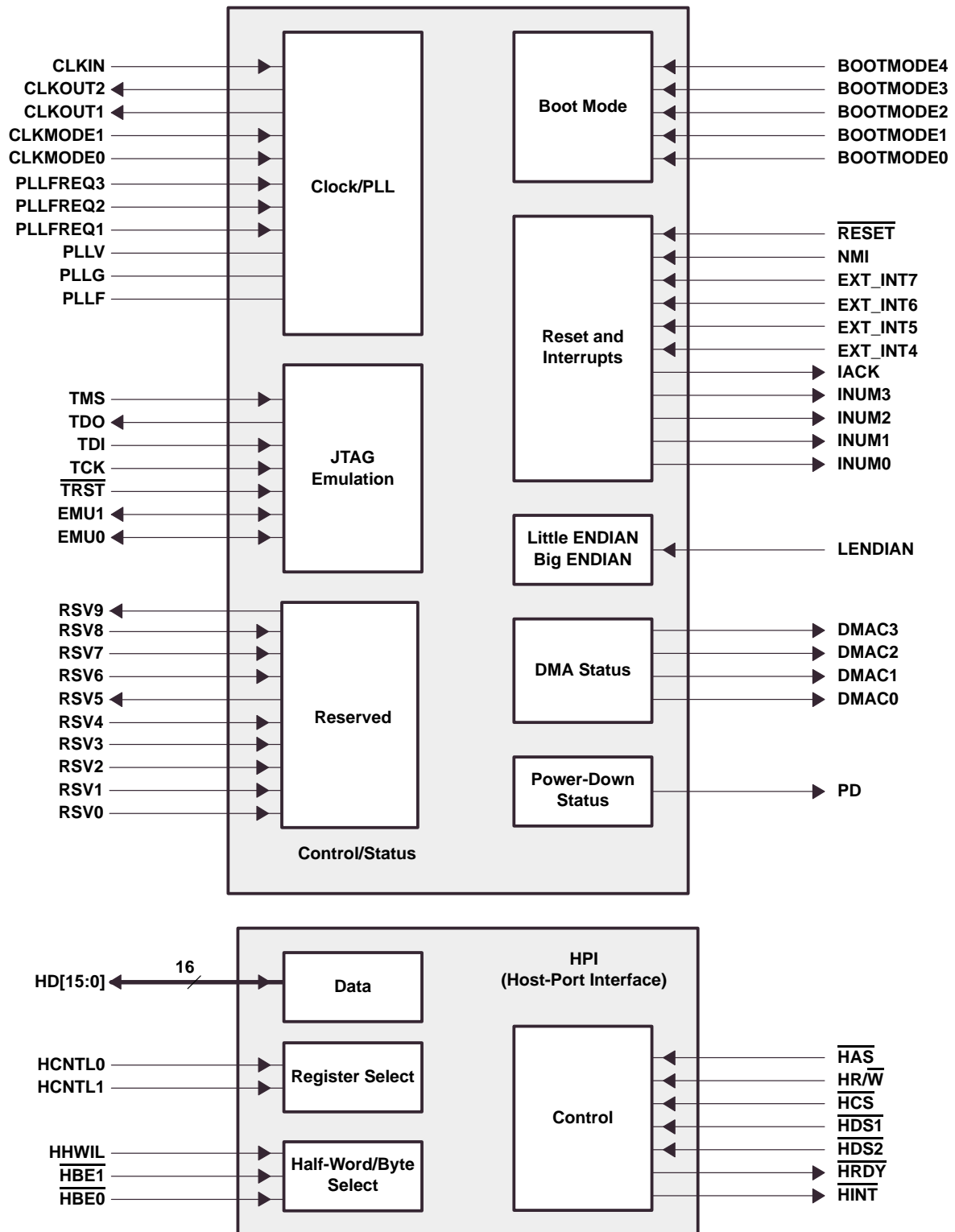


Figure 3. CPU and Peripheral Signals

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## signal groups description (continued)

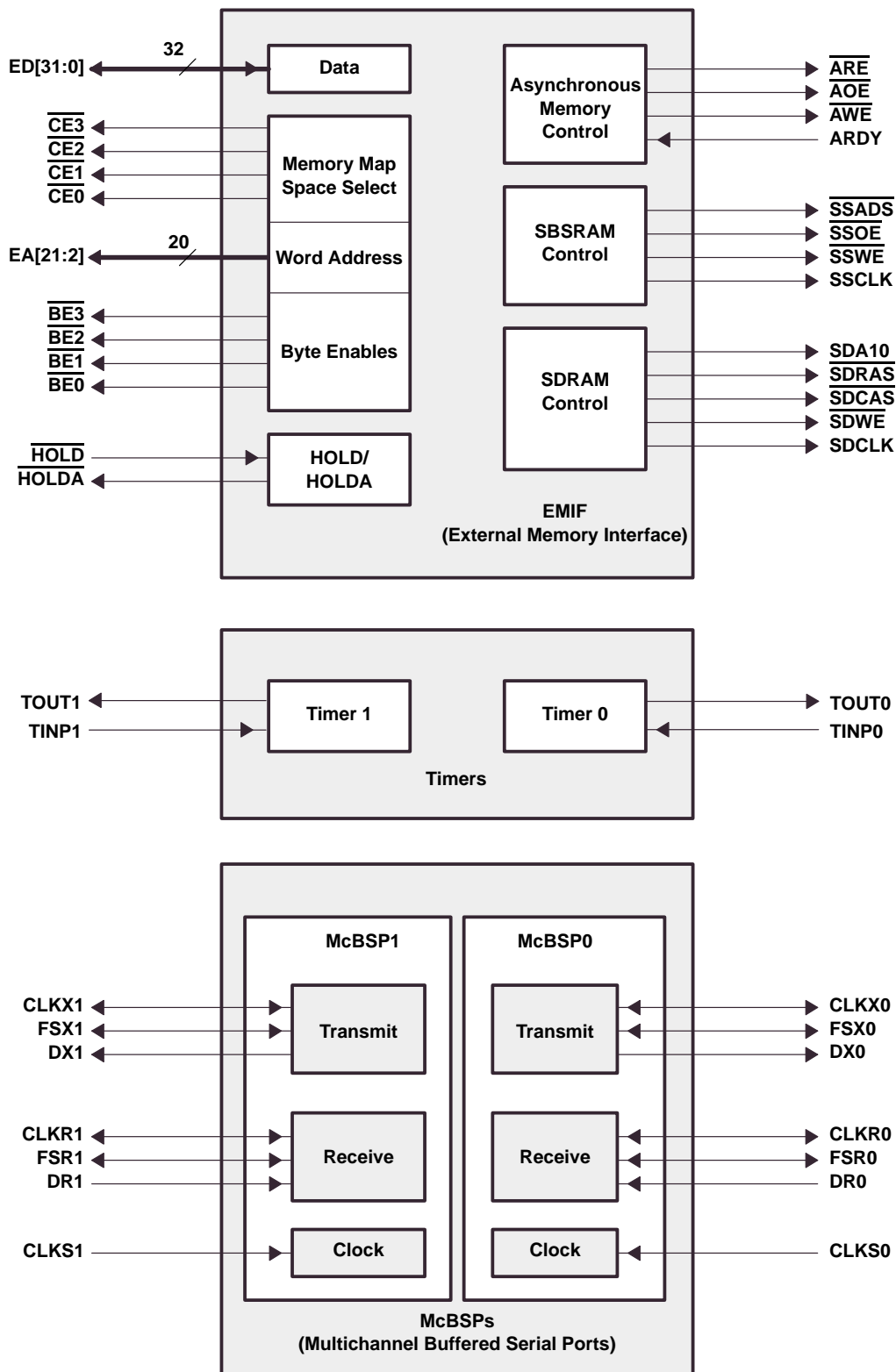


Figure 4. Peripheral Signals



**Signal Descriptions**

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
<b>CLOCK/PLL</b>			
CLKIN	A14	I	Clock Input
CLKOUT1	Y6	O	Clock output at full device speed
CLKOUT2	V9	O	Clock output at half of device speed
CLKMODE1	B17	I	Clock mode select
CLKMODE0	C17		
			• Selects whether the output clock frequency = input clock freq x4 or x1
PLLFREQ3	C13	I	PLL frequency range (3, 2, and 1)
PLLFREQ2	G11		
PLLFREQ1	F11		
			• The target range for CLKOUT1 frequency is determined by the 3-bit value of the PLLFREQ pins.
PLL $\nabla$	D12	A§	PLL analog $V_{CC}$ connection for the low-pass filter
PLL $\nabla$	G10	A§	PLL analog GND connection for the low-pass filter
PLLF	C12	A§	PLL low-pass filter connection to external components and a bypass capacitor
<b>JTAG EMULATION</b>			
TMS	K19	I	JTAG test port mode select (features an internal pull-up)
TDO	R12	O/Z	JTAG test port data out
TDI	R13	I	JTAG test port data in (features an internal pull-up)
TCK	M20	I	JTAG test port clock
TRST	N18	I	JTAG test port reset (features an internal pull-down)
EMU1	R20	I/O/Z	Emulation pin 1, pull-up with a dedicated 20-k $\Omega$ resistor
EMU0	T18	I/O/Z	Emulation pin 0, pull-up with a dedicated 20-k $\Omega$ resistor
<b>RESET AND INTERRUPTS</b>			
RESET	J20	I	Device reset
NMI	K21	I	Nonmaskable interrupt
			• Edge-driven (rising edge)
EXT_INT7	R16	I	External interrupts
EXT_INT6	P20		
EXT_INT5	R15		
EXT_INT4	R18		
			• Edge-driven (rising edge)
IACK	R11	O	Interrupt acknowledge for all active interrupts serviced by the CPU
INUM3	T19	O	Active interrupt identification number
INUM2	T20		
INUM1	T14		
INUM0	T16		
			• Valid during IACK for all active interrupts (not just external)
			• Encoding order follows the interrupt service fetch packet ordering
<b>LITTLE ENDIAN/BIG ENDIAN</b>			
LENDIAN	G20	I	If high, selects little-endian byte/half-word addressing order within a word If low, selects big-endian addressing
<b>POWER DOWN STATUS</b>			
PD	D19	O	Power-down mode 2 or 3 (active if high)

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

‡ PLL $\nabla$  and PLL $\nabla$  signals are not part of external voltage supply or ground. See the CLOCK/PLL documentation for information on how to connect those pins.

§ A = Analog Signal (PLL Filter)

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## Signal Descriptions (Continued)

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
<b>HOST PORT INTERFACE (HPI)</b>			
$\overline{\text{HINT}}$	H2	O/Z	Host interrupt (from DSP to host)
HCNTL1	J6	I	Host control – selects between control, address or data registers
HCNTL0	H6	I	Host control – selects between control, address or data registers
HHWIL	E4	I	Host halfword select – first or second halfword (not necessarily high or low order)
$\overline{\text{HBE1}}$	G6	I	Host byte select within word or half-word
$\overline{\text{HBE0}}$	F6	I	Host byte select within word or half-word
HR/W	D4	I	Host read or write select
HD15	D11	I/O/Z	Host port data (used for transfer of data, address and control)
HD14	B11		
HD13	A11		
HD12	G9		
HD11	D10		
HD10	A10		
HD9	C10		
HD8	B9		
HD7	F9		
HD6	C9		
HD5	A9		
HD4	B8		
HD3	D9		
HD2	D8		
HD1	B7		
HD0	C7		
$\overline{\text{HAS}}$	L6	I	Host address strobe
$\overline{\text{HCS}}$	C5	I	Host chip select
$\overline{\text{HDS1}}$	C4	I	Host data strobe 1
$\overline{\text{HDS2}}$	K6	I	Host data strobe 2
$\overline{\text{HRDY}}$	H3	O	Host ready (from DSP to host)
<b>BOOT MODE</b>			
BOOTMODE4	B16	I	Boot mode
BOOTMODE3	G14		
BOOTMODE2	F15		
BOOTMODE1	C18		
BOOTMODE0	D17		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground



Signal Descriptions (Continued)

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
EMIF – CONTROL SIGNALS COMMON TO ALL TYPES OF MEMORY			
$\overline{\text{CE3}}$	Y5	O/Z	Memory space enables <ul style="list-style-type: none"><li>• Enabled by bits 24 and 25 of the word address</li><li>• Only one asserted during any external data access</li></ul>
$\overline{\text{CE2}}$	V3	O/Z	
$\overline{\text{CE1}}$	T6	O/Z	
$\overline{\text{CE0}}$	U2	O/Z	
$\overline{\text{BE3}}$	R8	O/Z	Byte enable control <ul style="list-style-type: none"><li>• Decoded from the two lowest bits of the internal address</li><li>• Byte write enables for most types of memory</li><li>• Can be directly connected to SDRAM read and write mask signal (SDQM)</li></ul>
$\overline{\text{BE2}}$	T3	O/Z	
$\overline{\text{BE1}}$	T2	O/Z	
$\overline{\text{BE0}}$	R2	O/Z	
EMIF – ADDRESS			
EA21	L4	O/Z	External address (word address)
EA20	L3		
EA19	J2		
EA18	J1		
EA17	K1		
EA16	K2		
EA15	L2		
EA14	L1		
EA13	M1		
EA12	M2		
EA11	M6		
EA10	N4		
EA9	N1		
EA8	N2		
EA7	N6		
EA6	P4		
EA5	P3		
EA4	P2		
EA3	P1		
EA2	P6		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## Signal Descriptions (Continued)

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
EMIF – DATA			
ED31	U18	I/O/Z	External data
ED30	U20		
ED29	T15		
ED28	V18		
ED27	V17		
ED26	V16		
ED25	T12		
ED24	W17		
ED23	T13		
ED22	Y17		
ED21	T11		
ED20	Y16		
ED19	W15		
ED18	V14		
ED17	Y15		
ED16	R9		
ED15	Y14		
ED14	V13		
ED13	AA13		
ED12	T10		
ED11	Y13		
ED10	W12		
ED9	Y12		
ED8	Y11		
ED7	V10		
ED6	AA10		
ED5	Y10		
ED4	W10		
ED3	Y9		
ED2	AA9		
ED1	Y8		
ED0	W9		
EMIF – ASYNCHRONOUS MEMORY CONTROL			
$\overline{\text{ARE}}$	R7	O/Z	Asynchronous memory read enable
$\overline{\text{AOE}}$	T7	O/Z	Asynchronous memory output enable
$\overline{\text{AWE}}$	V5	O/Z	Asynchronous memory write enable
ARDY	R4	I	Asynchronous memory ready input

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground



**Signal Descriptions (Continued)**

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
EMIF – SYNCHRONOUS BURST SRAM CONTROL			
$\overline{\text{SSADS}}$	V8	O/Z	SBSRAM address strobe
$\overline{\text{SSOE}}$	W7	O/Z	SBSRAM output enable
$\overline{\text{SSWE}}$	Y7	O/Z	SBSRAM write enable
SSCLK	AA8	O/Z	SBSRAM clock
EMIF – SYNCHRONOUS DRAM CONTROL			
SDA10	V7	O/Z	SDRAM address 10 (separate for deactivate command)
$\overline{\text{SDRAS}}$	V6	O/Z	SDRAM row address strobe
$\overline{\text{SDCAS}}$	W5	O/Z	SDRAM column address strobe
$\overline{\text{SDWE}}$	T8	O/Z	SDRAM write enable
SDCLK	T9	O/Z	SDRAM clock
EMIF – BUS ARBITRATION			
$\overline{\text{HOLD}}$	R6	I	Hold request from the host
$\overline{\text{HOLDA}}$	B15	O	Hold request acknowledge to the host
TIMERS			
TOUT1	G2	O	Timer 1 or general-purpose output
TINP1	K3	I	Timer 1 or general-purpose input
TOUT0	M18	O	Timer 0 or general-purpose output
TINP0	J18	I	Timer 0 or general-purpose input
DMA ACTION COMPLETE			
DMAC3	E18	O	DMA action complete
DMAC2	F19		
DMAC1	E20		
DMAC0	G16		
MULTICHANNEL BUFFERED SERIAL PORT 1 (McBSP1)			
CLKS1	F4	I	External clock source (as opposed to internal)
CLKR1	H4	I/O/Z	Receive clock
CLKX1	J4	I/O/Z	Transmit clock
DR1	E2	I	Receive data
DX1	G4	O/Z	Transmit data
FSR1	F3	I/O/Z	Receive frame sync
FSX1	F2	I/O/Z	Transmit frame sync

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

# SM320C6201B, SMJ320C6201B

## DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

### Signal Descriptions (Continued)

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
<b>MULTICHANNEL BUFFERED SERIAL PORT 0 (McBSP0)</b>			
CLKS0	K18	I	External clock source (as opposed to internal)
CLKR0	L21	I/O/Z	Receive clock
CLKX0	K20	I/O/Z	Transmit clock
DR0	J21	I	Receive data
DX0	M21	O/Z	Transmit data
FSR0	P16	I/O/Z	Receive frame sync
FSX0	N16	I/O/Z	Transmit frame sync
<b>RESERVED FOR TEST</b>			
RSV0	N21	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV1	K16	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV2	B13	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV3	B14	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV4	F13	I	Reserved for testing, <b>pull-down</b> with a dedicated 20-kΩ resistor
RSV5	C15	O	Reserved (leave unconnected, <b>do not</b> connect to power or ground)
RSV6	F7	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV7	D7	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV8	B5	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV9	F16	O	Reserved (leave unconnected, <b>do not</b> connect to power or ground)
<b>SUPPLY VOLTAGE PINS</b>			
DVDD	C14	S	3.3-V supply voltage
	C8		
	E19		
	E3		
	H11		
	H13		
	H9		
	J10		
	J12		
	J14		
	J19		
	J3		
	J8		
	K11		
	K13		
	K15		
	K7		
	K9		
	L10		
	L12		
	L14		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground



**Signal Descriptions (Continued)**

SIGNAL NAME NO.		TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
DV <sub>DD</sub>	L8	S	3.3-V supply voltage
	M11		
	M13		
	M15		
	M7		
	M9		
	N10		
	N12		
	N14		
	N19		
	N3		
	N8		
	P11		
	P13		
	P9		
	U19		
	U3		
	W14		
	W8		
CV <sub>DD</sub>	A12	S	1.8-V supply voltage
	A13		
	B10		
	B12		
	B6		
	D15		
	D16		
	F10		
	F14		
	F8		
	G13		
	G7		
	G8		
	K4		
	M3		
	M4		
	A3		
	A5		
	A7		
	A16		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

SM320C6201B, SMJ320C6201B  
DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

Signal Descriptions (Continued)

SIGNAL NAME NO.		TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
CV <sub>DD</sub>	A18	S	1.8-V supply voltage
	AA4		
	AA6		
	AA15		
	AA17		
	AA19		
	B2		
	B4		
	B19		
	C1		
	C3		
	C20		
	D2		
	D21		
	E1		
	E6		
	E8		
	E10		
	E12		
	E14		
	E16		
	F5		
	F17		
	F21		
	G1		
	H5		
	H17		
	K5		
	K17		
	M5		
	M17		
	P5		
	P17		
	R21		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground



Signal Descriptions (Continued)

SIGNAL NAME	NO.	TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
CV <sub>DD</sub>	T1	S	1.8-V supply voltage
	T5		
	T17		
	U6		
	U8		
	U10		
	U12		
	U14		
	U16		
	U21		
	V1		
	V20		
	W2		
	W19		
	W21		
	Y3		
	Y18		
	Y20		
	AA11		
	AA12		
	F20		
	G18		
	H16		
	H18		
	L18		
	L19		
	L20		
	N20		
	P18		
	P19		
	R10		
	R14		
	U4		
	V11		
	V12		
	V15		
	W13		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

SM320C6201B, SMJ320C6201B  
DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

Signal Descriptions (Continued)

SIGNAL NAME NO.		TYPE†	DESCRIPTION
GROUND PINS			
VSS	C11	GND	Ground pins
	C16		
	C6		
	D5		
	G3		
	H10		
	H12		
	H14		
	H7		
	H8		
	J11		
	J13		
	J7		
	J9		
	K8		
	L7		
	L9		
	M8		
	N7		
	R3		
	A4		
	A6		
	A8		
	A15		
	A17		
	A19		
	AA3		
	AA5		
	AA7		
	AA14		
	AA16		
	AA18		
	B3		
	B18		
	B20		
	C2		
	C19		
	C21		
	D1		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

**Signal Descriptions (Continued)**

SIGNAL NAME NO.		TYPE†	DESCRIPTION
GROUND PINS (CONTINUED)			
VSS	D20	GND	Ground pins
	E5		
	E7		
	E9		
	E11		
	E13		
	E15		
	E17		
	E21		
	F1		
	G5		
	G17		
	G21		
	H1		
	J5		
	J17		
	L5		
	L17		
	N5		
	N17		
	P21		
	R1		
	R5		
	R17		
	T21		
	U1		
	U5		
	U7		
	U9		
	U11		
	U13		
	U15		
	U17		
	V2		
	V21		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## Signal Descriptions (Continued)

SIGNAL NAME NO.		TYPE†	DESCRIPTION
GROUND PINS (CONTINUED)			
VSS	W1	GND	Ground pins
	W3		
	W20		
	Y2		
	Y4		
	Y19		
	F18		
	G19		
	H15		
	J15		
	J16		
	K10		
	K12		
	K14		
	L11		
	L13		
	L15		
	M10		
	M12		
	M14		
	N11		
	N13		
	N15		
	N9		
	P10		
	P12		
	P14		
	P15		
	P7		
	P8		
	R19		
	T4		
	W11		
	W16		
	W6		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

**Signal Descriptions (Continued)**

SIGNAL NAME NO.		TYPE†	DESCRIPTION
REMAINING UNCONNECTED PINS			
NC	D13		Unconnected pins
	D14		
	D18		
	D3		
	D6		
	F12		
	G12		
	G15		
	H19		
	H20		
	H21		
	L16		
	M16		
	M19		
	V19		
	V4		
	W18		
	W4		

† I = Input, O = Output, Z = High Impedance, S = Supply Voltage, GND = Ground

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## development support

Texas Instruments offers an extensive line of development tools for the C6000 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 C6000-based applications:

### Software Development Tools:

Assembly optimizer  
Assembler/Linker  
Simulator  
Optimizing ANSI C compiler  
Application algorithms  
C/Assembly debugger and code profiler

### Hardware Development Tools:

Extended development system (XDS™) emulator (supports C6000 multiprocessor system debug)  
EVM (Evaluation Module)

The *TMS320 DSP Development Support Reference Guide* (SPRU011) contains information about development-support products for all TMS320 family member devices, including documentation. See this document for further information on TMS320 documentation or any TMS320 support products from Texas Instruments. An additional document, the *TMS320 Third-Party Support Reference Guide* (SPRU052), contains information about TMS320-related products from other companies in the industry. To receive TMS320 literature, contact the Product Information Center at (800) 477-8924.

See Table 2 for a complete listing of development-support tools for the C6000. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

**Table 2. 320C6000 Development-Support Tools**

DEVELOPMENT TOOL	PLATFORM	PART NUMBER
<b>Software</b>		
Ada 95 Compiler†	Sun Solaris 2.3™‡	AD0345AS8500RF - Single User AD0345BS8500RF - Multi-user
C Compiler/Assembler/Linker/Assembly Optimizer	Win32™	TMDX3246855-07
C Compiler/Assembler/Linker/Assembly Optimizer	SPARC™ Solaris™	TMDX324655-07
Simulator	Win32	TMDS3246851-07
Simulator	SPARC Solaris	TMDS3246551-07
XDS510™ Debugger/Emulation Software	Win32, Windows NT™	TMDX324016X-07
<b>Hardware</b>		
XDS510 Emulator§	PC	TMDS00510
XDS510WS™ Emulator¶	SCSI	TMDS00510WS
<b>Software/Hardware</b>		
EVM Evaluation Kit	PC/Win95/Windows NT	TMDX3260A6201
EVM Evaluation Kit (including TMDX3246855-07)	PC/Win95/Windows NT	TMDX326006201

† Contact IRVINE Compiler Corporation (949) 250-1366 to order.

‡ NT support estimated availability 1Q00.

§ Includes XDS510 board and JTAG emulation cable. TMDX324016X-07 C-source Debugger/Emulation software is not included.

¶ Includes XDS510WS box, SCSI cable, power supply, and JTAG emulation cable.

XDS, XDS510, and XDS510WS are trademarks of Texas Instruments Incorporated.

Win32 and Windows NT are trademarks of Microsoft Corporation.

SPARC is a trademark of SPARC International, Inc.

Solaris is a trademark of Sun Microsystems, Inc.



---

## **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: SMX, SM, or SMJ. Texas Instruments recommends two of three possible prefix designators for support tools: TMDX and TMDS. These prefixes represent evolutionary stages of product development from engineering prototypes (SMX/TMDX) through fully qualified production devices/tools (SMJ/TMDS). This development flow follows.

Device development evolutionary flow:

- |            |  |
|------------|--|
| <b>SMX</b> | Experimental device that is not necessarily representative of the final device's electrical specifications, 25°C tested, military/industrial ceramic dimpled Ball Grid Array package |
| <b>SM</b>  | Fully TI-qualified production device; offered in extended temperature ranges: –40°C to +90°C (S range), and –55°C to +115°C (W range); in ceramic dimpled BGA package                |
| <b>SMJ</b> | Fully SMD-qualified production device, –55°C to +115°C (W temperature range), in the ceramic dimpled Ball Grid Array package processed to MIL-PRF-38535                              |

Support tool development evolutionary flow:

- |             |  |
|-------------|--|
| <b>TMDX</b> | Development-support product that has not yet completed Texas Instruments internal qualification testing. |
| <b>TMDS</b> | 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 have been demonstrated fully. TI's standard warranty applies.

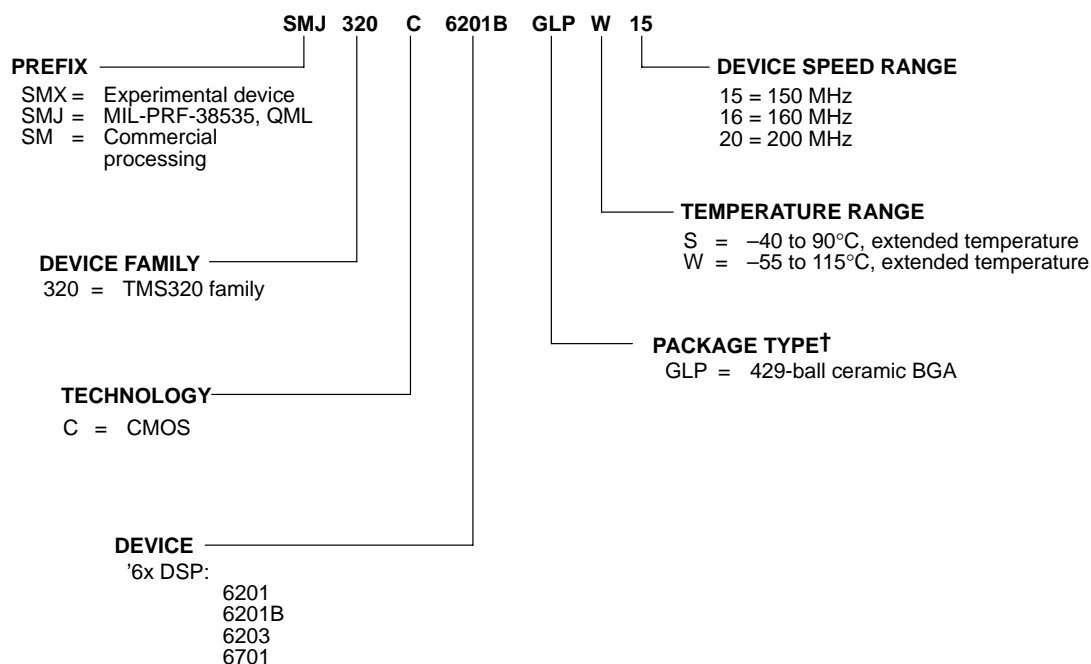
Predictions show that prototype devices (SMX or SM) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices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

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (GLP) and the device speed range in megahertz (for example, 15 is 150 MHz). Figure 5 provides a legend for reading the complete device name.

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## device and development-support tool nomenclature (continued)



† BGA = Ball Grid Array

**Figure 5. TMS320 Device Nomenclature (Including SMJ320C6201B)**

## 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 reference guides for all devices; technical briefs; development-support tools; and hardware and software applications. The following is a brief, descriptive list of support documentation specific to the C6x™ devices:

The *TMS320C6000 CPU and Instruction Set Reference Guide* (literature number SPRU189) describes the C6000 CPU architecture, instruction set, pipeline, and associated interrupts.

The *TMS320C6000 Peripherals Reference Guide* (literature number SPRU190) describes the functionality of the peripherals available on C6x devices, such as the external memory interface (EMIF), host-port interface (HPI), multichannel buffered serial ports (McBSPs), direct-memory-access (DMA), enhanced direct-memory-access (EDMA) controller, expansion bus (XB), clocking and phase-locked loop (PLL); and power-down modes. This guide also includes information on internal data and program memories.

The *TMS320C6000 Programmer's Guide* (literature number SPRU198) describes ways to optimize C and assembly code for C6x devices and includes application program examples.

The *TMS320C6x C Source Debugger User's Guide* (literature number SPRU188) describes how to invoke the C6x simulator and emulator versions of the C source debugger interface and discusses various aspects of the debugger, including: command entry, code execution, data management, breakpoints, profiling, and analysis.

The *TMS320C6x Peripheral Support Library Programmer's Reference* (literature number SPRU273) describes the contents of the C6x peripheral support library of functions and macros. It lists functions and macros both by header file and alphabetically, provides a complete description of each, and gives code examples to show how they are used.

C6x is a trademark of Texas Instruments Incorporated.





## documentation support (continued)

*TMS320C6000 Assembly Language Tools User's Guide* (literature number SPRU186) describes the assembly language tools (assembler, linker, and other tools used to develop assembly language code), assembler directives, macros, common object file format, and symbolic debugging directives for the C6000 generation of devices.

The *TMS320C6x Evaluation Module Reference Guide* (literature number SPRU269) provides instructions for installing and operating the C6x evaluation module. It also includes support software documentation, application programming interfaces, and technical reference material.

*TMS320C62x Multichannel Evaluation Module User's Guide* (literature number SPRU285) provides instructions for installing and operating the C62x multichannel evaluation module. It also includes support software documentation, application programming interfaces, and technical reference material.

*TMS320C62x Multichannel Evaluation Module Technical Reference* (SPRU308) provides provides technical reference information for the C62x multichannel evaluation module (McEVM). It includes support software documentation, application programming interface references, and hardware descriptions for the C62x McEVM.

*TMS320C6000 DSP/BIOS User's Guide* (literature number SPRU303) describes how to use DSP/BIOS tools and APIs to analyze embedded real-time DSP applications.

*Code Composer User's Guide* (literature number SPRU296) explains how to use the Code Composer development environment to build and debug embedded real-time DSP applications.

*Code Composer Studio Tutorial* (literature number SPRU301) introduces the Code Composer Studio integrated development environment and software tools.

The *TMS320C6000 Technical Brief* (literature number SPRU197) gives an introduction to the C62x/C67x devices, associated development tools, and third-party support.

A series of DSP textbooks is published by Prentice-Hall and John Wiley & Sons to support DSP 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 281/274-2323.

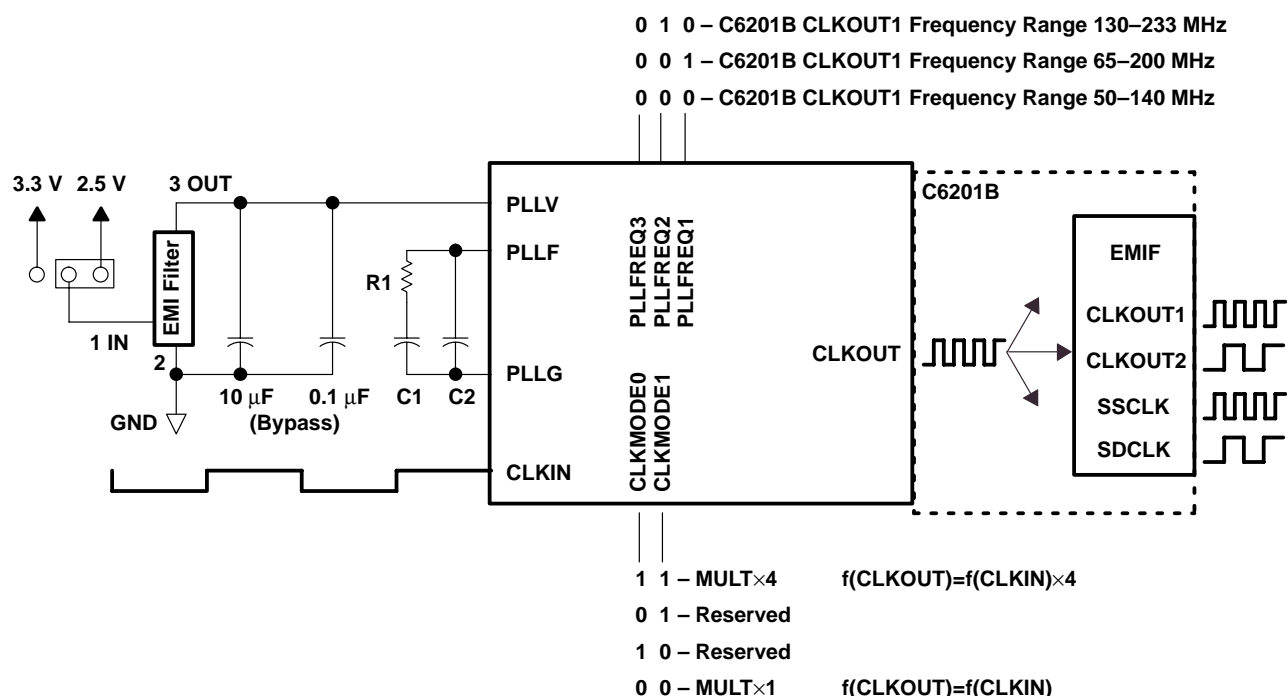
Information regarding TI DSP products is also available on the Worldwide Web at <http://www.ti.com> uniform resource locator (URL).

## clock PLL

All of the C62x clocks are generated from a single source through the CLKIN pin. This source clock either drives the PLL, which generates the internal CPU clock, or bypasses the PLL to become the CPU clock.

To use the PLL to generate the CPU clock, the filter circuit shown in Figure 6 must be properly designed. For the C6201B, it must be powered by the I/O voltage (3.3 V).

To configure the C62x PLL clock for proper operation, see Figure 6 and Table 3. To minimize the clock jitter, a single clean power supply should power both the C62x device and the external clock oscillator circuit. The minimum CLKIN rise and fall times should also be observed. See the *input and output clocks* section for input clock timing requirements. Guidelines for EMI filter selection are as follows: maximum attenuation frequency = 20-30 MHz, maximum dB attenuation = 45-50 db, and minimum dB attenuation above 30 MHz = 20 dB.



- NOTES:
- For the C6201B CLKMODE x4, values for C1, C2, and R1 are fixed and apply to all valid frequency ranges of CLKIN and CLKOUT.
  - For CLKMODE x1, the PLL is bypassed and all six external PLL components can be removed. For this case, the PLLV terminal has to be connected to a clean supply and the PLLG and PLLF terminals should be tied together.
  - Due to overlap of frequency ranges when choosing the PLLFREQ, more than one frequency range can contain the CLKOUT1 frequency. Choose the lowest frequency range that includes the desired frequency. For example, for CLKOUT1 = 133 MHz, a PLLFREQ value of 000b should be used for the C6201B. For CLKOUT1 = 200 MHz, PLLFREQ should be set to 001b for the C6201B. PLLFREQ values other than 000b, 001b, and 010b are reserved.
  - For the C6201B, the 3.3-V supply for the EMI filter (and PLLV) must be from the same 3.3-V power plane supplying the I/O voltage, DVDD.

Figure 6. PLL Block Diagram

## clock PLL (continued)

Table 3. 320C6201B PLL Component Selection Table

CLKMODE	CLKIN RANGE (MHz)	CPU CLOCK FREQUENCY (CLKOUT1) RANGE (MHz)	CLKOUT2 RANGE (MHz)	R1 ( $\Omega$ )	C1 (nF)	C2 (pF)	TYPICAL LOCK TIME ( $\mu$ s) <sup>†</sup>
x4	12.5–50	50–200	25–100	60.4	27	560	75

<sup>†</sup> Under some operating conditions, the maximum PLL lock time may vary as much as 150% from the specified typical value. For example, if the typical lock time is specified as 100  $\mu$ s, the maximum value may be as long as 250  $\mu$ s.

## power supply sequencing

TI DSPs do not require specific power sequencing between the core supply and the I/O supply. However, systems should be designed to ensure that neither supply is powered up for extended periods of time if the other supply is below the proper operating voltage.

## system-level design considerations

System-level design considerations, such as bus contention, may require supply sequencing to be implemented. In this case, the core supply should be powered up at the same time as, or prior to (and powered down after), the I/O buffers. This is to ensure that the I/O buffers receive valid inputs from the core before the output buffers are powered up, thus, preventing bus contention with other chips on the board.

## power-supply design considerations

For systems using the C6000™ DSP platform of devices, the core supply may be required to provide in excess of 2 A per DSP until the I/O supply is powered up. This extra current condition is a result of uninitialized logic within the DSP(s) and is corrected once the CPU sees an internal clock pulse. With the PLL enabled, as the I/O supply is powered on, a clock pulse is produced stopping the extra current draw from the supply. With the PLL disabled, an external clock pulse may be required to stop this extra current draw. A normal current state returns once the I/O power supply is turned on and the CPU sees a clock pulse. Decreasing the amount of time between the core supply power up and the I/O supply power up can minimize the effects of this current draw.

A dual-power supply with simultaneous sequencing, such as available with TPS563xx controllers or PT69xx plug-in power modules, can be used to eliminate the delay between core and I/O power up [see the *Using the TPS56300 to Power DSPs* application report (literature number SLVA088)]. A Schottky diode can also be used to tie the core rail to the I/O rail, effectively pulling up the I/O power supply to a level that can help initialize the logic within the DSP.

Core and I/O supply voltage regulators should be located close to the DSP (or DSP array) to minimize inductance and resistance in the power delivery path. Additionally, when designing for high-performance applications utilizing the C6000™ platform of DSPs, the PC board should include separate power planes for core, I/O, and ground, all bypassed with high-quality low-ESL/ESR capacitors.

SM320C6201B, SMJ320C6201B  
DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

absolute maximum ratings over operating case temperature range (unless otherwise noted)†

Supply voltage range, $CV_{DD}$ (see Note 1)	–0.3 V to 2.3 V
Supply voltage range, $DV_{DD}$ (see Note 1)	–0.3 V to 4 V
Input voltage range	–0.3 V to 4 V
Output voltage range	–0.3 V to 4 V
Operating case temperature range $T_C$ : (S temp version)	–40°C to 90°C
(W temp version)	–55°C to 115°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 1: All voltage values are with respect to  $V_{SS}$ .

recommended operating conditions

		C6201B			UNIT
		MIN	NOM	MAX	
$CV_{DD}$	Supply voltage	1.71	1.8	1.89	V
$DV_{DD}$	Supply voltage	3.14	3.30	3.46	V
$V_{SS}$	Supply ground	0	0	0	V
$V_{IH}$	High-level input voltage	2.0			V
$V_{IL}$	Low-level input voltage			0.8	V
$I_{OH}$	High-level output current			–12	mA
$I_{OL}$	Low-level output current			12	mA
$T_C$	Operating case temperature‡	S temp version		–40	°C
		W temp version		–55	
				115	

‡ Case temperature is measured at package bottom. There is no direct thermal path from the chip through the lid.

electrical characteristics over recommended ranges of supply voltage and operating case temperature (unless otherwise noted)

PARAMETER	TEST CONDITIONS	C6201B			UNIT
		MIN	TYP	MAX	
V <sub>OH</sub> High-level output voltage	DV <sub>DD</sub> = MIN, I <sub>OH</sub> = MAX	2.4			V
V <sub>OL</sub> Low-level output voltage	DV <sub>DD</sub> = MIN, I <sub>OL</sub> = MAX			0.6	V
I <sub>I</sub> Input current†	V <sub>I</sub> = V <sub>SS</sub> to DV <sub>DD</sub>			±10	µA
I <sub>OZ</sub> Off-state output current	V <sub>O</sub> = DV <sub>DD</sub> or 0 V			±10	µA
I <sub>DD2V</sub> Supply current, CPU + CPU memory access‡	CV <sub>DD</sub> = NOM, CPU clock = 167 MHz		380		mA
I <sub>DD2V</sub> Supply current, peripherals§	CV <sub>DD</sub> = NOM, CPU clock = 167 MHz		240		mA
I <sub>DD3V</sub> Supply current, I/O pins¶	DV <sub>DD</sub> = NOM, CPU clock = 167 MHz		90		mA
C <sub>i</sub> Input capacitance				15	pF
C <sub>o</sub> Output capacitance				15	pF

† TMS and TDI are not included due to internal pullups.

‡ TRST is not included due to internal pulldown.

‡ Measured with average CPU activity:

50% of time: 8 instructions per cycle, 32-bit DMEM access per cycle

50% of time: 2 instructions per cycle, 16-bit DMEM access per cycle

§ Measured with average peripheral activity:

50% of time: Timers at max rate, McBSPs at E1 rate, and DMA burst transfer between DMEM and SDRAM

50% of time: Timers at max rate, McBSPs at E1 rate, and DMA servicing McBSPs

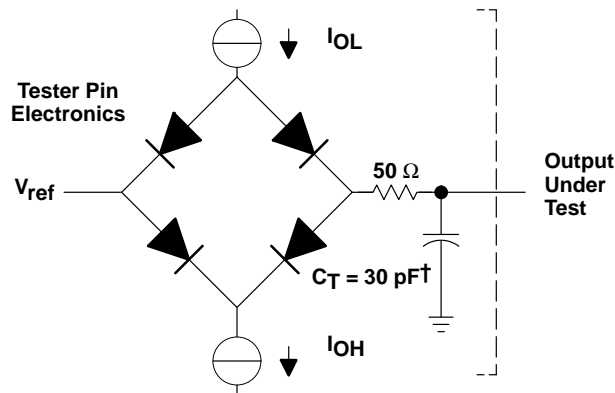
¶ Measured with average I/O activity (30-pF load):

25% of time: Reads from external SDRAM

25% of time: Writes to external SDRAM

50% of time: No activity

PARAMETER MEASUREMENT INFORMATION



† Typical distributed load circuit capacitance

Figure 7. TTL-Level Outputs

signal transition levels

All input and output timing parameters are referenced to 1.5 V for both “0” and “1” logic levels.

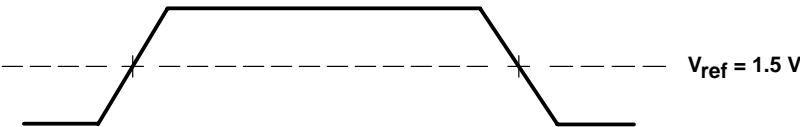


Figure 8. Input and Output Voltage Reference Levels for AC Timing Measurements

## INPUT AND OUTPUT CLOCKS

### timing requirements for CLKIN (see Figure 9)

NO.		C6201B-15				C6201B-20				UNIT
		CLKMODE = x4		CLKMODE = x1		CLKMODE = x4		CLKMODE = x1		
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
1	t <sub>c</sub> (CLKIN)    Cycle time, CLKIN	26.7		6.67		20		5		ns
2	t <sub>w</sub> (CLKINH)    Pulse duration, CLKIN high	*9.8		*2.7		*8		*2.35		ns
3	t <sub>w</sub> (CLKINL)    Pulse duration, CLKIN low	*9.8		*2.7		*8		*2.35		ns
4	t <sub>t</sub> (CLKIN)    Transition time, CLKIN	*5		*0.6		*5		*0.6		ns

\*Not production tested.

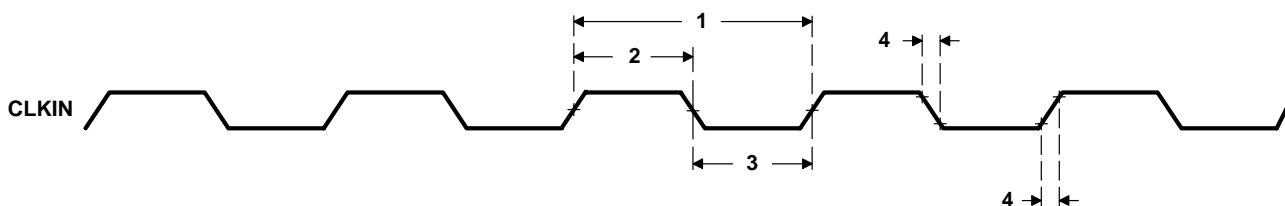


Figure 9. CLKIN Timings

### switching characteristics for CLKOUT1†‡ (see Figure 10)

NO.	PARAMETER	C6201B				UNIT
		CLKMODE = x4		CLKMODE = x1		
		MIN	MAX	MIN	MAX	
1	t <sub>c</sub> (CKO1)    Cycle time, CLKOUT1	*P – 0.7	*P + 0.7	*P – 0.7	*P + 0.7	ns
2	t <sub>w</sub> (CKO1H)    Pulse duration, CLKOUT1 high	*(P/2) – 0.5	*(P/2 ) + 0.5	*PH – 0.5	*PH + 0.5	ns
3	t <sub>w</sub> (CKO1L)    Pulse duration, CLKOUT1 low	*(P/2) – 0.5	*(P/2 ) + 0.5	*PL – 0.5	*PL + 0.5	ns
4	t <sub>t</sub> (CKO1)    Transition time, CLKOUT1		*0.6		*0.6	ns

† PH is the high period of CLKIN in ns and PL is the low period of CLKIN in ns.

‡ P = 1/CPU clock frequency in nanoseconds (ns).

\*Not production tested.

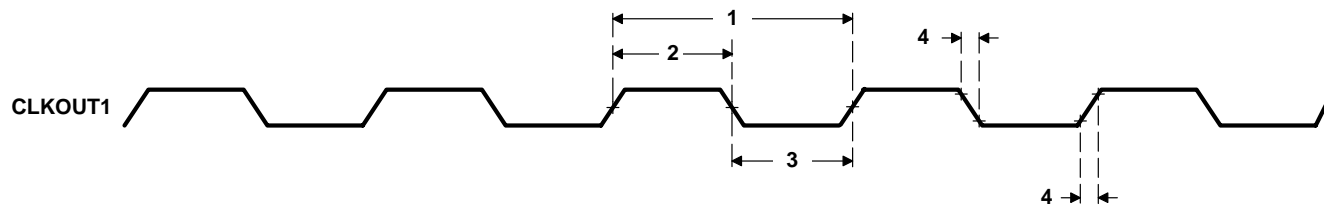


Figure 10. CLKOUT1 Timings

INPUT AND OUTPUT CLOCKS (CONTINUED)

switching characteristics for CLKOUT2† (see Figure 11)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	t <sub>c</sub> (CKO2) Cycle time, CLKOUT2	*2P – 0.7	*2P + 0.7	ns
2	t <sub>w</sub> (CKO2H) Pulse duration, CLKOUT2 high	*P – 0.9	*P + 0.7	ns
3	t <sub>w</sub> (CKO2L) Pulse duration, CLKOUT2 low	*P – 0.7	*P + 0.9	ns
4	t <sub>t</sub> (CKO2) Transition time, CLKOUT2		*0.6	ns

† P = 1/CPU clock frequency in ns.

\*Not production tested.

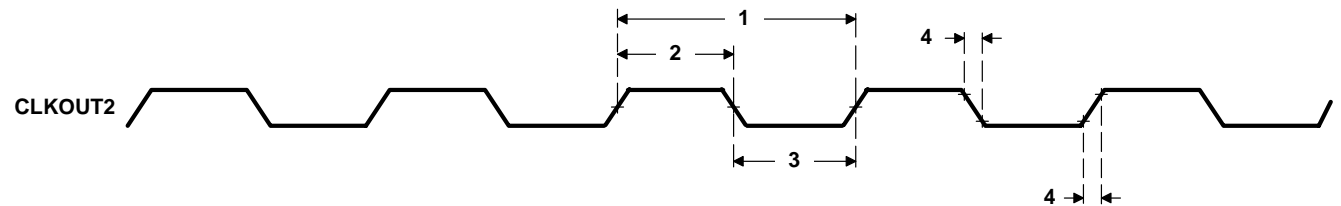


Figure 11. CLKOUT2 Timings



## INPUT AND OUTPUT CLOCKS (CONTINUED)

### SDCLK, SSCLK timing parameters

SDCLK timing parameters are the same as CLKOUT2 parameters.

SSCLK timing parameters are the same as CLKOUT1 or CLKOUT2 parameters, depending on SSCLK configuration.

### switching characteristics for the relation of SSCLK, SDCLK, and CLKOUT2 to CLKOUT1 (see Figure 12)<sup>†</sup>

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_d(\text{CKO1-SSCLK})$ Delay time, CLKOUT1 edge to SSCLK edge	$(P/2) + 0.2$	$(P/2) + 4.2$	ns
2	$t_d(\text{CKO1-SSCLK1/2})$ Delay time, CLKOUT1 edge to SSCLK edge (1/2 clock rate)	$(P/2) - 1$	$(P/2) + 2.4$	ns
3	$t_d(\text{CKO1-CKO2})$ Delay time, CLKOUT1 edge to CLKOUT2 edge	$*(P/2) - 1$	$*(P/2) + 2.4$	ns
4	$t_d(\text{CKO1-SDCLK})$ Delay time, CLKOUT1 edge to SDCLK edge	$(P/2) - 1$	$(P/2) + 2.4$	ns

<sup>†</sup> P = 1/CPU clock frequency in ns.

\*Not production tested.

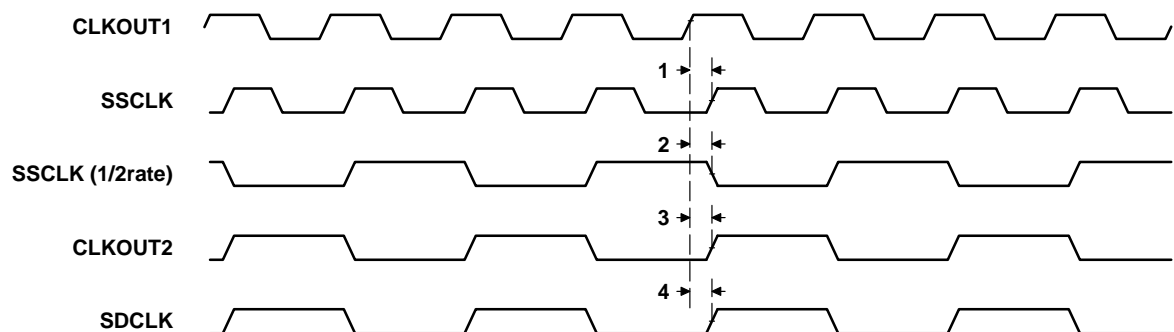


Figure 12. Relation of CLKOUT2, SDCLK, and SSCLK to CLKOUT1

## ASYNCHRONOUS MEMORY TIMING

**timing requirements for asynchronous memory cycles<sup>†</sup> (see Figure 13 and Figure 14)**

NO.		C6201B		UNIT
		MIN	MAX	
6	$t_{su}(EDV-CKO1H)$ Setup time, read EDx valid before CLKOUT1 high	4.0		ns
7	$t_h(CKO1H-EDV)$ Hold time, read EDx valid after CLKOUT1 high	0.8		ns
10	$t_{su}(ARDY-CKO1H)$ Setup time, ARDY valid before CLKOUT1 high	3.0		ns
11	$t_h(CKO1H-ARDY)$ Hold time, ARDY valid after CLKOUT1 high	1.8		ns

<sup>†</sup> To ensure data setup time, simply program the strobe width wide enough. ARDY is internally synchronized. If ARDY does meet setup or hold time, it may be recognized in the current cycle or the next cycle. Thus, ARDY can be an asynchronous input.

**switching characteristics for asynchronous memory cycles<sup>‡</sup> (see Figure 13 and Figure 14)**

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_d(CKO1H-CEV)$ Delay time, CLKOUT1 high to $\overline{CE}$ valid	-0.2	4.0	ns
2	$t_d(CKO1H-BEV)$ Delay time, CLKOUT1 high to $\overline{BEx}$ valid		4.0	ns
3	$t_d(CKO1H-BEIV)$ Delay time, CLKOUT1 high to $\overline{BEx}$ invalid	*-0.2		ns
4	$t_d(CKO1H-EAV)$ Delay time, CLKOUT1 high to EAx valid		4.0	ns
5	$t_d(CKO1H-EAIV)$ Delay time, CLKOUT1 high to EAx invalid	*-0.2		ns
8	$t_d(CKO1H-AOEV)$ Delay time, CLKOUT1 high to $\overline{AOE}$ valid	-0.2	4.0	ns
9	$t_d(CKO1H-AREV)$ Delay time, CLKOUT1 high to $\overline{ARE}$ valid	-0.2	4.0	ns
12	$t_d(CKO1H-EDV)$ Delay time, CLKOUT1 high to EDx valid		4.0	ns
13	$t_d(CKO1H-EDIV)$ Delay time, CLKOUT1 high to EDx invalid	*-0.2		ns
14	$t_d(CKO1H-AWEV)$ Delay time, CLKOUT1 high to $\overline{AWE}$ valid	-0.2	4.0	ns

<sup>‡</sup> The minimum delay is also the minimum output hold after CLKOUT1 high.

\*Not production tested.

## ASYNCHRONOUS MEMORY TIMING (CONTINUED)

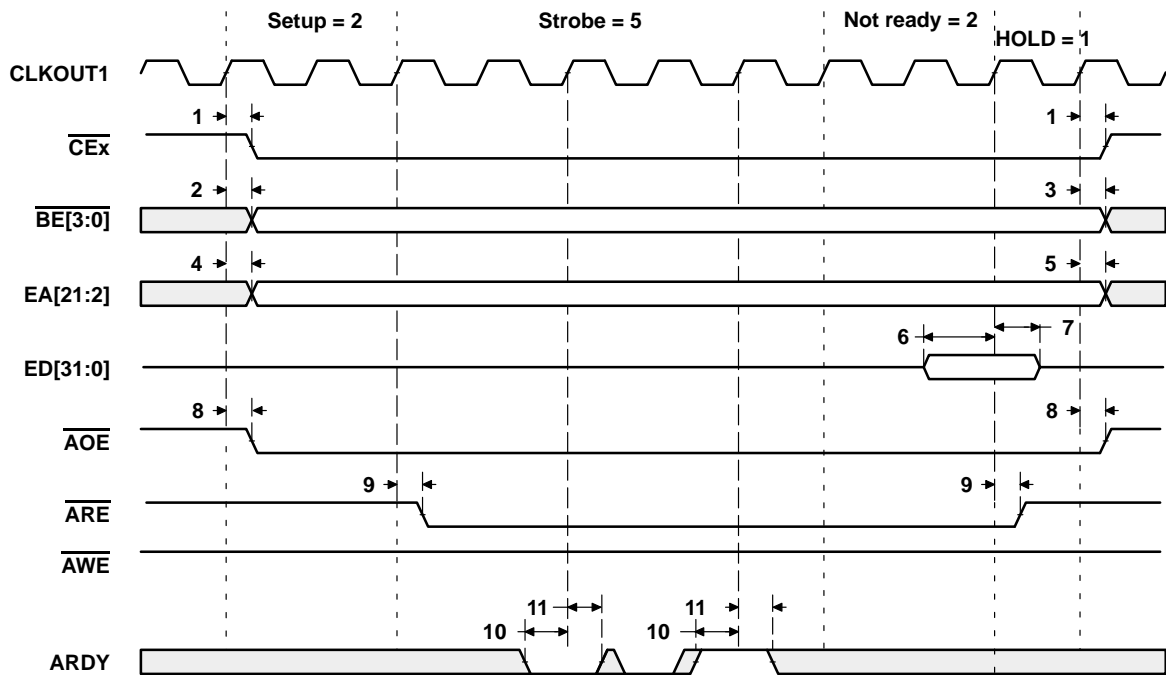


Figure 13. Asynchronous Memory Read Timing

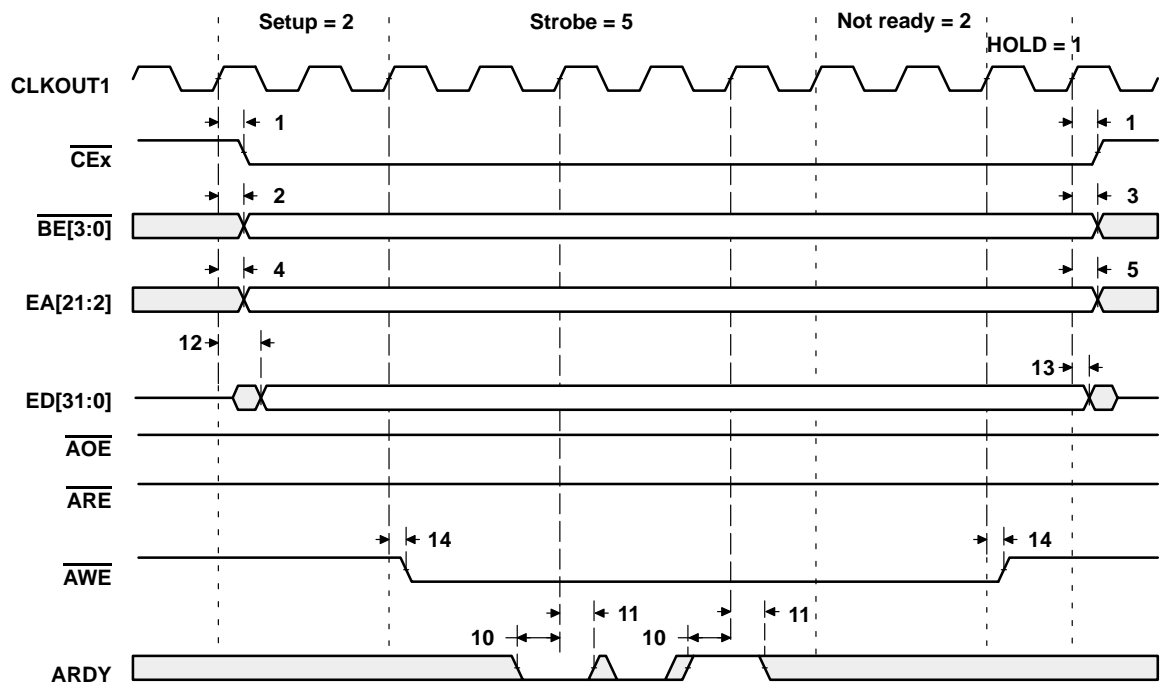


Figure 14. Asynchronous Memory Write Timing

## SYNCHRONOUS-BURST MEMORY TIMING

**timing requirements for synchronous-burst SRAM cycles (full-rate SSCLK) (see Figure 15)**

NO.		C6201B		UNIT
		MIN	MAX	
7	$t_{su}(EDV-SSCLKH)$ Setup time, read EDx valid before SSCLK high	1.7		ns
8	$t_h(SSCLKH-EDV)$ Hold time, read EDx valid after SSCLK high	1.5		ns

**switching characteristics for synchronous-burst SRAM cycles<sup>†</sup> (full-rate SSCLK)**  
**(see Figure 15 and Figure 16)**

NO.	PARAMETER		C6201B		UNIT
			MIN	MAX	
1	$t_{osu}(CEV-SSCLKH)$	Output setup time, $\overline{CE}x$ valid before SSCLK high	0.5P – 1.3		ns
2	$t_{oh}(SSCLKH-CEV)$	Output hold time, $\overline{CE}x$ valid after SSCLK high	0.5P – 2.3		ns
3	$t_{osu}(BEV-SSCLKH)$	Output setup time, $\overline{BE}x$ valid before SSCLK high	0.5P – 1.3		ns
4	$t_{oh}(SSCLKH-BEIV)$	Output hold time, $\overline{BE}x$ invalid after SSCLK high	*0.5P – 2.3		ns
5	$t_{osu}(EAV-SSCLKH)$	Output setup time, EAx valid before SSCLK high	0.5P – 1.3		ns
6	$t_{oh}(SSCLKH-EAIV)$	Output hold time, EAx invalid after SSCLK high	*0.5P – 2.3		ns
9	$t_{osu}(ADSV-SSCLKH)$	Output setup time, $\overline{SSADS}$ valid before SSCLK high	0.5P – 1.3		ns
10	$t_{oh}(SSCLKH-ADSV)$	Output hold time, $\overline{SSADS}$ valid after SSCLK high	0.5P – 2.3		ns
11	$t_{osu}(OEV-SSCLKH)$	Output setup time, $\overline{SSOE}$ valid before SSCLK high	0.5P – 1.3		ns
12	$t_{oh}(SSCLKH-OEV)$	Output hold time, $\overline{SSOE}$ valid after SSCLK high	0.5P – 2.3		ns
13	$t_{osu}(EDV-SSCLKH)$	Output setup time, EDx valid before SSCLK high	0.5P – 1.3		ns
14	$t_{oh}(SSCLKH-EDIV)$	Output hold time, EDx invalid after SSCLK high	*0.5P – 2.3		ns
15	$t_{osu}(WEV-SSCLKH)$	Output setup time, $\overline{SSWE}$ valid before SSCLK high	0.5P – 1.3		ns
16	$t_{oh}(SSCLKH-WEV)$	Output hold time, $\overline{SSWE}$ valid after SSCLK high	0.5P – 2.3		ns

<sup>†</sup> When the PLL is used (CLKMODE x4), P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

For CLKMODE x1, 0.5P is defined as PH (pulse duration of CLKIN high) for all output setup times; 0.5P is defined as PL (pulse duration of CLKIN low) for all output hold times.

\*Not production tested.

## SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)

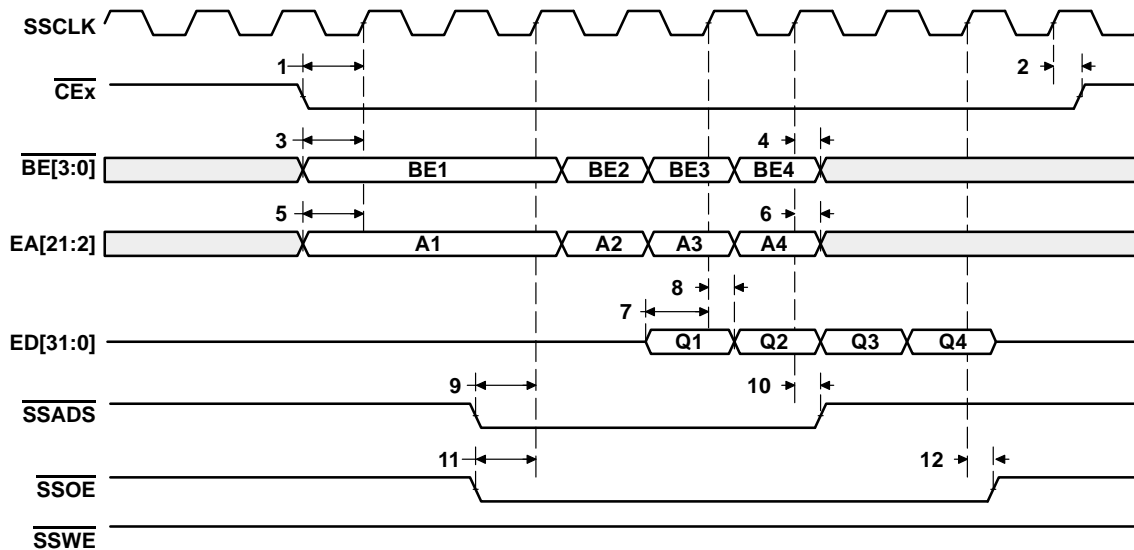


Figure 15. SBSRAM Read Timing (Full-Rate SSCLK)

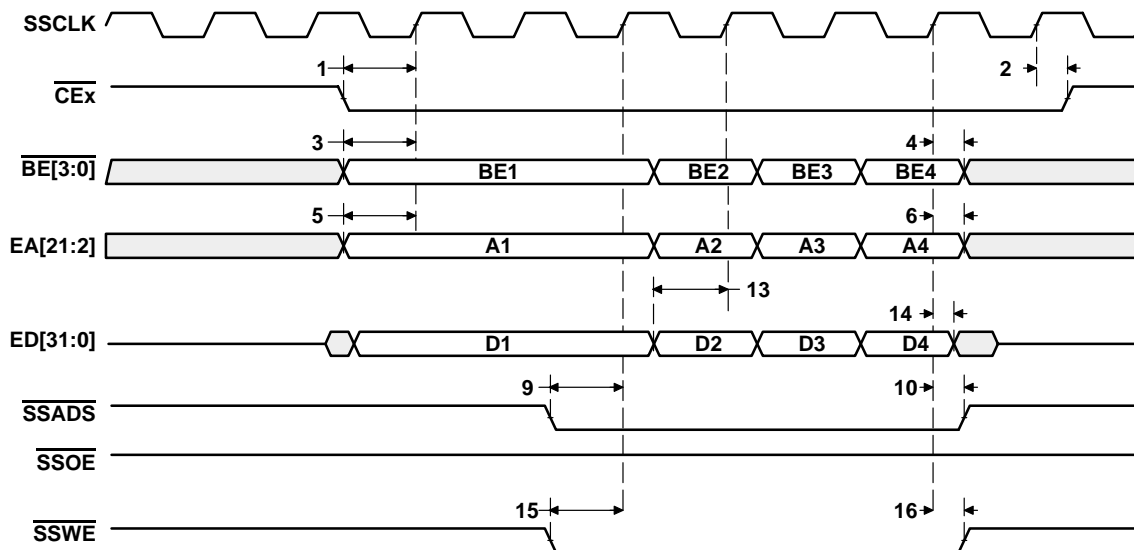


Figure 16. SBSRAM Write Timing (Full-Rate SSCLK)

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)

timing requirements for synchronous-burst SRAM cycles (half-rate SSCLK)  
(see Figure 17)

NO.		C6201B		UNIT
		MIN	MAX	
7	$t_{su}(EDV-SSCLKH)$ Setup time, read EDx valid before SSCLK high	2.5		ns
8	$t_h(SSCLKH-EDV)$ Hold time, read EDx valid after SSCLK high	1.5		ns

switching characteristics for synchronous-burst SRAM cycles<sup>†</sup> (half-rate SSCLK)  
(see Figure 17 and Figure 18)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_{osu}(CEV-SSCLKH)$ Output setup time, $\overline{CEx}$ valid before SSCLK high	1.5P – 3		ns
2	$t_{oh}(SSCLKH-CEV)$ Output hold time, $\overline{CEx}$ valid after SSCLK high	0.5P – 1.5		ns
3	$t_{osu}(BEV-SSCLKH)$ Output setup time, $\overline{BEx}$ valid before SSCLK high	1.5P – 3		ns
4	$t_{oh}(SSCLKH-BEIV)$ Output hold time, $\overline{BEx}$ invalid after SSCLK high	*0.5P – 1.5		ns
5	$t_{osu}(EAV-SSCLKH)$ Output setup time, EAx valid before SSCLK high	1.5P – 3		ns
6	$t_{oh}(SSCLKH-EAIV)$ Output hold time, EAx invalid after SSCLK high	*0.5P – 1.5		ns
9	$t_{osu}(ADSV-SSCLKH)$ Output setup time, $\overline{SSADS}$ valid before SSCLK high	1.5P – 3		ns
10	$t_{oh}(SSCLKH-ADSV)$ Output hold time, $\overline{SSADS}$ valid after SSCLK high	0.5P – 1.5		ns
11	$t_{osu}(OEV-SSCLKH)$ Output setup time, $\overline{SSOE}$ valid before SSCLK high	1.5P – 3		ns
12	$t_{oh}(SSCLKH-OEV)$ Output hold time, $\overline{SSOE}$ valid after SSCLK high	0.5P – 1.5		ns
13	$t_{osu}(EDV-SSCLKH)$ Output setup time, EDx valid before SSCLK high	1.5P – 3		ns
14	$t_{oh}(SSCLKH-EDIV)$ Output hold time, EDx invalid after SSCLK high	*0.5P – 1.5		ns
15	$t_{osu}(WEV-SSCLKH)$ Output setup time, $\overline{SSWE}$ valid before SSCLK high	1.5P – 3		ns
16	$t_{oh}(SSCLKH-WEV)$ Output hold time, $\overline{SSWE}$ valid after SSCLK high	0.5P – 1.5		ns

<sup>†</sup> When the PLL is used (CLKMODE x4), P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

For CLKMODE x1:

1.5P = P + PH, where P = 1/CPU clock frequency, and PH = pulse duration of CLKIN high.

0.5P = PL, where PL = pulse duration of CLKIN low.

\*Not production tested.

# SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)

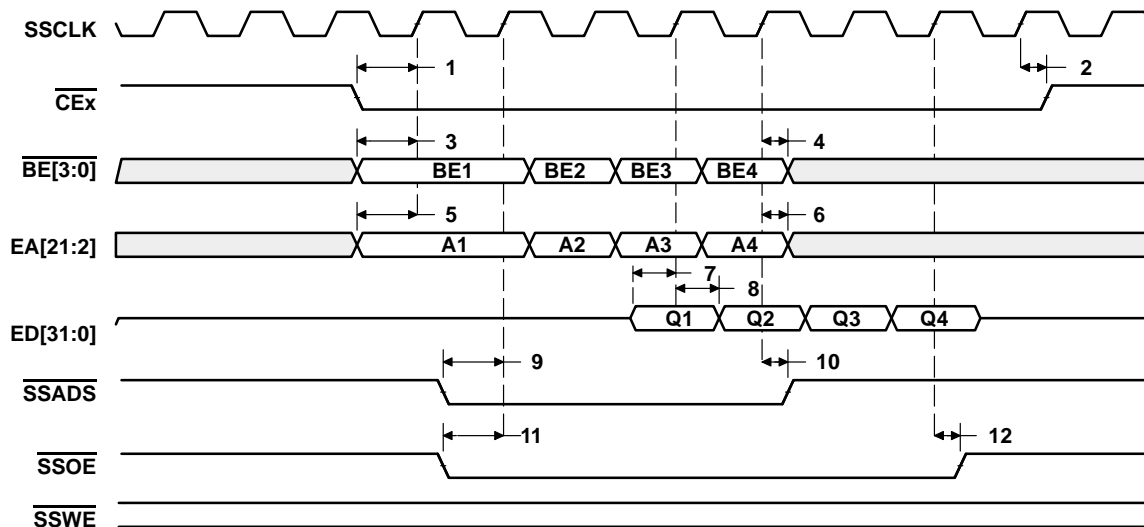


Figure 17. SBSRAM Read Timing (1/2 Rate SSCLK)

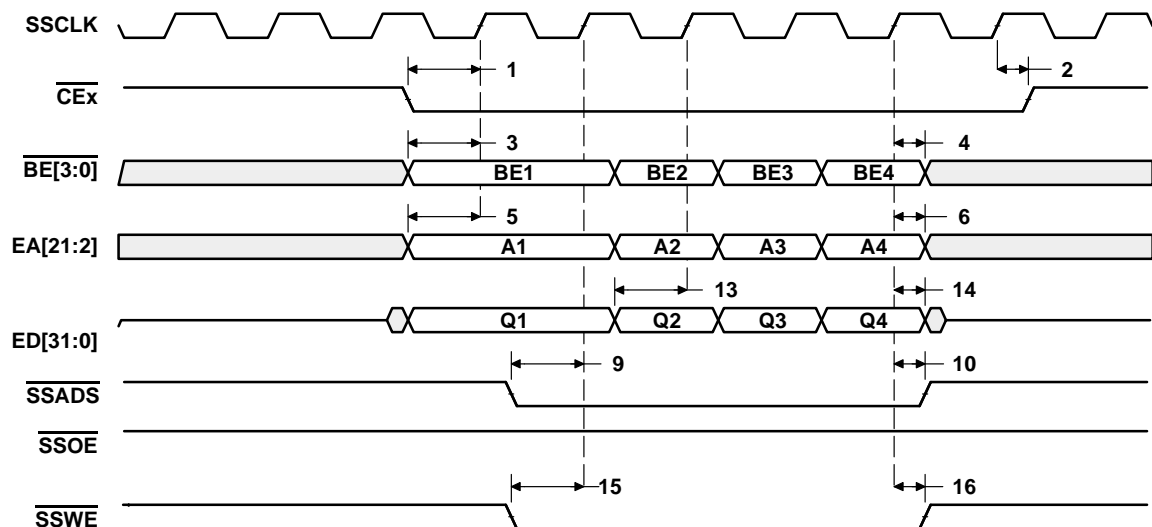


Figure 18. SBSRAM Write Timing (1/2 Rate SSCLK)

# SM320C6201B, SMJ320C6201B DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

## SYNCHRONOUS DRAM TIMING

timing requirements for synchronous DRAM cycles (see Figure 19)

NO.		C6201B		UNIT
		MIN	MAX	
7	$t_{su}(EDV-SDCLKH)$ Setup time, read EDx valid before SDCLK high	0.5		ns
8	$t_h(SDCLKH-EDV)$ Hold time, read EDx valid after SDCLK high	3		ns

switching characteristics for synchronous DRAM cycles<sup>†</sup> (see Figure 19–Figure 24)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_{osu}(CEV-SDCLKH)$ Output setup time, $\overline{CEx}$ valid before SDCLK high	1.5P – 3.5		ns
2	$t_{oh}(SDCLKH-CEV)$ Output hold time, $\overline{CEx}$ valid after SDCLK high	0.5P – 1		ns
3	$t_{osu}(BEV-SDCLKH)$ Output setup time, $\overline{BEx}$ valid before SDCLK high	1.5P – 3.5		ns
4	$t_{oh}(SDCLKH-BEIV)$ Output hold time, $\overline{BEx}$ invalid after SDCLK high	*0.5P – 1		ns
5	$t_{osu}(EAV-SDCLKH)$ Output setup time, EAx valid before SDCLK high	1.5P – 3.5		ns
6	$t_{oh}(SDCLKH-EAIV)$ Output hold time, EAx invalid after SDCLK high	*0.5P – 1		ns
9	$t_{osu}(SDCAS-SDCLKH)$ Output setup time, $\overline{SDCAS}$ valid before SDCLK high	1.5P – 3.5		ns
10	$t_{oh}(SDCLKH-SDCAS)$ Output hold time, $\overline{SDCAS}$ valid after SDCLK high	0.5P – 1		ns
11	$t_{osu}(EDV-SDCLKH)$ Output setup time, EDx valid before SDCLK high	1.5P – 3.5		ns
12	$t_{oh}(SDCLKH-EDIV)$ Output hold time, EDx invalid after SDCLK high	*0.5P – 1		ns
13	$t_{osu}(SDWE-SDCLKH)$ Output setup time, $\overline{SDWE}$ valid before SDCLK high	1.5P – 3.5		ns
14	$t_{oh}(SDCLKH-SDWE)$ Output hold time, $\overline{SDWE}$ valid after SDCLK high	0.5P – 1		ns
15	$t_{osu}(SDA10V-SDCLKH)$ Output setup time, SDA10 valid before SDCLK high	1.5P – 3.5		ns
16	$t_{oh}(SDCLKH-SDA10IV)$ Output hold time, SDA10 invalid after SDCLK high	*0.5P – 1		ns
17	$t_{osu}(SDRAS-SDCLKH)$ Output setup time, $\overline{SDRAS}$ valid before SDCLK high	1.5P – 3.5		ns
18	$t_{oh}(SDCLKH-SDRAS)$ Output hold time, $\overline{SDRAS}$ valid after SDCLK high	0.5P – 1		ns

<sup>†</sup> When the PLL is used (CLKMODE x4), P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

For CLKMODE x1:

1.5P = P + PH, where P = 1/CPU clock frequency, and PH = pulse duration of CLKIN high.

0.5P = PL, where PL = pulse duration of CLKIN low.

\*Not production tested.



# SYNCHRONOUS DRAM TIMING (CONTINUED)

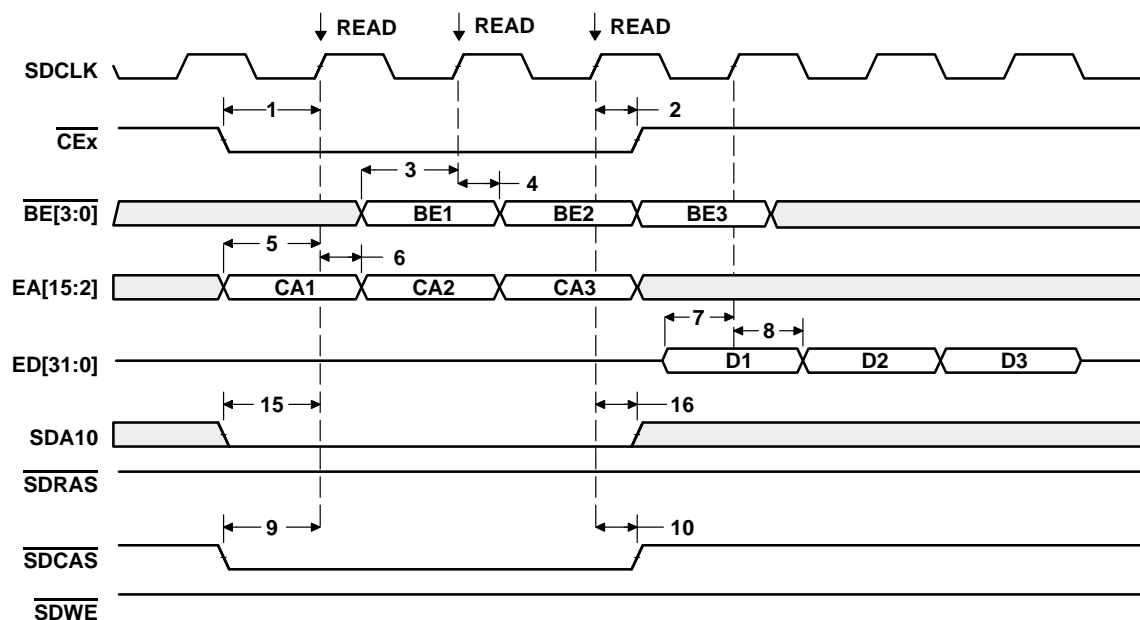


Figure 19. Three SDRAM Read Commands

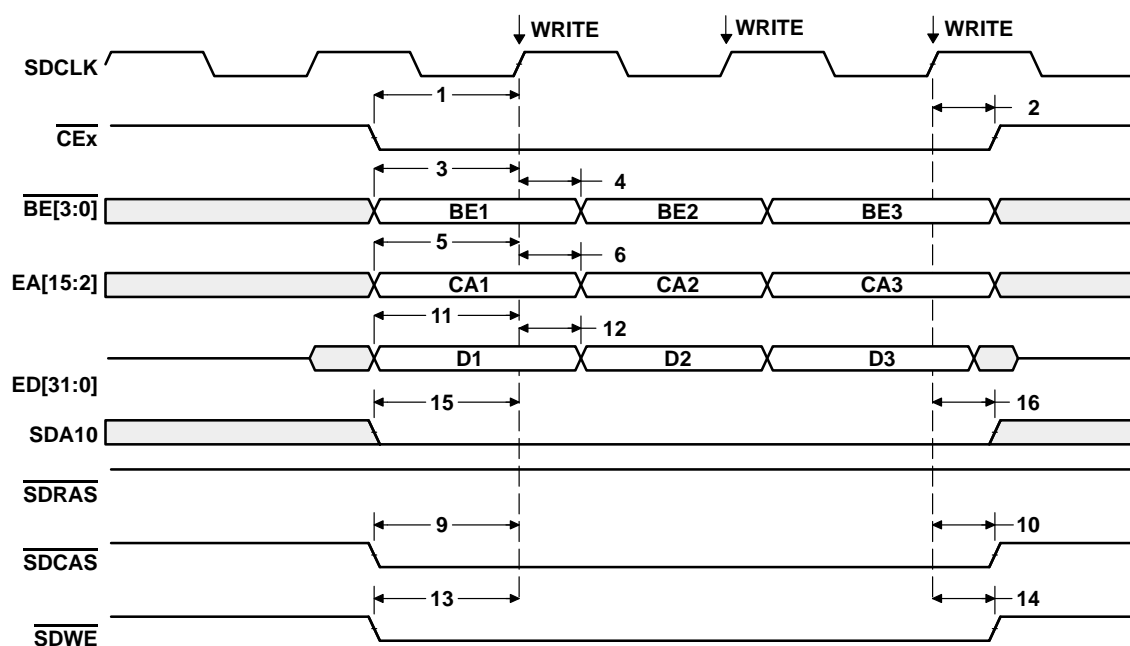


Figure 20. Three SDRAM WRT Commands

SYNCHRONOUS DRAM TIMING (CONTINUED)

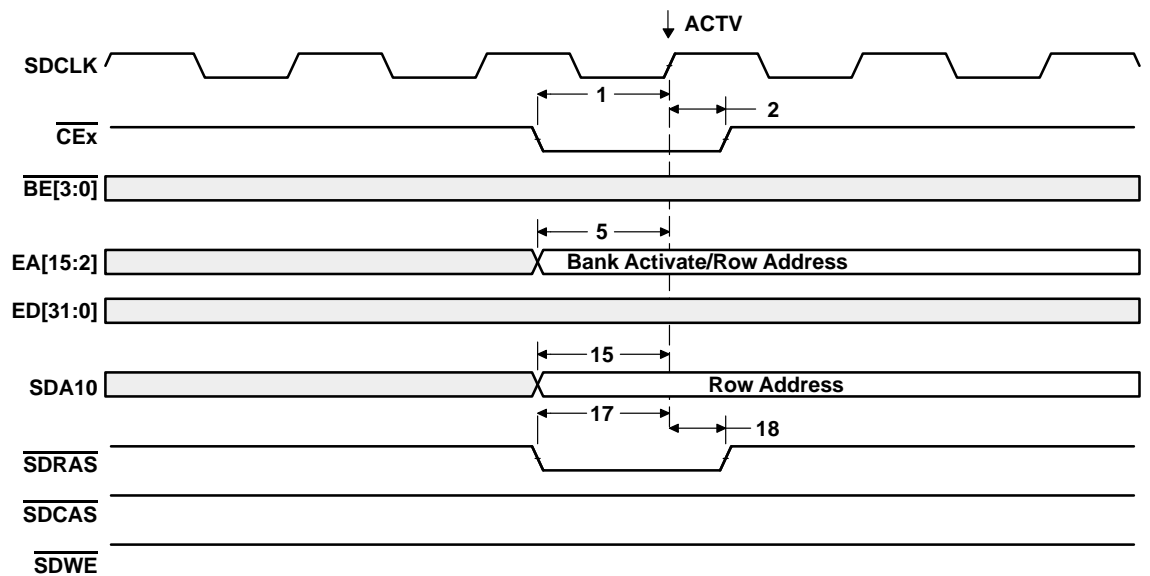


Figure 21. SDRAM ACTV Command

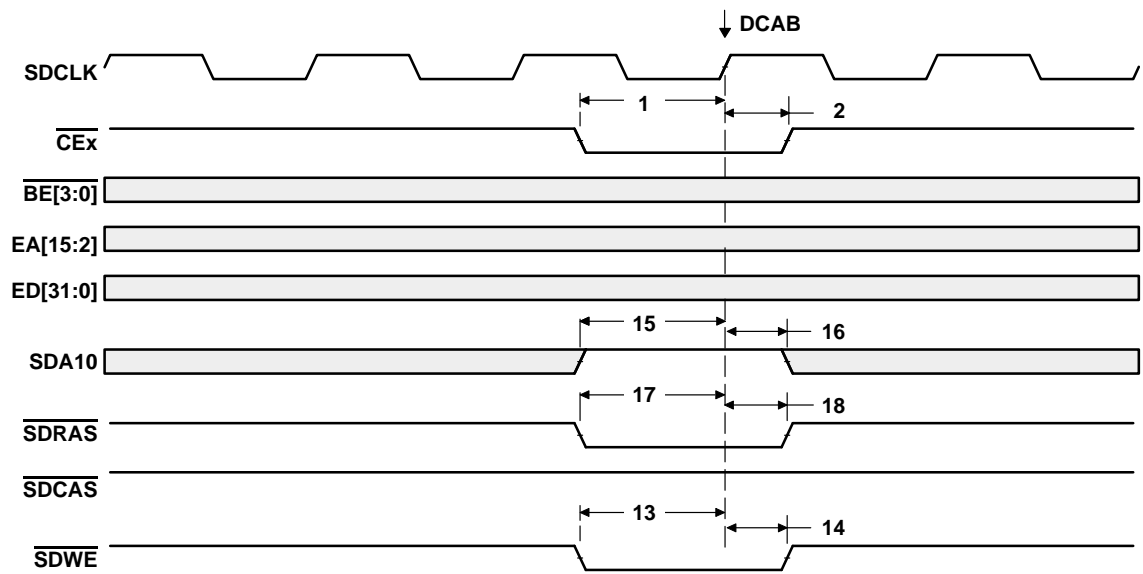


Figure 22. SDRAM DCAB Command

SYNCHRONOUS DRAM TIMING (CONTINUED)

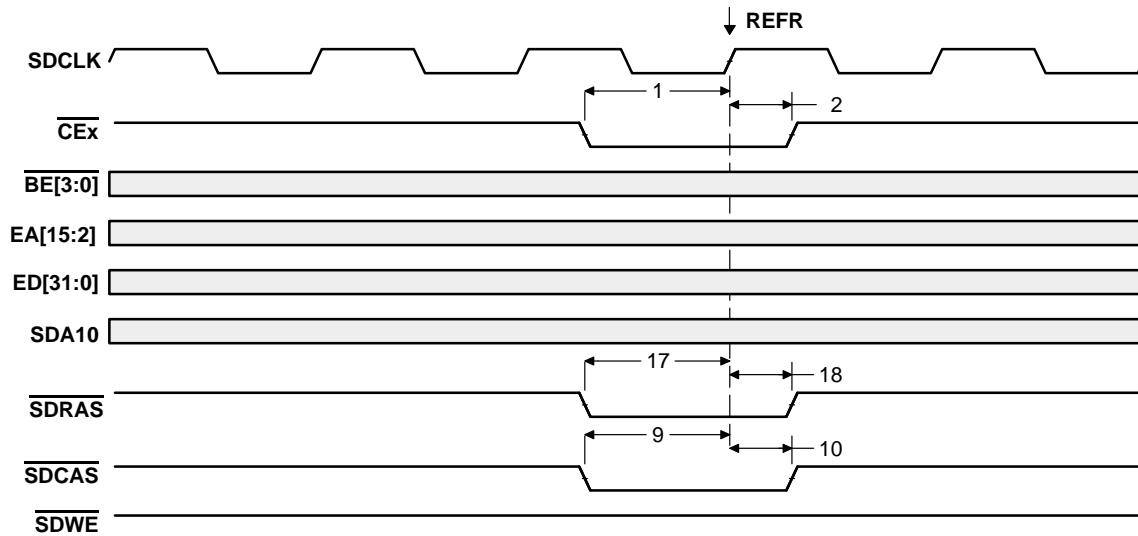


Figure 23. SDRAM REFR Command

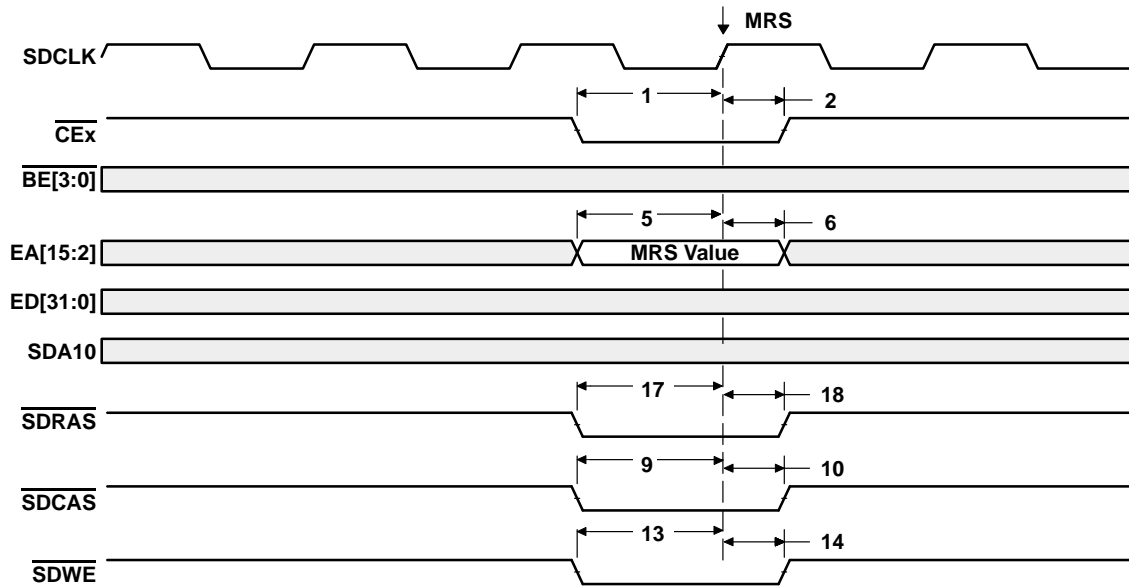


Figure 24. SDRAM MRS Command

## HOLD/HOLDA TIMING

timing requirements for the  $\overline{\text{HOLD}}/\overline{\text{HOLDA}}$  cycles<sup>†</sup> (see Figure 25)

NO.		C6201B		UNIT
		MIN	MAX	
1	$t_{su}(\overline{\text{HOLDH}}\text{-}\overline{\text{CKO1H}})$ Setup time, $\overline{\text{HOLD}}$ high before $\overline{\text{CLKOUT1}}$ high	*1		ns
2	$t_h(\overline{\text{CKO1H}}\text{-}\overline{\text{HOLDL}})$ Hold time, $\overline{\text{HOLD}}$ low after $\overline{\text{CLKOUT1}}$ high	*4		ns

<sup>†</sup>  $\overline{\text{HOLD}}$  is synchronized internally. Therefore, if setup and hold times are not met, it will either be recognized in the current cycle or in the next cycle. Thus,  $\overline{\text{HOLD}}$  can be an asynchronous input.

\*Not production tested.

switching characteristics for the  $\overline{\text{HOLD}}/\overline{\text{HOLDA}}$  cycles<sup>‡</sup> (see Figure 25)

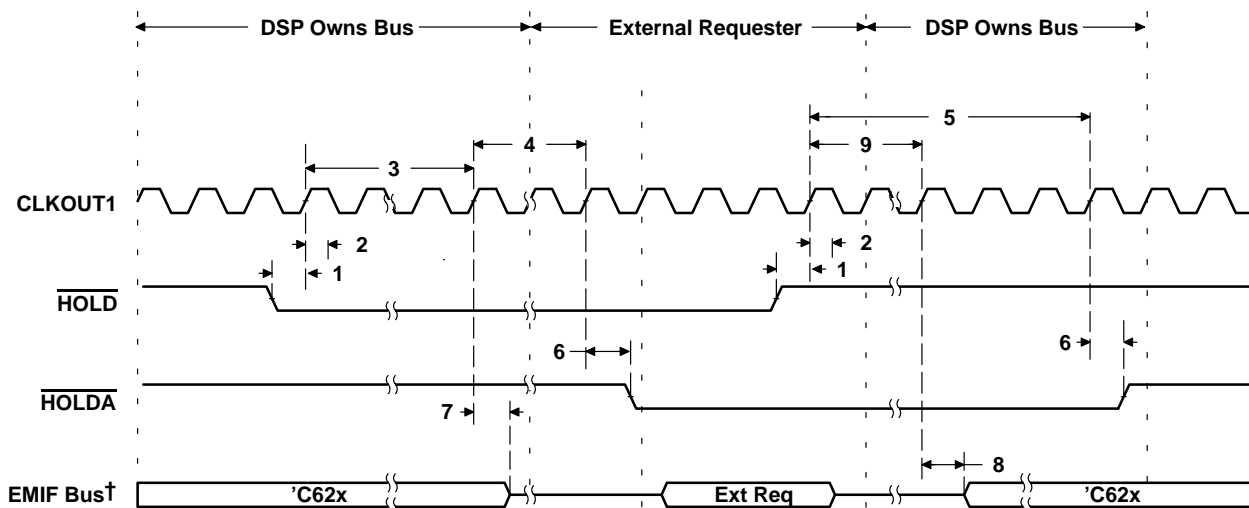
NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
3	$t_R(\overline{\text{HOLDL}}\text{-}\overline{\text{BHZ}})$ Response time, $\overline{\text{HOLD}}$ low to EMIF Bus high impedance	*4P	§	ns
4	$t_R(\overline{\text{BHZ}}\text{-}\overline{\text{HOLDAL}})$ Response time, EMIF Bus high impedance to $\overline{\text{HOLDA}}$ low	*P	*2P	ns
5	$t_R(\overline{\text{HOLDH}}\text{-}\overline{\text{HOLDAH}})$ Response time, $\overline{\text{HOLD}}$ high to $\overline{\text{HOLDA}}$ high	*4P	*7P	ns
6	$t_d(\overline{\text{CKO1H}}\text{-}\overline{\text{HOLDAL}})$ Delay time, $\overline{\text{CLKOUT1}}$ high to $\overline{\text{HOLDA}}$ valid	*1	8	ns
7	$t_d(\overline{\text{CKO1H}}\text{-}\overline{\text{BHZ}})$ Delay time, $\overline{\text{CLKOUT1}}$ high to EMIF Bus high impedance <sup>¶</sup>	*3	*11	ns
8	$t_d(\overline{\text{CKO1H}}\text{-}\overline{\text{BLZ}})$ Delay time, $\overline{\text{CLKOUT1}}$ high to EMIF Bus low impedance <sup>¶</sup>	*3	*11	ns
9	$t_R(\overline{\text{HOLDH}}\text{-}\overline{\text{BLZ}})$ Response time, $\overline{\text{HOLD}}$ high to EMIF Bus low impedance	*3P	*6P	ns

<sup>‡</sup> P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

\*Not production tested.

§ All pending EMIF transactions are allowed to complete before  $\overline{\text{HOLDA}}$  is asserted. The worst cases for this is an asynchronous read or write with external ARDY used or a minimum of eight consecutive SDRAM reads or writes when RBTR8 = 1. If no bus transactions are occurring, then the minimum delay time can be achieved. Also, bus hold can be indefinitely delayed by setting NOHOLD = 1.

¶ EMIF Bus consists of  $\overline{\text{CE}}[3:0]$ ,  $\overline{\text{BE}}[3:0]$ ,  $\overline{\text{ED}}[31:0]$ ,  $\overline{\text{EA}}[21:2]$ ,  $\overline{\text{ARE}}$ ,  $\overline{\text{AOE}}$ ,  $\overline{\text{AWE}}$ ,  $\overline{\text{SSADS}}$ ,  $\overline{\text{SSOE}}$ ,  $\overline{\text{SSWE}}$ ,  $\overline{\text{SDA10}}$ ,  $\overline{\text{SDRAS}}$ ,  $\overline{\text{SDCAS}}$ , and  $\overline{\text{SDWE}}$ .



<sup>†</sup> EMIF Bus consists of  $\overline{\text{CE}}[3:0]$ ,  $\overline{\text{BE}}[3:0]$ ,  $\overline{\text{ED}}[31:0]$ ,  $\overline{\text{EA}}[21:2]$ ,  $\overline{\text{ARE}}$ ,  $\overline{\text{AOE}}$ ,  $\overline{\text{AWE}}$ ,  $\overline{\text{SSADS}}$ ,  $\overline{\text{SSOE}}$ ,  $\overline{\text{SSWE}}$ ,  $\overline{\text{SDA10}}$ ,  $\overline{\text{SDRAS}}$ ,  $\overline{\text{SDCAS}}$ , and  $\overline{\text{SDWE}}$ .

Figure 25.  $\overline{\text{HOLD}}/\overline{\text{HOLDA}}$  Timing

## RESET TIMING

timing requirements for reset (see Figure 26)

NO.			C6201B		UNIT
			MIN	MAX	
1	$t_{w(RST)}$	Width of the $\overline{RESET}$ pulse (PLL stable) <sup>†</sup>	*10		CLKOUT1 cycles
		Width of the $\overline{RESET}$ pulse (PLL needs to sync up) <sup>‡</sup>	250		μs

<sup>†</sup> This parameter applies to CLKMODE x1 when CLKIN is stable and applies to CLKMODE x4 when CLKIN and PLL are stable.

<sup>‡</sup> This parameter only applies to CLKMODE x4. The  $\overline{RESET}$  signal is not connected internally to the clock PLL circuit. The PLL, however, may need up to 250 μs to stabilize following device power up or after PLL configuration has been changed. During that time,  $\overline{RESET}$  must be asserted to ensure proper device operation. See the *clock PLL* section for PLL lock times.

\*Not production tested.

switching characteristics during reset<sup>§</sup> (see Figure 26)

NO.	PARAMETER		C6201B		UNIT
			MIN	MAX	
2	$t_{R(RST)}$	Response time to change of value in $\overline{RESET}$ signal	2		CLKOUT1 cycles
3	$t_d(CKO1H-CKO2IV)$	Delay time, CLKOUT1 high to CLKOUT2 invalid	*-1		ns
4	$t_d(CKO1H-CKO2V)$	Delay time, CLKOUT1 high to CLKOUT2 valid		10	ns
5	$t_d(CKO1H-SDCLKIV)$	Delay time, CLKOUT1 high to SDCLK invalid	*-1		ns
6	$t_d(CKO1H-SDCLKV)$	Delay time, CLKOUT1 high to SDCLK valid		10	ns
7	$t_d(CKO1H-SSCKIV)$	Delay time, CLKOUT1 high to SSCLK invalid	*-1		ns
8	$t_d(CKO1H-SSCKV)$	Delay time, CLKOUT1 high to SSCLK valid		10	ns
9	$t_d(CKO1H-LOWIV)$	Delay time, CLKOUT1 high to low group invalid	*-1		ns
10	$t_d(CKO1H-LOWV)$	Delay time, CLKOUT1 high to low group valid		*10	ns
11	$t_d(CKO1H-HIGHIV)$	Delay time, CLKOUT1 high to high group invalid	*-1		ns
12	$t_d(CKO1H-HIGHV)$	Delay time, CLKOUT1 high to high group valid		*10	ns
13	$t_d(CKO1H-ZHZ)$	Delay time, CLKOUT1 high to Z group high impedance	*-1		ns
14	$t_d(CKO1H-ZV)$	Delay time, CLKOUT1 high to Z group valid		*10	ns

<sup>§</sup> Low group consists of:  $\overline{IACK}$ ,  $\overline{INUM}[3:0]$ ,  $\overline{DMAC}[3:0]$ ,  $\overline{PD}$ ,  $\overline{TOUT0}$ , and  $\overline{TOUT1}$

High group consists of:  $\overline{HINT}$

Z group consists of:  $\overline{EA}[21:2]$ ,  $\overline{ED}[31:0]$ ,  $\overline{CE}[3:0]$ ,  $\overline{BE}[3:0]$ ,  $\overline{ARE}$ ,  $\overline{AWE}$ ,  $\overline{AOE}$ ,  $\overline{SSADS}$ ,  $\overline{SSOE}$ ,  $\overline{SSWE}$ ,  $\overline{SDA10}$ ,  $\overline{SDRAS}$ ,  $\overline{SDCAS}$ ,  $\overline{SDWE}$ ,  $\overline{HD}[15:0]$ ,  $\overline{CLKX0}$ ,  $\overline{CLKX1}$ ,  $\overline{FSX0}$ ,  $\overline{FSX1}$ ,  $\overline{DX0}$ ,  $\overline{DX1}$ ,  $\overline{CLKR0}$ ,  $\overline{CLKR1}$ ,  $\overline{FSR0}$ , and  $\overline{FSR1}$ .

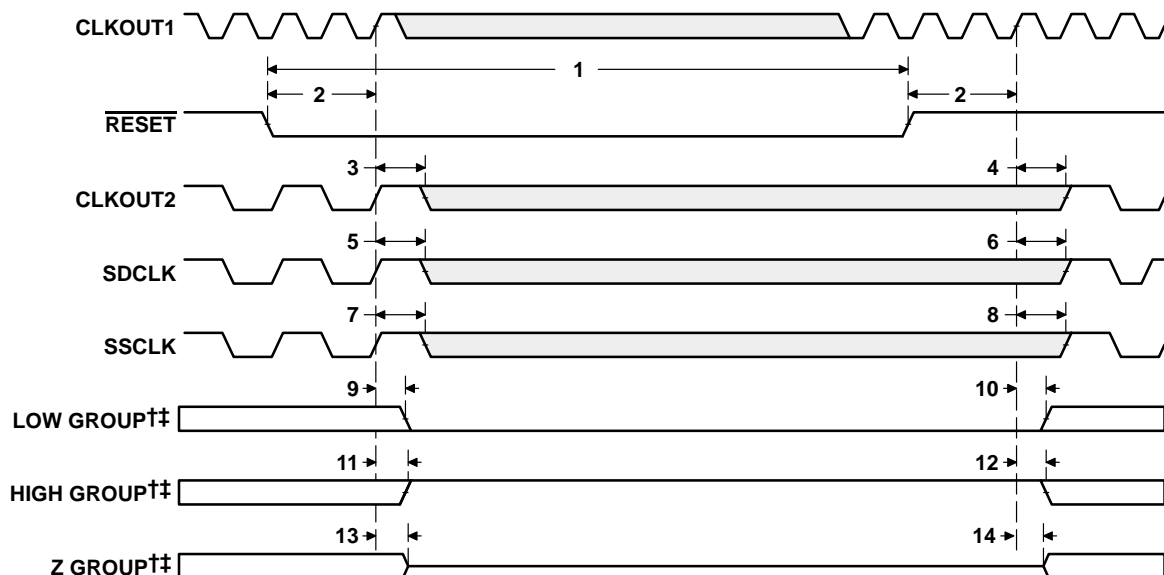
<sup>¶</sup>  $\overline{HRDY}$  is gated by input  $\overline{HCS}$ .

If  $\overline{HCS} = 0$  at device reset,  $\overline{HRDY}$  belongs to the high group.

If  $\overline{HCS} = 1$  at device reset,  $\overline{HRDY}$  belongs to the low group.

\*Not production tested.

### RESET TIMING (CONTINUED)



† Low group consists of: IACK, INUM[3:0], DMAC[3:0], PD, TOUT0, and TOUT1

High group consists of: HINT

Z group consists of: EA[21:2], ED[31:0], CE[3:0], BE[3:0], ARE, AWE, AOE, SSADS, SSOE, SSWE, SDA10, SDRAS, SDCAS, SDWE, HD[15:0], CLKX0, CLKX1, FSX0, FSX1, DX0, DX1, CLKR0, CLKR1, FSR0, and FSR1.

‡ HRDY is gated by input HCS.

If HCS = 0 at device reset, HRDY belongs to the high group.

If HCS = 1 at device reset, HRDY belongs to the low group.

Figure 26. Reset Timing

## EXTERNAL INTERRUPT TIMING

timing requirements for interrupt response cycles<sup>†‡</sup> (see Figure 27)

NO.			C6201B		UNIT
			MIN	MAX	
2	$t_{w(ILOW)}$	Width of the interrupt pulse low	*2P		ns
3	$t_{w(IHIGH)}$	Width of the interrupt pulse high	*2P		ns

<sup>†</sup> Interrupt signals are synchronized internally and are potentially recognized one cycle later if setup and hold times are violated. Thus, they can be connected to asynchronous inputs.

<sup>‡</sup> P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

\*Not production tested.

switching characteristics during interrupt response cycles<sup>§</sup> (see Figure 27)

NO.	PARAMETER		C6201B		UNIT
			MIN	MAX	
1	$t_{R(EINTH-IACKH)}$	Response time, EXT_INTx high to IACK high	*9P		ns
4	$t_{d(CKO2L-IACKV)}$	Delay time, CLKOUT2 low to IACK valid	*-4	6	ns
5	$t_{d(CKO2L-INUMV)}$	Delay time, CLKOUT2 low to INUMx valid		6	ns
6	$t_{d(CKO2L-INUMIV)}$	Delay time, CLKOUT2 low to INUMx invalid	*-4		ns

<sup>§</sup> P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

When the PLL is used (CLKMODE x4), 0.5P = 1/(2 × CPU clock frequency).

For CLKMODE x1: 0.5P = PH, where PH is the high period of CLKIN.

\*Not production tested.

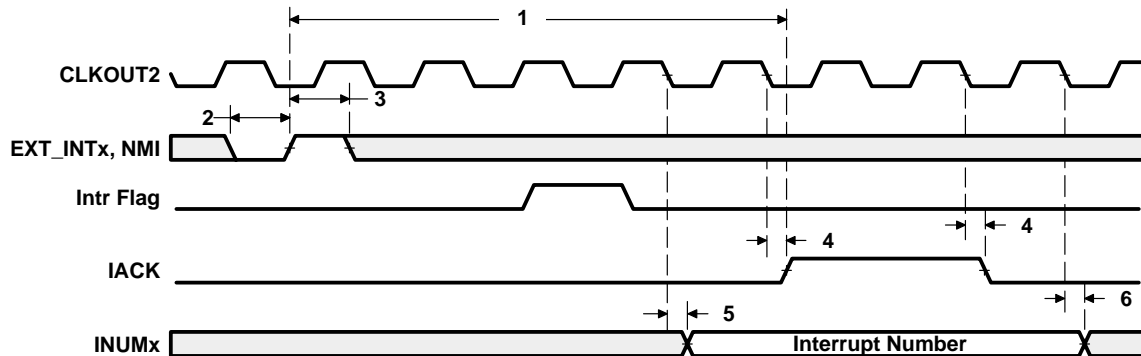


Figure 27. Interrupt Timing

## HOST-PORT INTERFACE TIMING

timing requirements for host-port interface cycles<sup>†‡</sup> (see Figure 28, Figure 29, Figure 30, and Figure 31)

NO.			C6201B		UNIT
			MIN	MAX	
1	$t_{su}(\overline{SEL}\text{-HSTBL})$	Setup time, select signals <sup>§</sup> valid before $\overline{HSTROBE}$ low	4		ns
2	$t_h(\overline{HSTBL}\text{-SEL})$	Hold time, select signals <sup>§</sup> valid after $\overline{HSTROBE}$ low	2		ns
3	$t_w(\overline{HSTBL})$	Pulse duration, $\overline{HSTROBE}$ low	2P		ns
4	$t_w(\overline{HSTBH})$	Pulse duration, $\overline{HSTROBE}$ high between consecutive accesses	*2P		ns
10	$t_{su}(\overline{SEL}\text{-HASL})$	Setup time, select signals <sup>§</sup> valid before $\overline{HAS}$ low	4		ns
11	$t_h(\overline{HASL}\text{-SEL})$	Hold time, select signals <sup>§</sup> valid after $\overline{HAS}$ low	2		ns
12	$t_{su}(\overline{HDV}\text{-HSTBH})$	Setup time, host data valid before $\overline{HSTROBE}$ high	4		ns
13	$t_h(\overline{HSTBH}\text{-HDV})$	Hold time, host data valid after $\overline{HSTROBE}$ high	2		ns
14	$t_h(\overline{HRDY}\text{-HSTBL})$	Hold time, $\overline{HSTROBE}$ low after $\overline{HRDY}$ low. $\overline{HSTROBE}$ should not be inactivated until $\overline{HRDY}$ is active (low); otherwise, HPI writes will not complete properly.	*1		ns
18	$t_{su}(\overline{HASL}\text{-HSTBL})$	Setup time, $\overline{HAS}$ low before $\overline{HSTROBE}$ low	*2		ns
19	$t_h(\overline{HSTBL}\text{-HASL})$	Hold time, $\overline{HAS}$ low after $\overline{HSTROBE}$ low	*2		ns

<sup>†</sup>  $\overline{HSTROBE}$  refers to the following logical operation on  $\overline{HCS}$ ,  $\overline{HDS1}$ , and  $\overline{HDS2}$ :  $[\text{NOT}(\overline{HDS1} \text{ XOR } \overline{HDS2})] \text{ OR } \overline{HCS}$ .

<sup>‡</sup> The effects of internal clock jitter are included at test. There is no need to adjust timing numbers for internal clock jitter. P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

<sup>§</sup> Select signals include:  $\overline{HCNTRL}[1:0]$ ,  $\overline{HR}/\overline{W}$ , and  $\overline{HHWIL}$ .

\*Not production tested.

switching characteristics during host-port interface cycles<sup>†‡</sup> (see Figure 28, Figure 29, Figure 30, and Figure 31)

NO.	PARAMETER		C6201B		UNIT
			MIN	MAX	
5	$t_d(\overline{HCS}\text{-HRDY})$	Delay time, $\overline{HCS}$ to $\overline{HRDY}$ <sup>†</sup>	*1	9	ns
6	$t_d(\overline{HSTBL}\text{-HRDYH})$	Delay time, $\overline{HSTROBE}$ low to $\overline{HRDY}$ high <sup>#</sup>	*3	12	ns
7	$t_{oh}(\overline{HSTBL}\text{-HDLZ})$	Output hold time, HD low impedance after $\overline{HSTROBE}$ low for an HPI read	*4		ns
8	$t_d(\overline{HDV}\text{-HRDYH})$	Delay time, HD valid to $\overline{HRDY}$ low	*P – 3	*P + 3	ns
9	$t_{oh}(\overline{HSTBH}\text{-HDV})$	Output hold time, HD valid after $\overline{HSTROBE}$ high	*1	*12	ns
15	$t_d(\overline{HSTBH}\text{-HDLZ})$	Delay time, $\overline{HSTROBE}$ high to HD high impedance	*3	*12	ns
16	$t_d(\overline{HSTBL}\text{-HDV})$	Delay time, $\overline{HSTROBE}$ low to HD valid	*2	*12	ns
17	$t_d(\overline{HSTBH}\text{-HRDYH})$	Delay time, $\overline{HSTROBE}$ high to $\overline{HRDY}$ high <sup>  </sup>	*3	12	ns

<sup>†</sup>  $\overline{HSTROBE}$  refers to the following logical operation on  $\overline{HCS}$ ,  $\overline{HDS1}$ , and  $\overline{HDS2}$ :  $[\text{NOT}(\overline{HDS1} \text{ XOR } \overline{HDS2})] \text{ OR } \overline{HCS}$ .

<sup>‡</sup> The effects of internal clock jitter are included at test. There is no need to adjust timing numbers for internal clock jitter. P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

<sup>†</sup>  $\overline{HCS}$  enables  $\overline{HRDY}$ , and  $\overline{HRDY}$  is always low when  $\overline{HCS}$  is high. The case where  $\overline{HRDY}$  goes high when  $\overline{HCS}$  falls indicates that HPI is busy completing a previous HPID write or READ with autoincrement.

<sup>#</sup> This parameter is used during an HPID read. At the beginning of the first half-word transfer on the falling edge of  $\overline{HSTROBE}$ , the HPI sends the request to the DMA auxiliary channel, and  $\overline{HRDY}$  remains high until the DMA auxiliary channel loads the requested data into HPID.

<sup>||</sup> This parameter is used after the second half-word of an HPID write or autoincrement read.  $\overline{HRDY}$  remains low if the access is not an HPID write or autoincrement read. Reading or writing to HPIC or HPIA does not affect the  $\overline{HRDY}$  signal.

\*Not production tested.



HOST-PORT INTERFACE TIMING (CONTINUED)

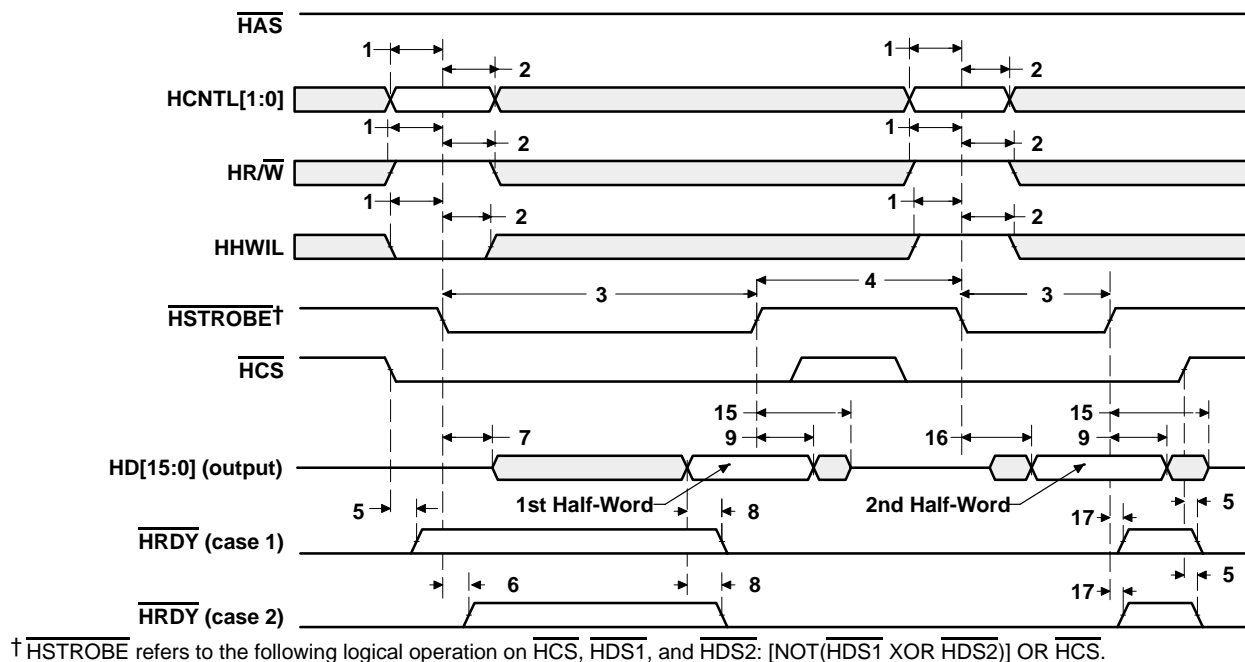


Figure 28. HPI Read Timing ( $\overline{HAS}$  Not Used, Tied High)

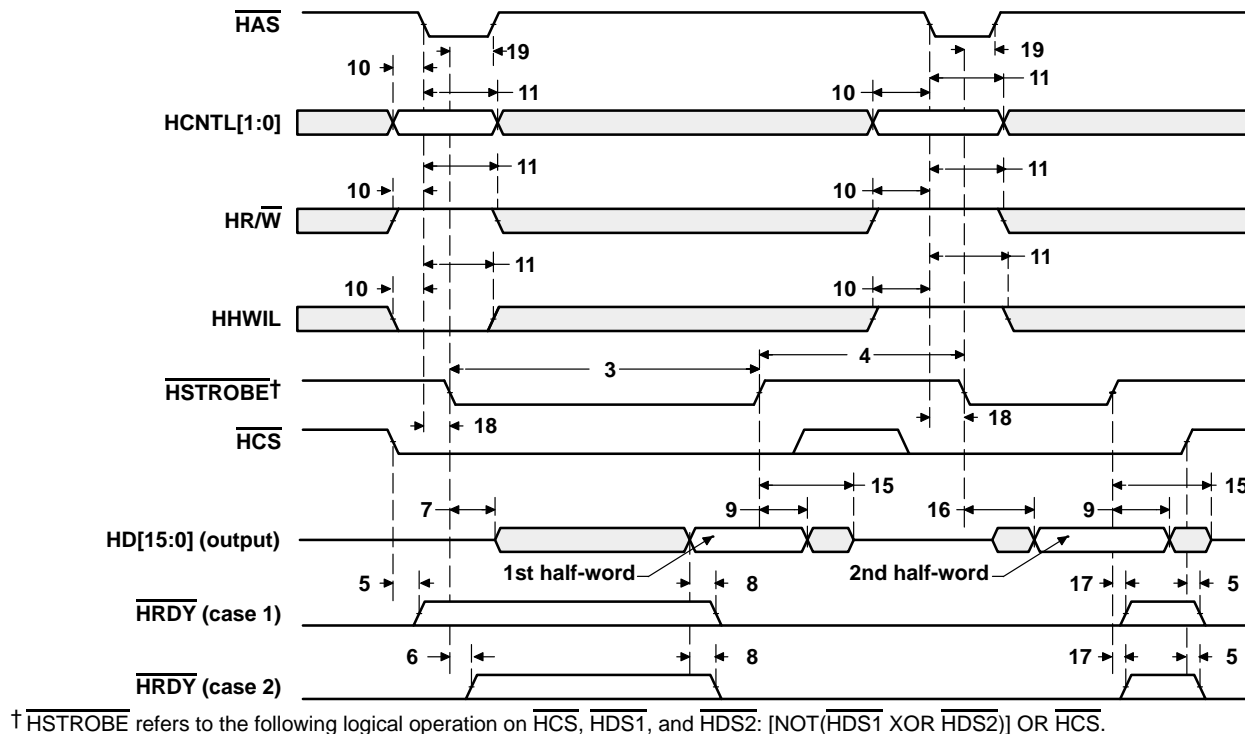
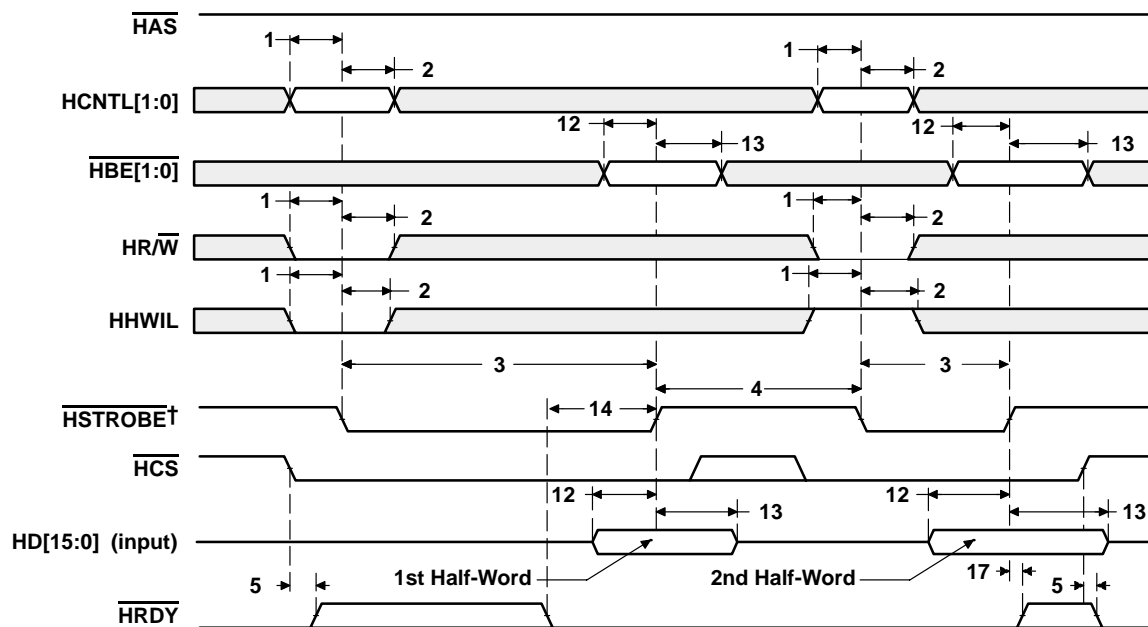


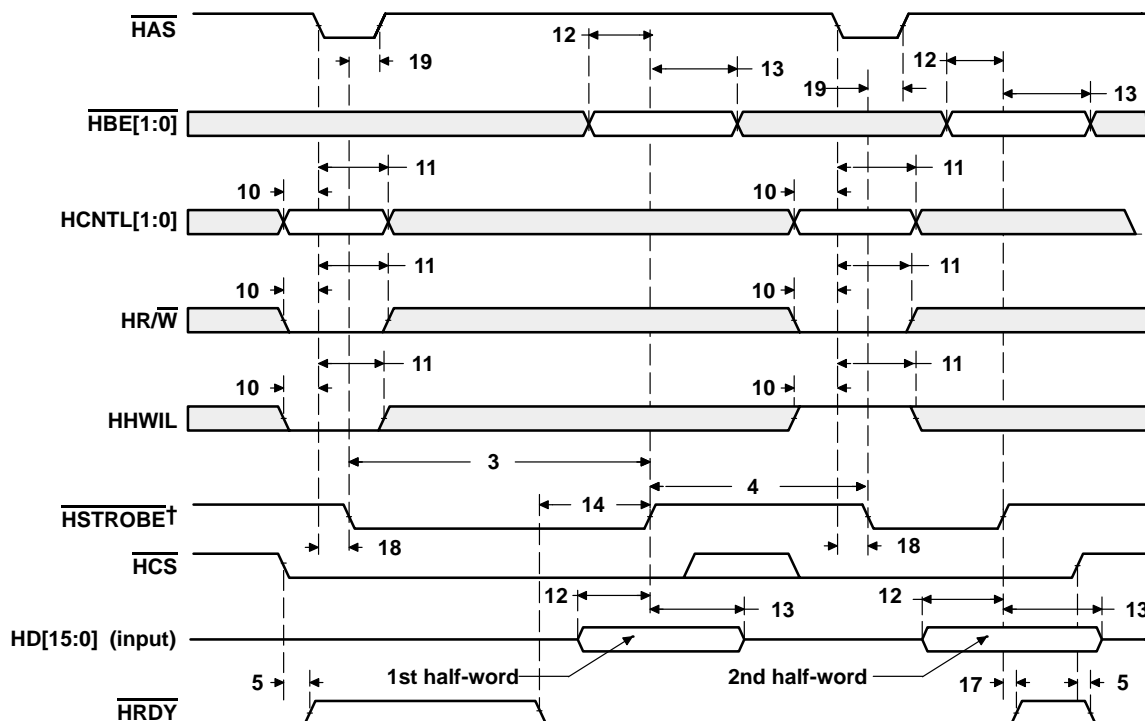
Figure 29. HPI Read Timing ( $\overline{HAS}$  Used)

**HOST-PORT INTERFACE TIMING (CONTINUED)**



†  $\text{HSTROBE}$  refers to the following logical operation on  $\overline{\text{HCS}}$ ,  $\text{HDS1}$ , and  $\text{HDS2}$ :  $[\text{NOT}(\text{HDS1 XOR HDS2})] \text{ OR } \overline{\text{HCS}}$ .

**Figure 30. HPI Write Timing ( $\overline{\text{HAS}}$  Not Used, Tied High)**



†  $\text{HSTROBE}$  refers to the following logical operation on  $\overline{\text{HCS}}$ ,  $\text{HDS1}$ , and  $\text{HDS2}$ :  $[\text{NOT}(\text{HDS1 XOR HDS2})] \text{ OR } \overline{\text{HCS}}$ .

**Figure 31. HPI Write Timing ( $\overline{\text{HAS}}$  Used)**

## MULTICHANNEL BUFFERED SERIAL PORT TIMING

timing requirements for McBSP<sup>†‡</sup>(see Figure 32)

NO.				C6201B		UNIT
				MIN	MAX	
2	t <sub>c</sub> (CKRX)	Cycle time, CLKR/X	CLKR/X ext	*2P		ns
3	t <sub>w</sub> (CKRX)	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X ext	*P – 1		ns
5	t <sub>su</sub> (FRH-CKRL)	Setup time, external FSR high before CLKR low	CLKR int	*9		ns
			CLKR ext	2		
6	t <sub>h</sub> (CKRL-FRH)	Hold time, external FSR high after CLKR low	CLKR int	*6		ns
			CLKR ext	3		
7	t <sub>su</sub> (DRV-CKRL)	Setup time, DR valid before CLKR low	CLKR int	8		ns
			CLKR ext	1		
8	t <sub>h</sub> (CKRL-DRV)	Hold time, DR valid after CLKR low	CLKR int	3		ns
			CLKR ext	4		
10	t <sub>su</sub> (FXH-CKXL)	Setup time, external FSX high before CLKX low	CLKX int	9		ns
			CLKX ext	2		
11	t <sub>h</sub> (CKXL-FXH)	Hold time, external FSX high after CLKX low	CLKX int	6		ns
			CLKX ext	3		

<sup>†</sup> CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

<sup>‡</sup> P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

\*Not production tested

# SM320C6201B, SMJ320C6201B

## DIGITAL SIGNAL PROCESSOR

SGUS031B – APRIL 2000 – REVISED AUGUST 2001

### MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

#### switching characteristics for McBSP†‡§ (see Figure 32)

NO.	PARAMETER		C6201B		UNIT
			MIN	MAX	
1	$t_d(\text{CKSH-CKRXH})$	Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input	3	10	ns
2	$t_c(\text{CKRX})$	Cycle time, CLKR/X	2P		ns
3	$t_w(\text{CKRX})$	Pulse duration, CLKR/X high or CLKR/X low	*C – 1.6¶	*C + 1¶	ns
4	$t_d(\text{CKRH-FRV})$	Delay time, CLKR high to internal FSR valid	CLKR int	*–2.5	3
9	$t_d(\text{CKXH-FXV})$	Delay time, CLKX high to internal FSX valid	CLKX int	*–2	3
			CLKX ext	*3	*9
12	$t_{dis}(\text{CKXH-DXHZ})$	Disable time, DX high impedance following last data bit from CLKX high	CLKX int	*–1	*4
			CLKX ext	*3	*9
13	$t_d(\text{CKXH-DXV})$	Delay time, CLKX high to DX valid	CLKX int	*–1	*4
			CLKX ext	*3	*9
14	$t_d(\text{FXH-DXV})$	Delay time, FSX high to DX valid	FSX int	*–1	*3
		ONLY applies when in data delay 0 (XDATDLY = 00b) mode	FSX ext	*3	*9

† CLKRP = CLKXP = FSRP = FSXP = 0. If polarity of any of the signals is inverted, then the timing references of that signal are also inverted.

‡ Minimum delay times also represent minimum output hold times.

§ P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

\*Not production tested.

¶ C = H or L

S = sample rate generator input clock = P if CLKSM = 1 (P = 1/CPU clock frequency)

= sample rate generator input clock = P\_clks if CLKSM = 0 (P\_clks = CLKS period)

H = CLKX high pulse width = (CLKGDV/2 + 1) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

L = CLKX low pulse width = (CLKGDV/2) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero



MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

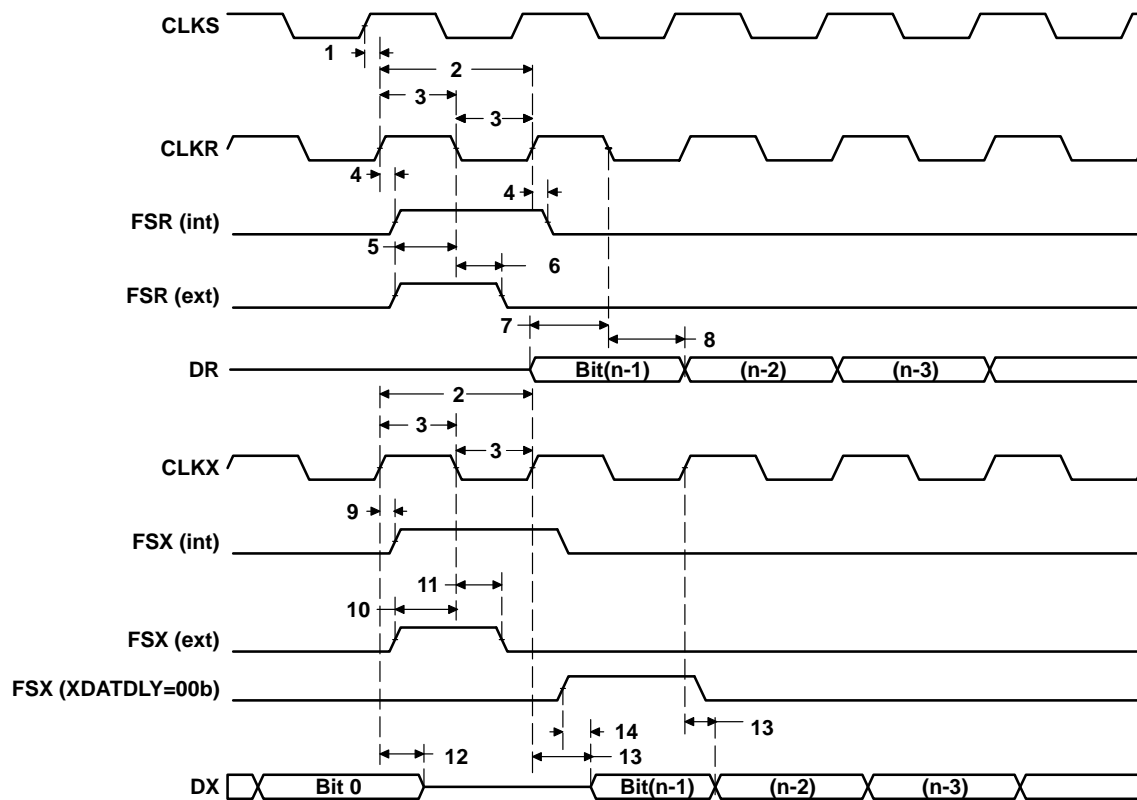


Figure 32. McBSP Timings

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

timing requirements for FSR when GSYNC = 1 (see Figure 33)

NO.		C6201B		UNIT
		MIN	MAX	
1	$t_{su}(FRH-CKSH)$ Setup time, FSR high before CLKS high	4		ns
2	$t_h(CKSH-FRH)$ Hold time, FSR high after CLKS high	4		ns

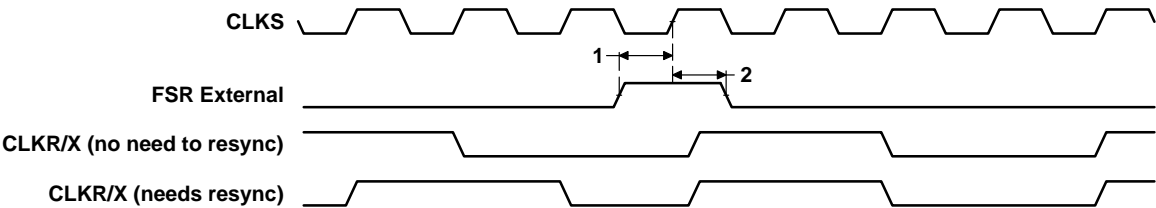


Figure 33. FSR Timing When GSYNC = 1

timing requirements for McBSP as SPI master or slave: CLKSTP = 10b, CLKXP = 0<sup>†‡</sup> (see Figure 34)

NO.		C6201B				UNIT
		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	
4	$t_{su}(DRV-CKXL)$ Setup time, DR valid before CLKX low	12		2 – 3P	ns	
5	$t_h(CKXL-DRV)$ Hold time, DR valid after CLKX low	4		5 + 6P	ns	

<sup>†</sup> P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

<sup>‡</sup> For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

switching characteristics for McBSP as SPI master or slave: CLKSTP = 10b, CLKXP = 0†‡  
(see Figure 34)

NO.	PARAMETER		C6201B				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	t <sub>h</sub> (CKXL-FXL)	Hold time, FSX low after CLKX low¶	T – 2	*T + 3			ns
2	t <sub>d</sub> (FXL-CKXH)	Delay time, FSX low to CLKX high#	*L – 2	L + 3			ns
3	t <sub>d</sub> (CKXH-DXV)	Delay time, CLKX high to DX valid	*–2	4	*3P + 4	5P + 17	ns
6	t <sub>dis</sub> (CKXL-DXHZ)	Disable time, DX high impedance following last data bit from CLKX low	*L – 2	*L + 3			ns
7	t <sub>dis</sub> (FXH-DXHZ)	Disable time, DX high impedance following last data bit from FSX high			*P + 3	*3P + 17	ns
8	t <sub>d</sub> (FXL-DXV)	Delay time, FSX low to DX valid			*2P + 2	4P + 17	ns

†  $P = 1/\text{CPU clock frequency}$  in ns. For example, when running parts at 200 MHz, use  $P = 5$  ns.

‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting  $\text{CLKSM} = \text{CLKGDV} = 1$ .

§  $S = \text{sample rate generator input clock} = P$  if  $\text{CLKSM} = 1$  ( $P = 1/\text{CPU clock frequency}$ )

$= \text{sample rate generator input clock} = P\_clks$  if  $\text{CLKSM} = 0$  ( $P\_clks = \text{CLKS period}$ )

$T = \text{CLKX period} = (1 + \text{CLKGDV}) * S$

$H = \text{CLKX high pulse width} = (\text{CLKGDV}/2 + 1) * S$  if  $\text{CLKGDV}$  is even

$= (\text{CLKGDV} + 1)/2 * S$  if  $\text{CLKGDV}$  is odd or zero

$L = \text{CLKX low pulse width} = (\text{CLKGDV}/2) * S$  if  $\text{CLKGDV}$  is even

$= (\text{CLKGDV} + 1)/2 * S$  if  $\text{CLKGDV}$  is odd or zero

¶  $\text{FSRP} = \text{FSXP} = 1$ . As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.

$\text{CLKXM} = \text{FSXM} = 1$ ,  $\text{CLKRM} = \text{FSRM} = 0$  for master McBSP

$\text{CLKXM} = \text{CLKRM} = \text{FSXM} = \text{FSRM} = 0$  for slave McBSP

\*Not production tested.

# FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

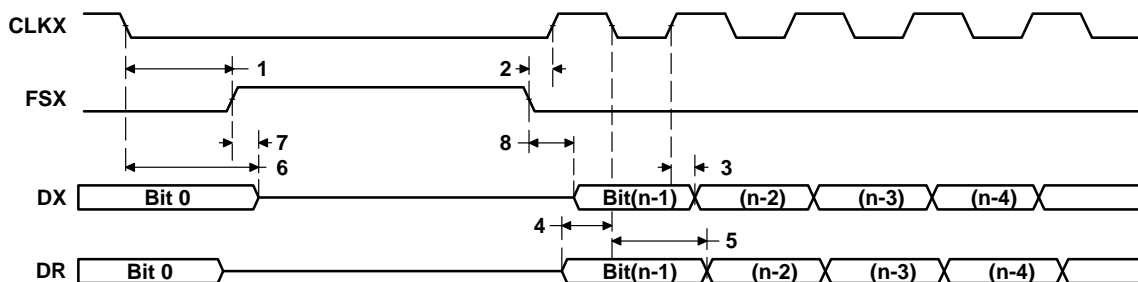


Figure 34. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 0

¶  $\text{FSRP} = \text{FSXP} = 1$ . As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.

$\text{CLKXM} = \text{FSXM} = 1$ ,  $\text{CLKRM} = \text{FSRM} = 0$  for master McBSP

$\text{CLKXM} = \text{CLKRM} = \text{FSXM} = \text{FSRM} = 0$  for slave McBSP

# FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

**MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)**

**timing requirements for McBSP as SPI master or slave: CLKSTP = 11b, CLKXP = 0†‡ (see Figure 35)**

NO.		C6201B				UNIT
		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	
4	$t_{su}(DRV-CKXH)$ Setup time, DR valid before CLKX high	12		2 – 3P		ns
5	$t_h(CKXH-DRV)$ Hold time, DR valid after CLKX high	4		5 + 6P		ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

‡ For all SPI slave modes, CLKX is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

**switching characteristics for McBSP as SPI master or slave: CLKSTP = 11b, CLKXP = 0†‡ (see Figure 35)**

NO.	PARAMETER		C6201B				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	t <sub>h</sub> (CKXL-FXL)	Hold time, FSX low after CLKX low¶	L – 2	*L + 3			ns
2	t <sub>d</sub> (FXL-CKXH)	Delay time, FSX low to CLKX high#	*T – 2	T + 3			ns
3	t <sub>d</sub> (CKXL-DXV)	Delay time, CLKX low to DX valid	*–2	4	*3P + 4	5P + 17	ns
6	t <sub>dis</sub> (CKXL-DXHZ)	Disable time, DX high impedance following last data bit from CLKX low	*–2	*4	*3P + 3	*5P + 17	ns
7	t <sub>d</sub> (FXL-DXV)	Delay time, FSX low to DX valid	*H – 2	H + 4	*2P + 2	4P + 17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

‡ For all SPI slave modes, CLKX is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

§ S = sample rate generator input clock = P if CLKSM = 1 (P = 1/CPU clock frequency)

= sample rate generator input clock = P\_clks if CLKSM = 0 (P\_clks = CLKS period)

T = CLKX period = (1 + CLKGDV) \* S

H = CLKX high pulse width = (CLKGDV/2 + 1) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

L = CLKX low pulse width = (CLKGDV/2) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

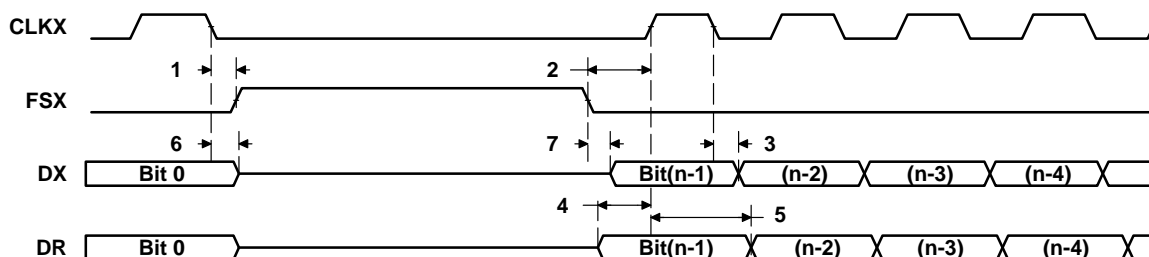
¶ FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.

CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP

CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

\*Not production tested.

# FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).



**Figure 35. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 0**



MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

timing requirements for McBSP as SPI master or slave: CLKSTP = 10b, CLKXP = 1†‡ (see Figure 36)

NO.		C6201B				UNIT
		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	
4	$t_{su}(DRV-CKXH)$ Setup time, DR valid before CLKX high	12		2 – 3P	ns	
5	$t_h(CKXH-DRV)$ Hold time, DR valid after CLKX high	4		5 + 6P	ns	

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

switching characteristics for McBSP as SPI master or slave: CLKSTP = 10b, CLKXP = 1†‡ (see Figure 36)

NO.	PARAMETER		C6201B				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	t <sub>h</sub> (CKXH-FXL)	Hold time, FSX low after CLKX high¶	T – 2	*T + 3			ns
2	t <sub>d</sub> (FXL-CKXL)	Delay time, FSX low to CLKX low#	*H – 2	H + 3			ns
3	t <sub>d</sub> (CKXL-DXV)	Delay time, CLKX low to DX valid	*–2	4	*3P + 4	5P + 17	ns
6	t <sub>dis</sub> (CKXH-DXHZ)	Disable time, DX high impedance following last data bit from CLKX high	*H – 2	*H + 3			ns
7	t <sub>dis</sub> (FXH-DXHZ)	Disable time, DX high impedance following last data bit from FSX high			*P + 3	*3P + 17	ns
8	t <sub>d</sub> (FXL-DXV)	Delay time, FSX low to DX valid			*2P + 2	4P + 17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

§ S = sample rate generator input clock = P if CLKSM = 1 (P = 1/CPU clock frequency)

= sample rate generator input clock = P\_clks if CLKSM = 0 (P\_clks = CLK period)

T = CLKX period = (1 + CLKGDV) \* S

H = CLKX high pulse width = (CLKGDV/2 + 1) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

L = CLKX low pulse width = (CLKGDV/2) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

¶ FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.

CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP

CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

\*Not production tested.

# FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).

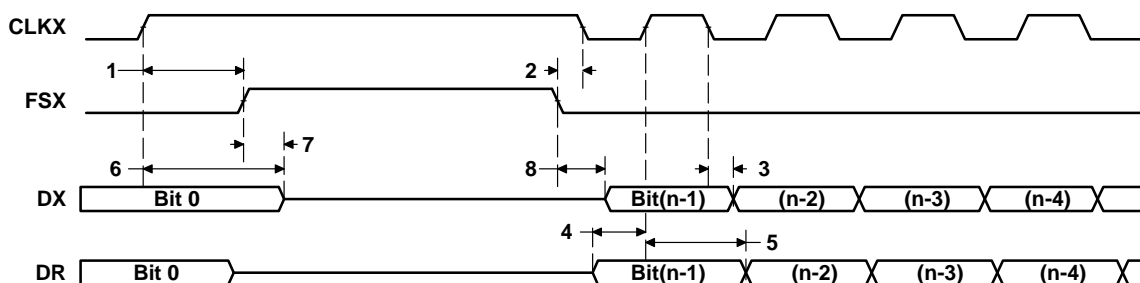


Figure 36. McBSP Timing as SPI Master or Slave: CLKSTP = 10b, CLKXP = 1

**MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)**

**timing requirements for McBSP as SPI master or slave: CLKSTP = 11b, CLKXP = 1†‡ (see Figure 37)**

NO.		C6201B				UNIT
		MASTER		SLAVE		
		MIN	MAX	MIN	MAX	
4	$t_{su}(DRV-CKXL)$ Setup time, DR valid before CLKX low	12		2 – 3P	ns	
5	$t_h(CKXL-DRV)$ Hold time, DR valid after CLKX low	4		5 + 6P	ns	

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

**switching characteristics for McBSP as SPI master or slave: CLKSTP = 11b, CLKXP = 1†‡ (see Figure 37)**

NO.	PARAMETER		C6201B				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	t <sub>h</sub> (CKXH-FXL)	Hold time, FSX low after CLKX high¶	H – 2	*H + 3			ns
2	t <sub>d</sub> (FXL-CKXL)	Delay time, FSX low to CLKX low#	*T – 2	T + 1			ns
3	t <sub>d</sub> (CKXH-DXV)	Delay time, CLKX high to DX valid	*–2	4	*3P + 3	5P + 17	ns
6	t <sub>dis</sub> (CKXH-DXHZ)	Disable time, DX high impedance following last data bit from CLKX high	*–2	*4	*3P + 3	*5P + 17	ns
7	t <sub>d</sub> (FXL-DXV)	Delay time, FSX low to DX valid	*L – 2	L + 4	*2P + 2	4P + 17	ns

† P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1.

§ S = sample rate generator input clock = P if CLKSM = 1 (P = 1/CPU clock frequency)

= sample rate generator input clock = P\_clks if CLKSM = 0 (P\_clks = CLKS period)

T = CLKX period = (1 + CLKGDV) \* S

H = CLKX high pulse width = (CLKGDV/2 + 1) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

L = CLKX low pulse width = (CLKGDV/2) \* S if CLKGDV is even  
= (CLKGDV + 1)/2 \* S if CLKGDV is odd or zero

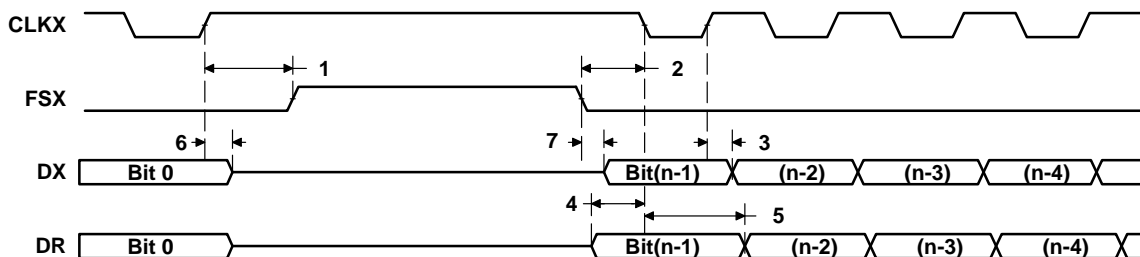
¶ FSRP = FSXP = 1. As a SPI master, FSX is inverted to provide active-low slave-enable output. As a slave, the active-low signal input on FSX and FSR is inverted before being used internally.

CLKXM = FSXM = 1, CLKRM = FSRM = 0 for master McBSP

CLKXM = CLKRM = FSXM = FSRM = 0 for slave McBSP

\*Not production tested.

# FSX should be low before the rising edge of clock to enable slave devices and then begin a SPI transfer at the rising edge of the master clock (CLKX).



**Figure 37. McBSP Timing as SPI Master or Slave: CLKSTP = 11b, CLKXP = 1**

## DMAC, TIMER, POWER-DOWN TIMING

### switching characteristics for DMAC outputs (see Figure 38)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_d(\text{CKO1H-DMACV})$ Delay time, CLKOUT1 high to DMAC valid	*2	10	ns

\*Not production tested.

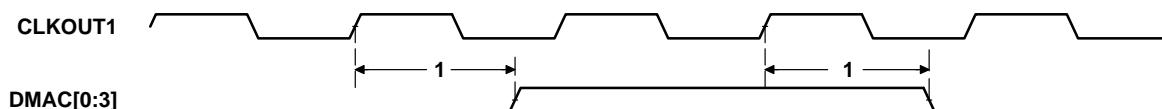


Figure 38. DMAC Timing

### timing requirements for timer inputs<sup>†</sup> (see Figure 39)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_w(\text{TINP})$ Pulse duration, TINP high or low	*2P		ns

<sup>†</sup> P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.

\*Not production tested.

### switching characteristics for timer outputs (see Figure 39)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
2	$t_d(\text{CKO1H-TOUTV})$ Delay time, CLKOUT1 high to TOUT valid	*2	9	ns

\*Not production tested.

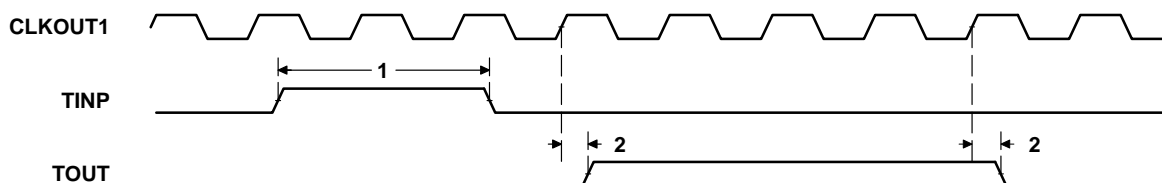


Figure 39. Timer Timing

### switching characteristics for power-down outputs (see Figure 40)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
1	$t_d(\text{CKO1H-PDV})$ Delay time, CLKOUT1 high to PD valid	*2	9	ns

\*Not production tested.

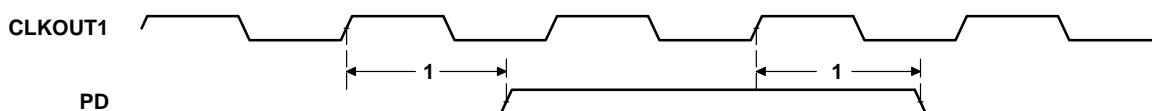


Figure 40. Power-Down Timing

JTAG TEST-PORT TIMING

timing requirements for JTAG test port (see Figure 41)

NO.		C6201B		UNIT
		MIN	MAX	
1	$t_c(\text{TCK})$ Cycle time, TCK	*50		ns
3	$t_{su}(\text{TDIV-TCKH})$ Setup time, TDI/TMS/TRST valid before TCK high	*10		ns
4	$t_h(\text{TCKH-TDIV})$ Hold time, TDI/TMS/TRST valid after TCK high	*5		ns

\*Not production tested.

switching characteristics for JTAG test port (see Figure 41)

NO.	PARAMETER	C6201B		UNIT
		MIN	MAX	
2	$t_d(\text{TCKL-TDOV})$ Delay time, TCK low to TDO valid	*0	*15	ns

\*Not production tested.

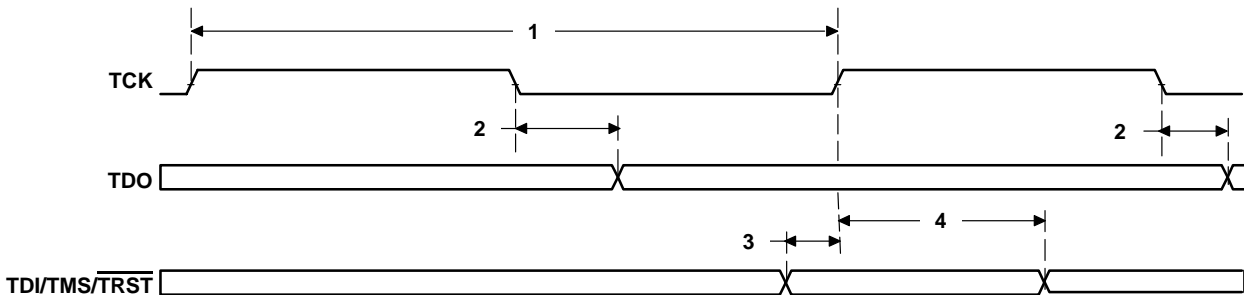
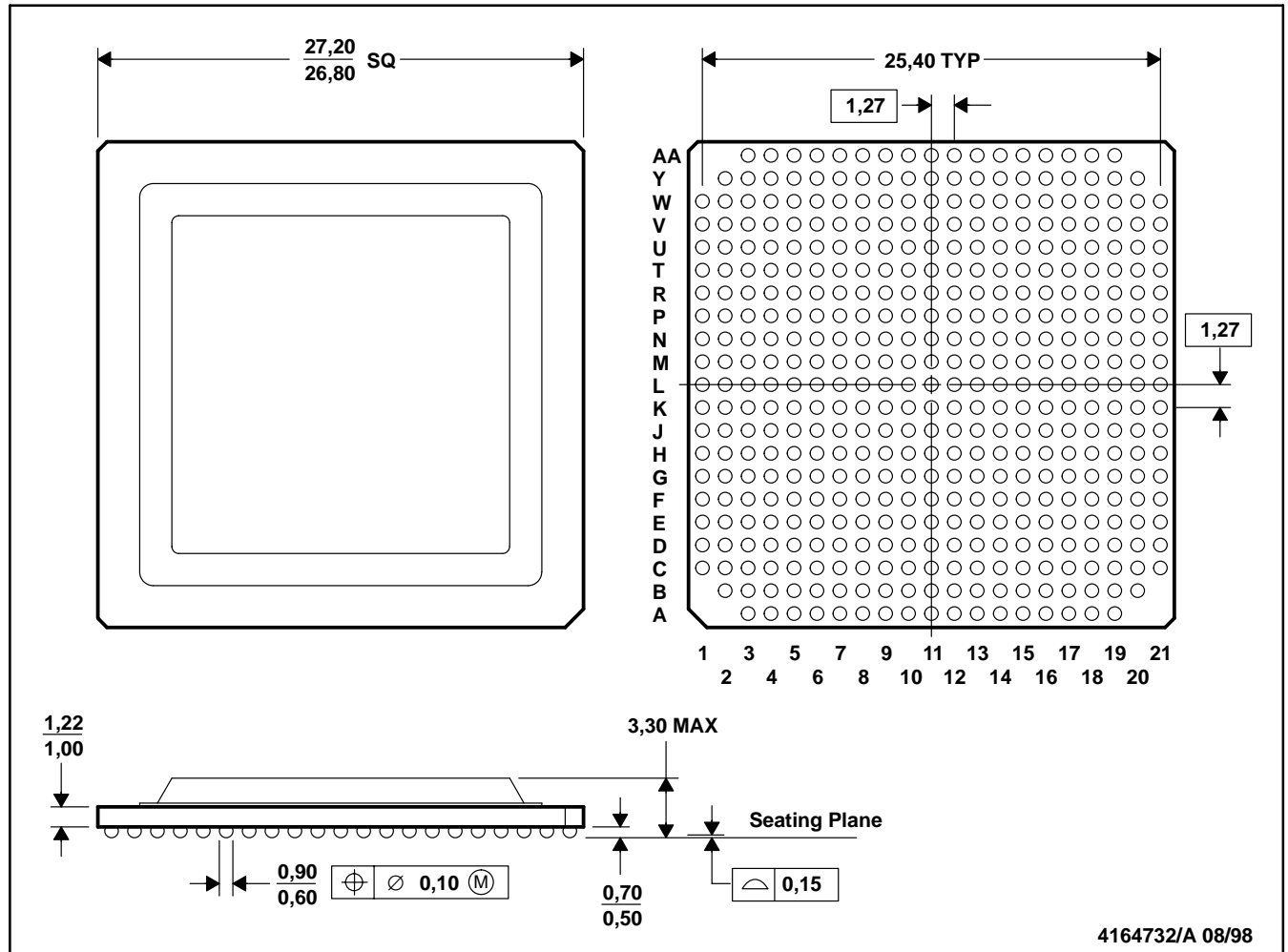


Figure 41. JTAG Test-Port Timing

**MECHANICAL DATA**

**GLP (S-CBGA-N429)**

**CERAMIC BALL GRID ARRAY**



- NOTES:
- A. All linear dimensions are in millimeters.
  - B. This drawing is subject to change without notice.
  - C. Falls within JEDEC MO-156
  - D. Flip chip application only
  - E. For 320C6201B (1.8 V core device).
  - F. Package weight for GLP is 7.65 grams.

**thermal resistance characteristics (S-CBGA package)**

NO		°C/W	Air Flow
1	R $\theta_{JC}$ Junction-to-Case, measured to the bottom of solder ball	3.0	N/A
2	R $\theta_{JC}$ Junction-to-Case, measured to the top of the package lid	7.3	N/A
3	R $\theta_{JA}$ Junction-to-Ambient	14.5	0
4	R $\theta_{JMA}$ Junction-to-Moving-Air	11.8	150 fpm
5		11.1	250 fpm
6		10.2	500 fpm
7	R $\theta_{JB}$ Junction-to-Board, measured by soldering a thermocouple to one of the middle traces on the board at the edge of the package	6.2	N/A

## PACKAGING INFORMATION

Orderable Device	Status <sup>(1)</sup>	Package Type	Package Drawing	Pins	Package Qty	Eco Plan <sup>(2)</sup>	Lead/Ball Finish	MSL Peak Temp <sup>(3)</sup>
5962-9857801QXA	ACTIVE	FC/CSP	GLP	429	1	TBD	Call TI	N / A for Pkg Type
SM320C6201BGLPS20	ACTIVE	FC/CSP	GLP	429	1	TBD	Call TI	N / A for Pkg Type
SM320C6201BGLPW15	ACTIVE	FC/CSP	GLP	429	1	TBD	Call TI	N / A for Pkg Type
SMJ320C6201BGLPW15	ACTIVE	FC/CSP	GLP	429	1	TBD	Call TI	N / A for Pkg Type

<sup>(1)</sup> The marketing status values are defined as follows:

**ACTIVE:** Product device recommended for new designs.

**LIFEBUY:** TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.

**NRND:** Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.

**PREVIEW:** Device has been announced but is not in production. Samples may or may not be available.

**OBSOLETE:** TI has discontinued the production of the device.

<sup>(2)</sup> Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check <http://www.ti.com/productcontent> for the latest availability information and additional product content details.

**TBD:** The Pb-Free/Green conversion plan has not been defined.

**Pb-Free (RoHS):** TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.

**Pb-Free (RoHS Exempt):** This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.

**Green (RoHS & no Sb/Br):** TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

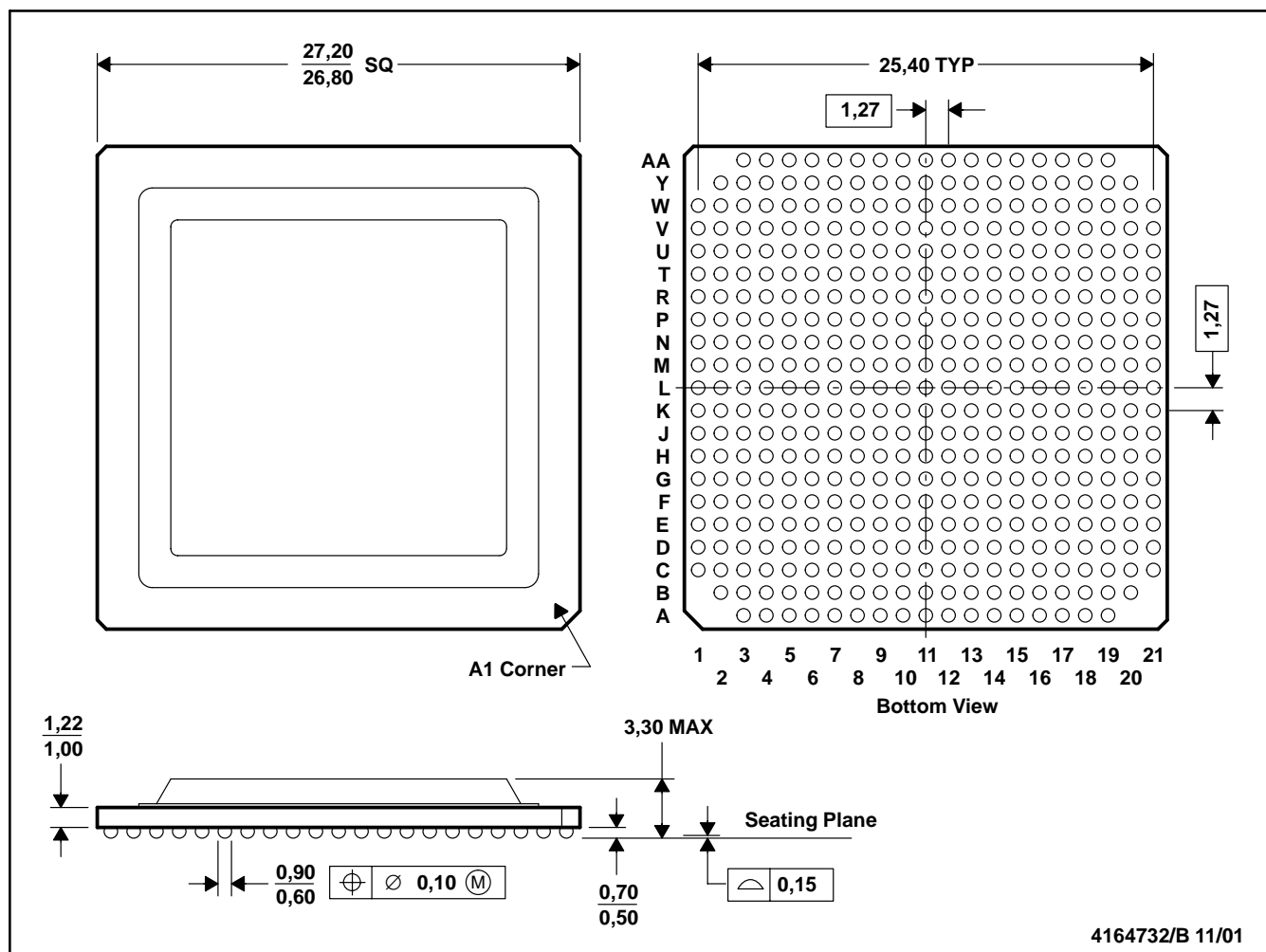
<sup>(3)</sup> MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

**Important Information and Disclaimer:** The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

## GLP (S-CBGA-N429)

## CERAMIC BALL GRID ARRAY



- NOTES: A. All linear dimensions are in millimeters.  
 B. This drawing is subject to change without notice.  
 C. Falls within JEDEC MO-156  
 D. Flip chip application only

## IMPORTANT NOTICE

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

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

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

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

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

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

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

<b>Products</b>		<b>Applications</b>	
Amplifiers	<a href="http://amplifier.ti.com">amplifier.ti.com</a>	Audio	<a href="http://www.ti.com/audio">www.ti.com/audio</a>
Data Converters	<a href="http://dataconverter.ti.com">dataconverter.ti.com</a>	Automotive	<a href="http://www.ti.com/automotive">www.ti.com/automotive</a>
DSP	<a href="http://dsp.ti.com">dsp.ti.com</a>	Broadband	<a href="http://www.ti.com/broadband">www.ti.com/broadband</a>
Interface	<a href="http://interface.ti.com">interface.ti.com</a>	Digital Control	<a href="http://www.ti.com/digitalcontrol">www.ti.com/digitalcontrol</a>
Logic	<a href="http://logic.ti.com">logic.ti.com</a>	Military	<a href="http://www.ti.com/military">www.ti.com/military</a>
Power Mgmt	<a href="http://power.ti.com">power.ti.com</a>	Optical Networking	<a href="http://www.ti.com/opticalnetwork">www.ti.com/opticalnetwork</a>
Microcontrollers	<a href="http://microcontroller.ti.com">microcontroller.ti.com</a>	Security	<a href="http://www.ti.com/security">www.ti.com/security</a>
Low Power Wireless	<a href="http://www.ti.com/lpw">www.ti.com/lpw</a>	Telephony	<a href="http://www.ti.com/telephony">www.ti.com/telephony</a>
		Video & Imaging	<a href="http://www.ti.com/video">www.ti.com/video</a>
		Wireless	<a href="http://www.ti.com/wireless">www.ti.com/wireless</a>

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

Copyright © 2006, Texas Instruments Incorporated