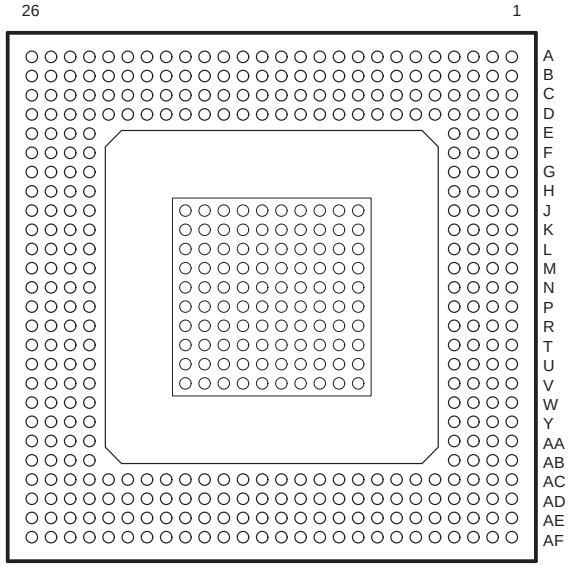
- **Highest Performance Fixed-Point Digital Signal Processor (DSP) TMX320C6201**
  - **5-ns Instruction Cycle Time**
  - **200-MHz Clock Rate**
  - **Eight 32-Bit Instructions/Cycle**
  - **1600 MIPS**
- **VelociTI™ Advanced Very Long Instruction Word (VLIW) 'C62x CPU Core**
  - **Eight Highly Independent 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)**
- **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**

GGP 352-PIN (GJC 452-PIN) BGA PACKAGE
(BOTTOM VIEW)†



† For devices revision 3.0 and higher, a central 10 x 10 grid is present for thermal dissipation and should be connected to GND.

- **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**
  - **SPI-Compatible (Motorola™)**
- **Two 32-Bit General-Purpose Timers**
- **Flexible Phase-Locked-Loop (PLL) Clock Generator**
- **IEEE-1149.1 (JTAG‡) Boundary-Scan Compatible**
- **352-Pin Ball Grid Array (BGA) Package (GGP Suffix)**
- **452-Pin BGA Package (GJC Suffix)**
- **0.25-μm/5-Level Metal Process**
  - **CMOS Technology**
- **3.3-V I/Os, 2.5-V Internal**

ADVANCE INFORMATION

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.

VelociTI is a trademark of Texas Instruments Incorporated.
Motorola is a trademark of Motorola, Inc.
‡ IEEE Standard 1149.1-1990 Standard-Test-Access Port and Boundary Scan Architecture.

Signal Descriptions

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
CLOCK/PLL			
CLKIN	C10	I	Clock Input
CLKOUT1	AF22	O	Clock output at full device speed
CLKOUT2	AF20	O	Clock output at half of device speed
CLKMODE1	C6	I	Clock mode select • Selects whether the output clock frequency = input clock freq x4 or x1
CLKMODE0	C5		
PLLFREQ3	A9	I	PLL frequency range (3, 2, and 1) • Selects one of three frequency ranges bounding the CLKOUT1 signal. • CLKOUT1 frequency determines the 3-bit value for the PLLFREQ pins.
PLLFREQ2	D11		
PLLFREQ1	B10		
PLLV‡	D12	A§	PLL analog V_{CC} connection for the low-pass filter
PLLG‡	C12	A§	PLL analog GND connection for the low-pass filter
PLLF	A11	A§	PLL low-pass filter connection to external components and a bypass capacitor
JTAG EMULATION			
TMS	L3	I	JTAG test port mode select (features an internal pull-up)
TDO	W2	O/Z	JTAG test port data out
TDI	R4	I	JTAG test port data in (features an internal pull-up)
TCK	R3	I	JTAG test port clock
$\overline{\text{TRST}}$	T1	I	JTAG test port reset (features an internal pull-down)
EMU1	Y1	I/O/Z	Emulation pin 1, pull-up with a dedicated 20-kΩ resistor
EMU0	W3	I/O/Z	Emulation pin 0, pull-up with a dedicated 20-kΩ resistor
CONTROL			
$\overline{\text{RESET}}$	K2	I	Device reset
NMI	L2	I	Nonmaskable interrupt • Edge-driven (rising edge)
EXT_INT7	U3	I	External interrupts • Edge-driven (rising edge)
EXT_INT6	V2		
EXT_INT5	W1		
EXT_INT4	U4		
IACK	Y2	O	Interrupt acknowledge for all active interrupts serviced by the CPU
INUM3	AA1	O	Active interrupt identification number • Valid during IACK for all active interrupts (not just external) • Encoding order follows the interrupt service fetch packet ordering
INUM2	W4		
INUM1	AA2		
INUM0	AB1		
LENDIAN	H3	I	If high, selects little-endian byte/half-word addressing order within a word If low, selects big-endian addressing
PD	D3	O	Power-down mode 3 (active if high)

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

‡ PLLV and PLLG 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)

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
HOST PORT INTERFACE (HPI)			
$\overline{\text{HINT}}$	H26	O/Z	Host interrupt (from DSP to host)
HCNTL1	F23	I	Host control – selects between control, address or data registers
HCNTL0	D25	I	Host control – selects between control, address or data registers
HHWIL	C26	I	Host halfword select – first or second halfword (not necessarily high or low order)
$\overline{\text{HBE1}}$	E23	I	Host byte select within word or half-word
$\overline{\text{HBE0}}$	D24	I	Host byte select within word or half-word
HR/$\overline{\text{W}}$	C23	I	Host read or write select
HD15	B13	I/O/Z	Host port data (used for transfer of data, address and control)
HD14	B14		
HD13	C14		
HD12	B15		
HD11	D15		
HD10	B16		
HD9	A17		
HD8	B17		
HD7	D16		
HD6	B18		
HD5	A19		
HD4	C18		
HD3	B19		
HD2	C19		
HD1	B20		
HD0	B21		
$\overline{\text{HAS}}$	C22	I	Host address strobe
$\overline{\text{HCS}}$	B23	I	Host chip select
$\overline{\text{HDS1}}$	D22	I	Host data strobe 1
$\overline{\text{HDS2}}$	A24	I	Host data strobe 2
$\overline{\text{HRDY}}$	J24	O	Host ready (from DSP to host)
BOOT MODE			
BOOTMODE4	D8	I	Boot mode
BOOTMODE3	B4		
BOOTMODE2	A3		
BOOTMODE1	D5		
BOOTMODE0	C4		

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

ADVANCE INFORMATION

ADVANCE INFORMATION

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
EMIF – CONTROL SIGNALS COMMON TO ALL TYPES OF MEMORY			
$\overline{CE3}$	AE22	O/Z	Memory space enables • Enabled by bits 24 and 25 of the word address • Only one asserted during any external data access
$\overline{CE2}$	AD26	O/Z	
$\overline{CE1}$	AB24	O/Z	
$\overline{CE0}$	AC26	O/Z	
$\overline{BE3}$	AB25	O/Z	Byte enable control • Decoded from the two lowest bits of the internal address • Byte write enables for most types of memory • Can be directly connected to SDRAM read and write mask signal (SDQM)
$\overline{BE2}$	AA24	O/Z	
$\overline{BE1}$	Y23	O/Z	
$\overline{BE0}$	AA26	O/Z	
EMIF – ADDRESS			
EA21	J26	O/Z	External address (word address)
EA20	K25		
EA19	L24		
EA18	K26		
EA17	M26		
EA16	M25		
EA15	P25		
EA14	P24		
EA13	R25		
EA12	T26		
EA11	R23		
EA10	U26		
EA9	U25		
EA8	T23		
EA7	V26		
EA6	V25		
EA5	W26		
EA4	V24		
EA3	W25		
EA2	Y26		

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

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
EMIF – DATA			
ED31	AB2	I/O/Z	External data
ED30	AC1		
ED29	AA4		
ED28	AD1		
ED27	AC3		
ED26	AD4		
ED25	AF3		
ED24	AE4		
ED23	AD5		
ED22	AF4		
ED21	AE5		
ED20	AD6		
ED19	AE6		
ED18	AD7		
ED17	AC8		
ED16	AF7		
ED15	AD9		
ED14	AD10		
ED13	AF9		
ED12	AC11		
ED11	AE10		
ED10	AE11		
ED9	AF11		
ED8	AE14		
ED7	AF15		
ED6	AE15		
ED5	AF16		
ED4	AC15		
ED3	AE17		
ED2	AF18		
ED1	AF19		
ED0	AC17		
EMIF – ASYNCHRONOUS MEMORY CONTROL			
$\overline{\text{ARE}}$	Y24	O/Z	Asynchronous memory read enable
$\overline{\text{AOE}}$	AC24	O/Z	Asynchronous memory output enable
$\overline{\text{AWE}}$	AD23	O/Z	Asynchronous Memory write enable
ARDY	W23	I	Asynchronous memory ready input

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

ADVANCE INFORMATION

ADVANCE INFORMATION

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
EMIF – SYNCHRONOUS BURST SRAM CONTROL			
$\overline{\text{SSADS}}$	AC20	O/Z	SBSRAM address strobe
$\overline{\text{SSOE}}$	AF21	O/Z	SBSRAM output enable
$\overline{\text{SSWE}}$	AD19	O/Z	SBSRAM write enable
SSCLK	AD17	O/Z	SBSRAM clock
EMIF – SYNCHRONOUS DRAM CONTROL			
SDA10	AD21	O/Z	SDRAM address 10 (separate for deactivate command)
$\overline{\text{SDRAS}}$	AF24	O/Z	SDRAM row address strobe
$\overline{\text{SDCAS}}$	AD22	O/Z	SDRAM column address strobe
$\overline{\text{SDWE}}$	AF23	O/Z	SDRAM write enable
SDCLK	AE20	O/Z	SDRAM clock
EMIF – BUS ARBITRATION			
$\overline{\text{HOLD}}$	AA25	I	Hold request from the host
$\overline{\text{HOLDA}}$	A7	O	Hold request acknowledge to the host
TIMERS			
TOUT1	H24	O/Z	Timer 1 or general-purpose output
TINP1	K24	I	Timer 1 or general-purpose input
TOUT0	M4	O/Z	Timer 0 or general-purpose output
TINP0	K4	I	Timer 0 or general-purpose input
DMA ACTION COMPLETE			
DMAC3	D2	O	DMA action complete
DMAC2	F4		
DMAC1	D1		
DMAC0	E2		
MULTICHANNEL BUFFERED SERIAL PORT 1 (McBSP1)			
CLKS1	E25	I	External clock source (as opposed to internal)
CLKR1	H23	I/O/Z	Receive clock
CLKX1	F26	I/O/Z	Transmit clock
DR1	D26	I	Receive data
DX1	G23	O/Z	Transmit data
FSR1	E26	I/O/Z	Receive frame sync
FSX1	F25	I/O/Z	Transmit frame sync

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

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
MULTICHANNEL BUFFERED SERIAL PORT 0 (McBSP0)			
CLKS0	L4	I	External clock source (as opposed to internal)
CLKR0	M2	I/O/Z	Receive clock
CLKX0	L1	I/O/Z	Transmit clock
DR0	J1	I	Receive data
DX0	R1	O/Z	Transmit data
FSR0	P4	I/O/Z	Receive frame sync
FSX0	P3	I/O/Z	Transmit frame sync
RESERVED FOR TEST			
RSV0	T2	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV1	G2	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV2	C11	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV3	B9	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV4	A6	I	Reserved for testing, ***pull-down*** with a dedicated 20-kΩ resistor
RSV5	C8	O	Reserved (leave unconnected, ***do not*** connect to power or ground)
RSV6	C21	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV7	B22	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
RSV8	A23	I	Reserved for testing, pull-up with a dedicated 20-kΩ resistor
SUPPLY VOLTAGE PINS			
DV_{DD}	A10	S	3.3-V supply voltage
	A15		
	A18		
	A21		
	A22		
	B7		
	C1		
	D17		
	F3		
	G24		
	G25		
	H25		
	J25		
	L25		
	M3		
	N3		
	N23		
	R26		
	T24		
	U24		
	W24		

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

ADVANCE INFORMATION

ADVANCE INFORMATION

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
DV_{DD}	Y4	S	3.3-V supply voltage
	AB3		
	AB4		
	AB26		
	AC6		
	AC10		
	AC19		
	AC21		
	AC22		
	AC25		
	AD11		
	AD13		
	AD15		
	AD18		
	AE18		
	AE21		
	AF5		
	AF6		
	AF17		
CV_{DD}	A5	S	2.5-V supply voltage
	A12		
	A16		
	A20		
	B2		
	B6		
	B11		
	B12		
	B25		
	C3		
	C15		
	C20		
	C24		
	D4		
	D6		
	D7		
	D9		
	D14		
	D18		
	D20		

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

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
SUPPLY VOLTAGE PINS (CONTINUED)			
CV_{DD}	D23	S	2.5-V supply voltage
	E1		
	F1		
	H4		
	J4		
	J23		
	K1		
	K23		
	M1		
	M24		
	N4		
	N25		
	P2		
	P23		
	T3		
	T4		
	U1		
	V4		
	V23		
	AC4		
	AC9		
	AC12		
	AC13		
	AC18		
	AC23		
	AD3		
	AD8		
	AD14		
	AD24		
	AE2		
	AE8		
	AE12		
	AE25		
	AF12		

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

ADVANCE INFORMATION

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
GROUND PINS			
V_{SS}	A1	GND	Ground pins
	A2		
	A4		
	A13		
	A14		
	A25		
	A26		
	B1		
	B3		
	B5		
	B24		
	B26		
	C2		
	C7		
	C13		
	C16		
	C17		
	C25		
	D13		
	D19		
	E3		
	E24		
	F2		
	F24		
	G3		
	G4		
	G26		
	J3		
	L23		
	L26		
	M23		
	N1		
	N2		
	N24		
	N26		
	P1		
	P26		
	R24		
	T25		

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

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
GROUND PINS (CONTINUED)			
V_{SS}	U2	GND	Ground pins
	U23		
	V1		
	V3		
	Y3		
	Y25		
	AA3		
	AA23		
	AB23		
	AC2		
	AC5		
	AC7		
	AC14		
	AC16		
	AD2		
	AD12		
	AD16		
	AD20		
	AD25		
	AE1		
	AE3		
	AE7		
	AE9		
	AE13		
	AE16		
	AE19		
	AE23		
	AE24		
	AE26		
	AF1		
	AF2		
	AF8		
	AF10		
	AF13		
	AF14		
	AF25		
	AF26		

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

ADVANCE INFORMATION

ADVANCE INFORMATION

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
THERMAL DISSIPATION PINS			
THB	J9, J10, J11, J12, J13, J14, J15, J16, J17, J18, K9, K10, K11, K12, K13, K14, K15, K16, K17, K18, L9, L10, L11, L12, L13, L14, L15, L16, L17, L18, M9, M10, M11, M12, M13, M14, M15, M16		**For GJC package only.** Thermal dissipation pins (applicable only for revision 3.0 and higher). The thermal dissipation pins make up a central 10 x 10 grid array and should be connected to GND. (Note: For revision 2.0 designs to be compatible with future revisions of ’C62x, a 10 x 10 array of individual pads should be connected to the ground plane to channel the heat away from the thermal balls into the board.)

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

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
THERMAL DISSIPATION PINS (CONTINUED)			
THB	M17 M18 N9 N10 N11 N12 N13 N14 N15 N16 N17 N18 P9 P10 P11 P12 P13 P14 P15 P16 P17 P18 R9 R10 R11 R12 R13 R14 R15 R16 R17 R18 T9 T10 T11 T12 T13 T14		**For GJC package only.** Thermal dissipation pins (applicable only for revision 3.0 and higher). The thermal dissipation pins make up a central 10 x 10 grid array and should be connected to GND. (Note: For revision 2.0 designs to be compatible with future revisions of ’C62x, a 10 x 10 array of individual pads should be connected to the ground plane to channel the heat away from the thermal balls into the board.)

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

ADVANCE INFORMATION

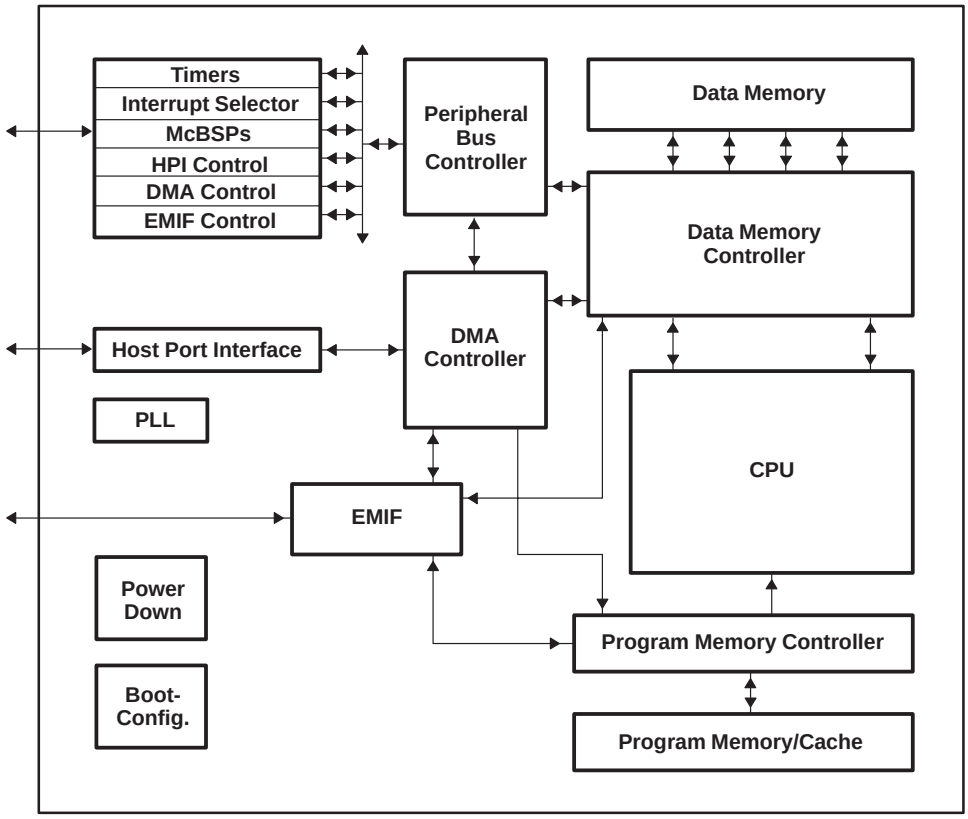
ADVANCE INFORMATION

Signal Descriptions (Continued)

SIGNAL NAME	SIGNAL NO.	TYPE†	DESCRIPTION
THERMAL DISSIPATION PINS (CONTINUED)			
THB	T15 T16 T17 T18 U9 U10 U11 U12 U13 U14 U15 U16 U17 U18 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18		**For GJC package only.** Thermal dissipation pins (applicable only for revision 3.0 and higher). The thermal dissipation pins make up a central 10 x 10 grid array and should be connected to GND. (Note: For revision 2.0 designs to be compatible with future revisions of ’C62x, a 10 x 10 array of individual pads should be connected to the ground plane to channel the heat away from the thermal balls into the board.)
REMAINING UNCONNECTED PINS			
NC	A8 B8 C9 D10 D21 E4 G1 H1 H2 J2 K3 R2		Unconnected pins

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

functional block diagram



signal groups

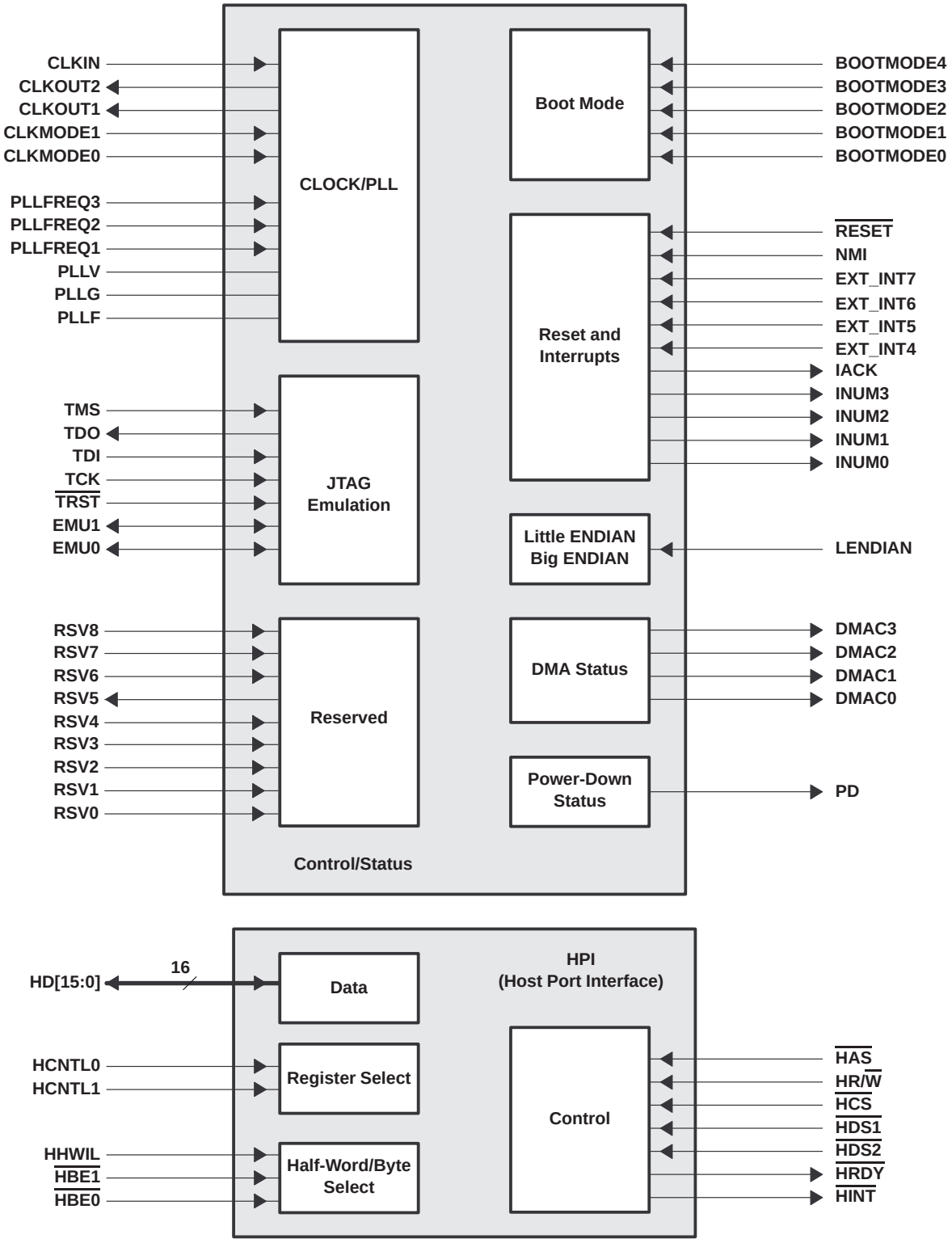


Figure 1. CPU and Peripheral Signals

signal groups (continued)

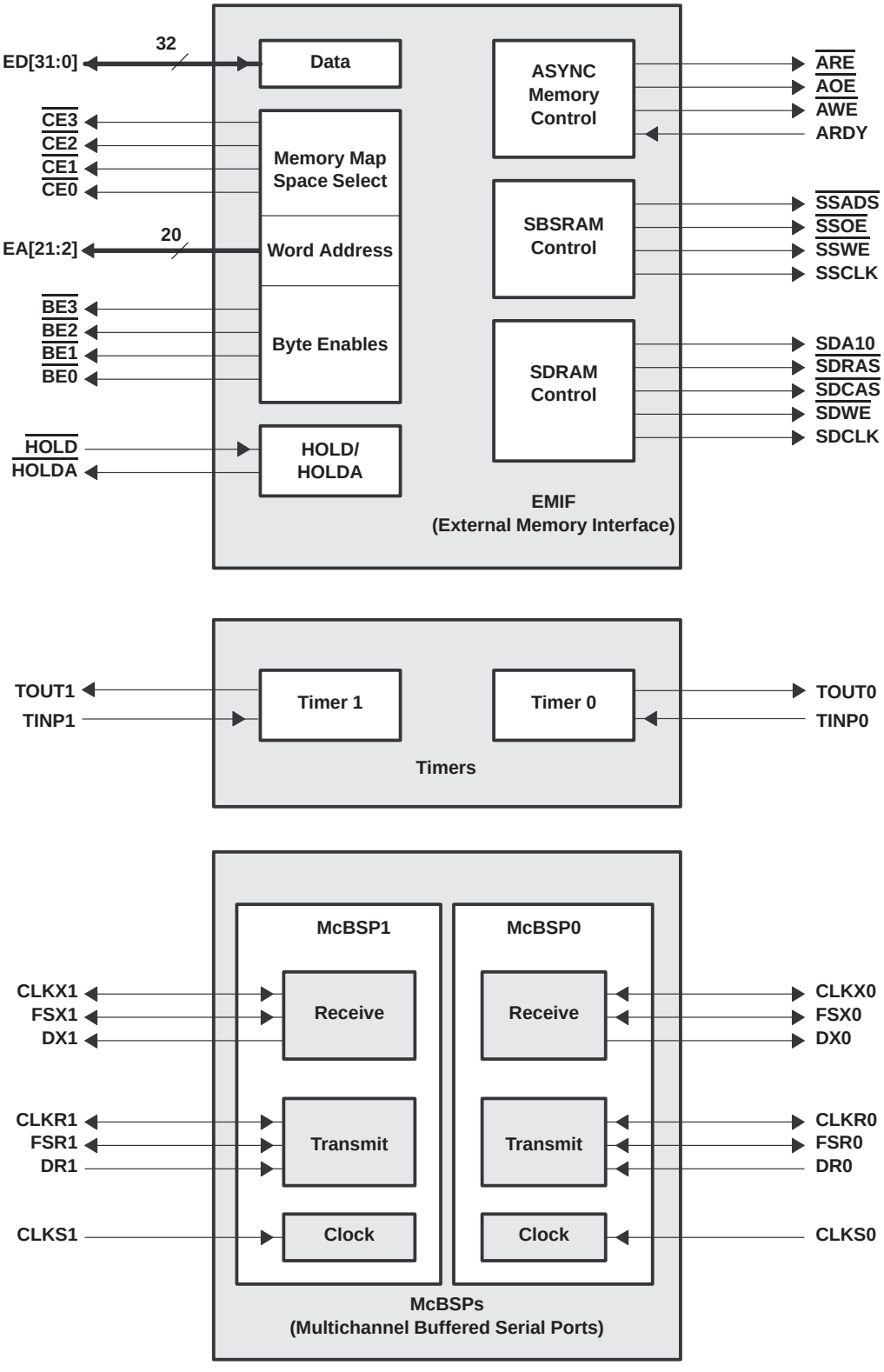


Figure 2. Peripheral Signals

ADVANCE INFORMATION

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 don't 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 contain 16 32-bit registers each for the total of 32 general-purpose registers. The two sets of functional units, along with two register files, comprise sides A and B of the CPU (see Figure 3 and Figure 4). 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 registers on the other side, by which the two sets of functional units can access data from the register files on opposite sides. 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) comprise 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)

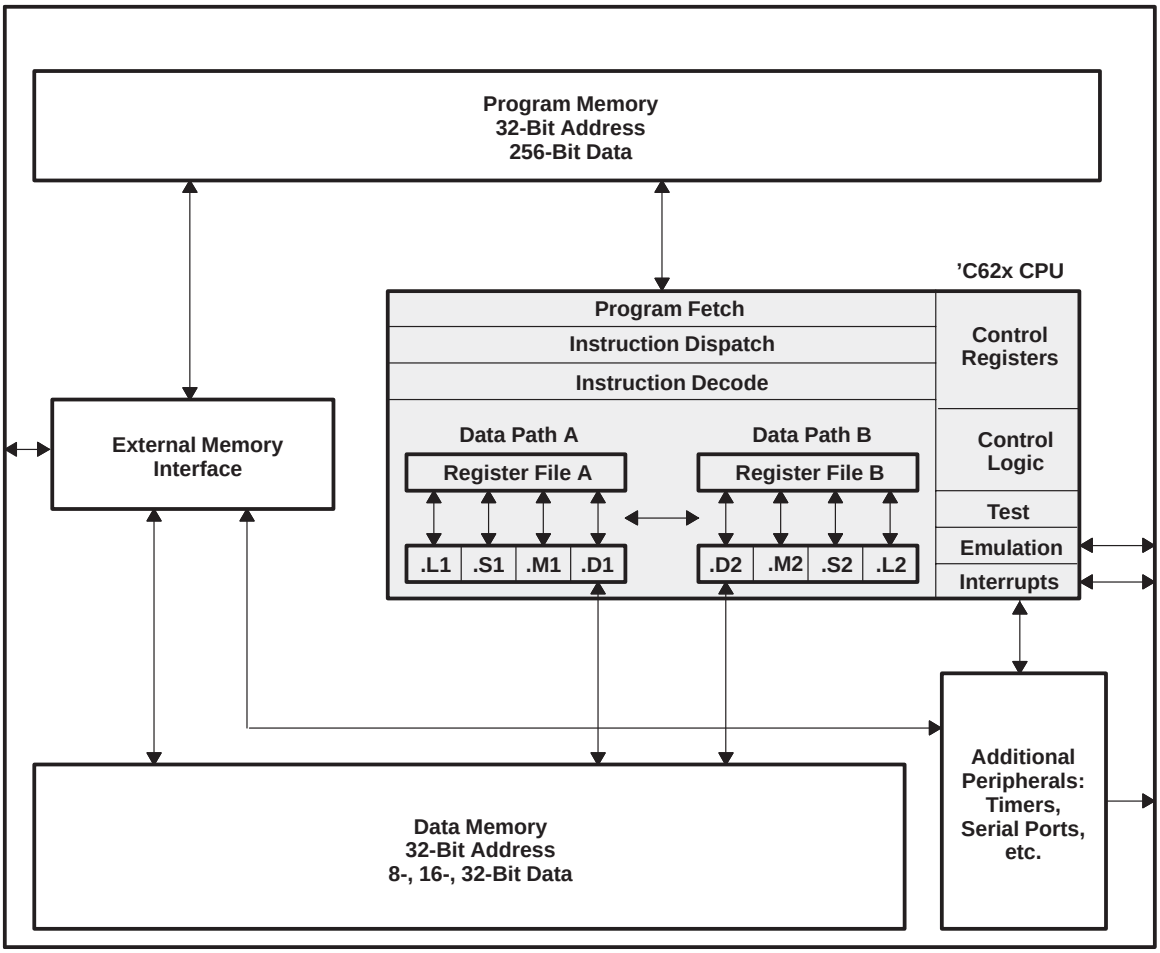


Figure 3. CPU Block Diagram

CPU description (continued)

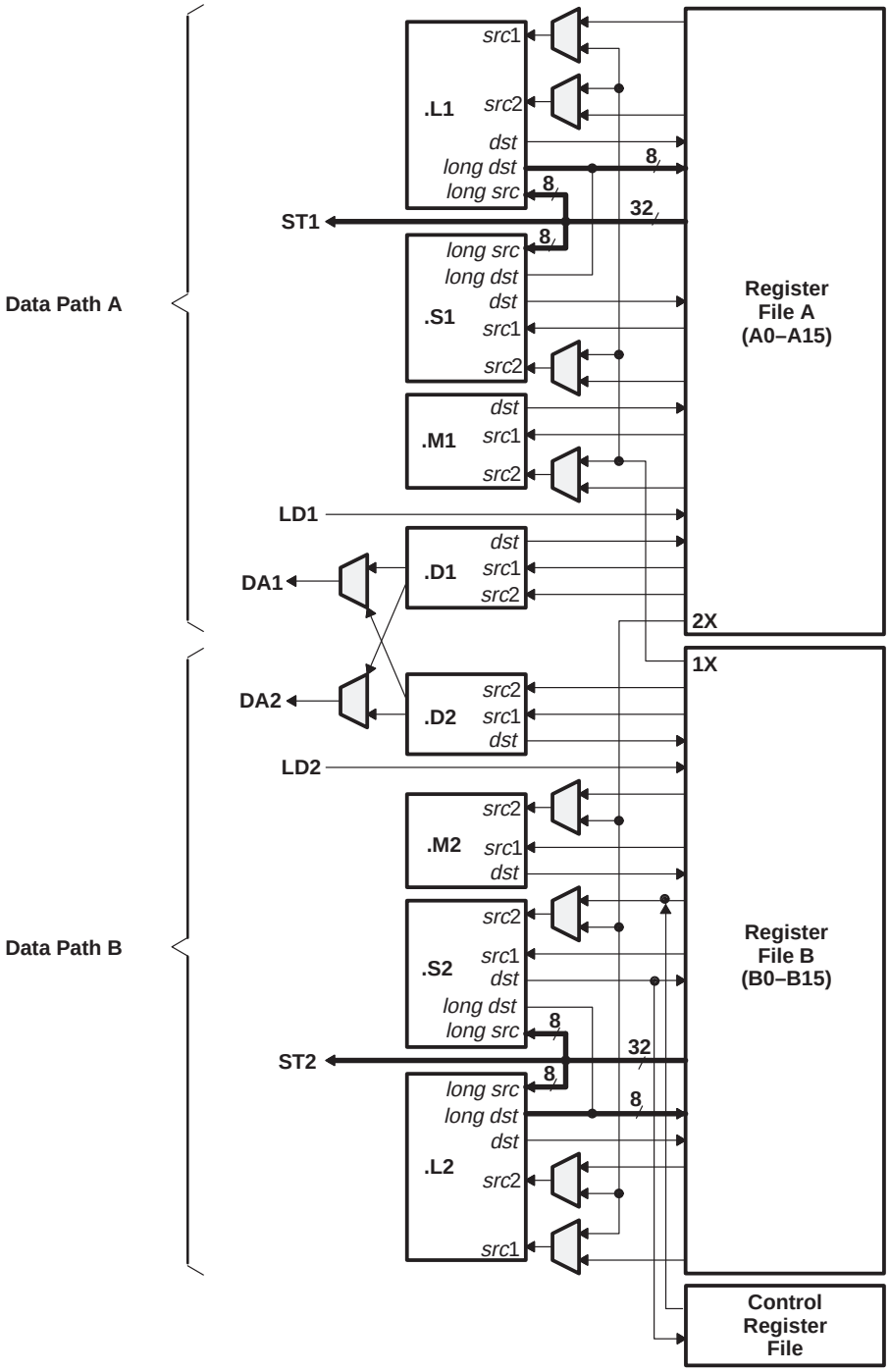


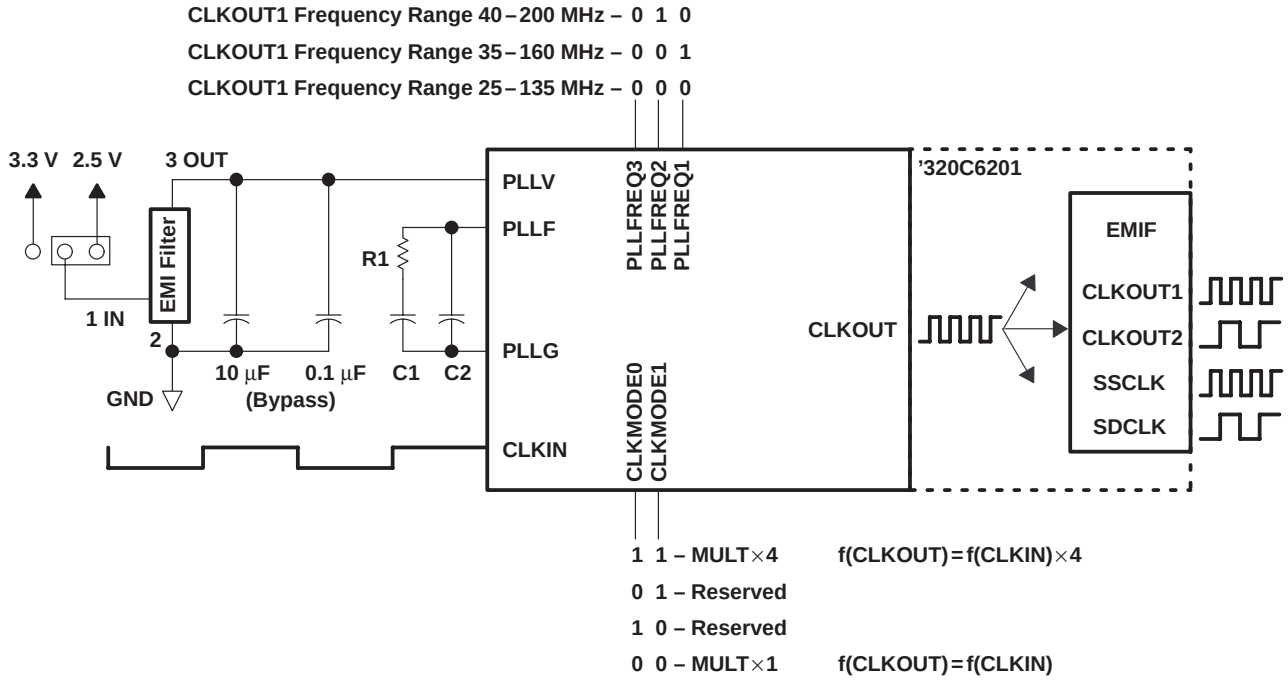
Figure 4. TMX320C62x CPU Data Paths

clock PLL

All of the ’C62xx 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 5 must be properly designed. Note that for revision 2.x silicon, the EMI filter must be powered by the core voltage (2.5 V), and for revision 3.x silicon, it must be powered by the I/O voltage (3.3 V).

To configure the ’C62x PLL clock for proper operation, see Figure 5 and Table 1. In order 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. Please refer to the section titled “input and output clocks” for input clock timing requirements.



NOTES: A. For CLKMODE x4, values for C1, C2, and R2 depend on CLKIN and CLKOUT frequencies.
B. For CLKMODE x1, the PLL is by-passed and all six external PLL components can be removed. For this case, the PLLV terminal has to be connected to a clean 2.5-V supply and the PLLG and PLLF terminals should be tied together.
C. 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 including the desired frequency. For example, for CLKOUT1 = 133 MHz, choose PLLFREQ value of 000b. For CLKOUT1 = 166 or 200 MHz, choose PLLFREQ value of 010b. PLLFREQ values other than 000b, 001b, and 010b are reserved.
D. EMI filter manufacturer TDK part number ACF451832-153-T.

Figure 5. PLL Block Diagram

clock PLL (continued)

Table 1. PLL Component Selection Table†

CYCLE TIME (ns)	CLKMODE	CLKIN (MHz)	CLKOUT1 (MHz)	R1 (Ω)	C1 (μF)	C2 (pF)	EMI FILTER PART NO.‡	LOCK TIME (μs)§
5	x4	50	200	16.9	0.15	2700	TDK #153	59
5.5	x4	45.5	181.8	13.7	0.18	3900	TDK #153	49
6	x4	41.6	166.7	17.4	0.15	3300	TDK #153	68
6.5	x4	38.5	153.8	16.2	0.18	3900	TDK #153	70
7	x4	35.7	142.9	15	0.22	3900	TDK #153	72
7.5	x4	33.3	133.3	16.2	0.22	3900	TDK #153	84
8	x4	31.3	125	14	0.27	4700	TDK #153	77
8.5	x4	29.4	117.7	11.8	0.33	6800	TDK #153	67
9	x4	27.7	111.1	11	0.39	6800	TDK #153	68
9.5	x4	26.3	105.3	10.5	0.39	8200	TDK #153	65
10	x4	25	100	10	0.47	8200	TDK #153	68

† 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 2.5-V supply and the PLLG and PLLF terminals should be tied together.
‡ Full EMI filter part number : ACF 451832-153-T
§ 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 μs, its maximum value may be as long as 250 μs.

power supply sequencing

Since there are two supplies, an additional issue to consider is power-up sequencing. Ideally, the core voltage (CV_{DD}) should be powered up first, followed by the I/O supply voltage (DV_{DD}). In some cases, this may not be practical. If the two power supplies may not be powered up in the above order, then care must be taken to ensure that I/O supply voltage (DV_{DD}) does not exceed the core voltage (CV_{DD}) by more than 2 V. It is also required that the two supplies achieve 95% levels within 25 ms.

ADVANCE INFORMATION

development support

Texas Instruments (TI™) offers an extensive line of development tools for the ’C6x 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 ’C6x-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 ’C6x 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. Refer to 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 Literature Response Center at 800/477-8924.

See Table 2 for a complete listing of development support tools for the ’C6x. For information on pricing and availability, contact the nearest TI field sales office or authorized distributor.

Table 2. TMS320C6x Development Support Tools

DEVELOPMENT TOOL	PLATFORM	PART NUMBER
Software		
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
Hardware		
XDS510 Emulator†	PC	TMDS00510
XDS510WS™ Emulator‡	SCSI	TMDS00510WS
Software/Hardware		
EVM Evaluation Kit	PC/Win95/Windows NT	TMDX3260A6201
EVM Evaluation Kit (including TMDX3246855–07)	PC/Win95/Windows NT	TMDX326006201

† Includes XDS510 board and JTAG emulation cable. TMDX324016X-07 C-source Debugger/Emulation software not included.
‡ Includes XDS510WS box, SCSI cable, power supply, and JTAG emulation cable.

ADVANCE INFORMATION

device and development support tool nomenclature

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

Device development evolutionary flow:

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

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

TMS Fully-qualified production device

Support tool development evolutionary flow:

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

TMDS Fully-qualified development support product

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

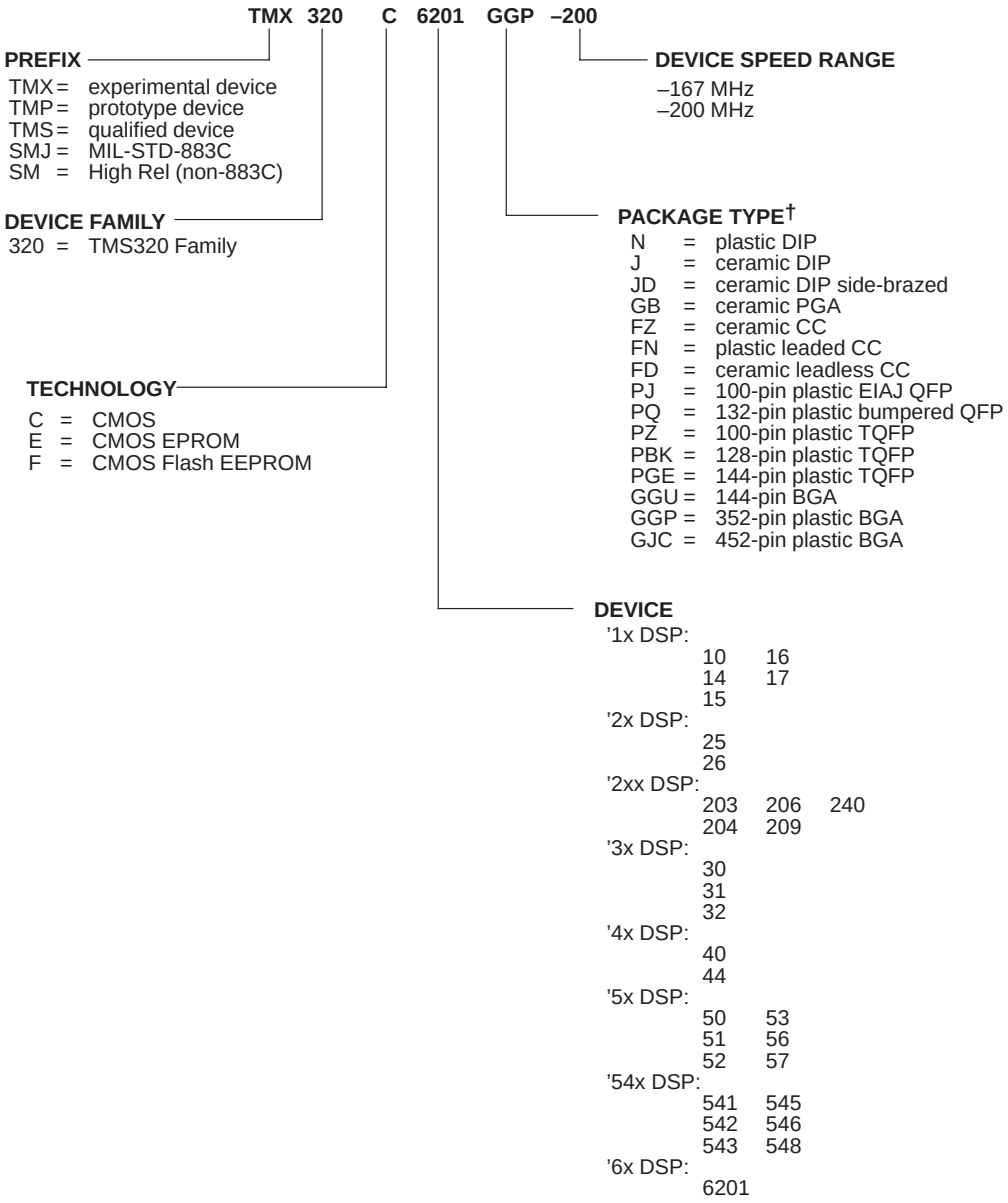
"Developmental product is intended for internal evaluation purposes."

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

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

TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, GGP or GJC) and the device speed range in megahertz (for example, -200 is 200 MHz). Figure 6 provides a legend for reading the complete device name for any TMS320 family member.

device and development support tool nomenclature (continued)



† DIP = Dual-In-Line Package
PGA = Pin Grid Array
CC = Chip Carrier
QFP = Quad Flat Package
TQFP = Thin Quad Flat Package
BGA = Ball Grid Array

Figure 6. TMS320 Device Nomenclature (Including TMX320C6201)

ADVANCE INFORMATION

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 *TMS320C62x/C67x CPU and Instruction Set Reference Guide* (literature number SPRU189) describes the 'C62x/C67x CPU architecture, instruction set, pipeline, associated interrupts.

The *TMS320C6201/C6701 Peripherals Reference Guide* (literature number SPRU190) describes functionally the peripherals available on 'C6x devices, such as the external memory interface (EMIF), host port interface (HPI), multichannel buffered serial ports (McBSP), direct memory access (DMA) controller, clocking and phase-locked loop (PLL); and power-down modes. This guide also includes information on internal data and program memories.

The *TMS320C62x/C67x Programmer's Guide* (literature number SPRU198) describes ways to optimize C and assembly code for 'C6x devices and includes application program examples.

The *TMS320C6x Optimizing C Compiler User's Guide* (literature number SPRU187) describes the 'C6x C compiler and the assembly optimizer, explaining that the C compiler accepts ANSI standard C source code and produces assembly language source code for the 'C6x generation devices, and that the assembly optimizer helps to optimize the programmer's assembly code.

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.

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.

The *TMS320C62x/C67x Technical Brief* (literature number SPRU197) gives an introduction to the 'C62x/C67x devices, associated development tools, and third-party support.

For general background information on DSPs and TI devices, see the three-volume publication *Digital Signal Processing Applications with the TMS320 Family* (literature numbers SPRA012, SPRA016, and SPRA017).

A series of DSP textbooks is published by Prentice-Hall and John Wiley & Sons to support digital signal processing research and education. The TMS320 newsletter, *Details on Signal Processing*, is published quarterly and distributed to update TMS320 customers on product information. The TMS320 DSP bulletin board service (BBS) provides access to information pertaining to the TMS320 family, including documentation, source code and object code for many DSP algorithms and utilities. The BBS can be reached at 281/274-2323.

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

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

Supply voltage range, CV_{DD} (see Note 1) . . . . . –0.3 V to 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 . . . . . 0°C to 90°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

		MIN	NOM	MAX	UNIT
CV_{DD}	Supply voltage‡	2.38	2.50	2.62	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	Case temperature	0		90	°C

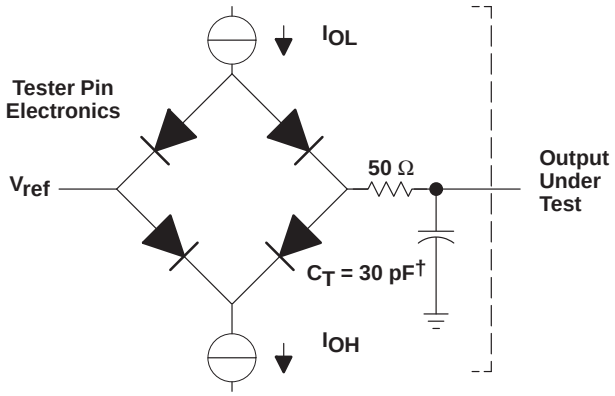
‡ CV_{DD} will be 1.8 V ± 5% in Revision 3.0 and above.

electrical characteristics over recommended ranges of supply voltage and operating case temperature

PARAMETER		TEST CONDITIONS		MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage	DV_{DD} = MIN,	I_{OH} = MAX	2.4			V
V_{OL}	Low-level output voltage	DV_{DD} = MIN,	I_{OL} = MAX			0.6	V
I_I	Input current	$V_I = V_{SS}$ to DV_{DD}				±10	uA
I_{OZ}	Off-state output current	$V_O = DV_{DD}$ or 0 V				±10	uA
I_{DD2V}	Supply current	CV_{DD} = MAX,	CLKOUT1 = 200 MHz		1.8		A
I_{DD3V}	Supply current	DV_{DD} = MAX,	CLKOUT1 = 200 MHz		258		mA
C_I	Input capacitance					10	pF
C_O	Output capacitance					10	pF

ADVANCE INFORMATION

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.

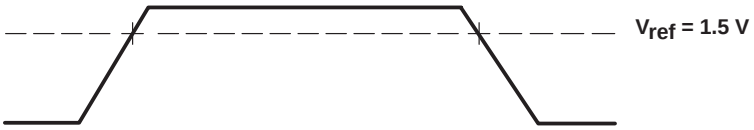


Figure 8. Input and Output Voltage Reference Levels for AC Timing Measurements

ADVANCE INFORMATION

INPUT AND OUTPUT CLOCKS

CLKIN timing parameters (see Figure 9)

NO.			’C6201-167				’C6201-200				UNIT
			CLKMODE = x4		CLKMODE = x1		CLKMODE = x4		CLKMODE = x1		
			MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	
1	$t_{c(CLKIN)}$	Cycle time, CLKIN	24		6		20		5		ns
2	$t_{w(CLKINH)}$	Pulse duration, CLKIN high	9.8		2.7		8		2.25		ns
3	$t_{w(CLKINL)}$	Pulse duration, CLKIN low	9.8		2.7		8		2.25		ns
4	$t_{t(CLKIN)}$	Transition time, CLKIN†		5		0.6		5		0.6	ns

† Values specified from characterization data and not tested.

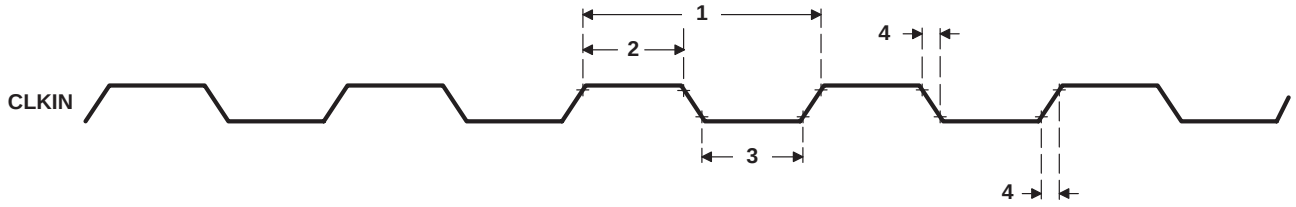


Figure 9. CLKIN Timing

CLKOUT1 timing parameters‡§ (see Figure 10)

NO.			’C6201-167 ’C6201-200				UNIT
			CLKMODE = x4		CLKMODE = x1		
			MIN	MAX	MIN	MAX	
1	$t_{c(CKO1)}$	Cycle time, CLKOUT1	P – 0.7	P + 0.7	P – 0.7	P + 0.7	ns
2	$t_{w(CKO1H)}$	Pulse duration, CLKOUT1 high	(P/2) – 0.5	(P/2)+ 0.5	PH – 0.5	PH + 0.5	ns
3	$t_{w(CKO1L)}$	Pulse duration, CLKOUT1 low	(P/2) – 0.5	(P/2)+ 0.5	PL – 0.5	PL + 0.5	ns
4	$t_{t(CKO1)}$	Transition time, CLKOUT1†		0.6		0.6	ns

† Values specified from characterization data and not tested.
‡ P = 1/CPU clock frequency in nanoseconds (ns).
§ PH is the high period of CLKOUT1 in ns and PL is the low period of CLKOUT1 in ns.

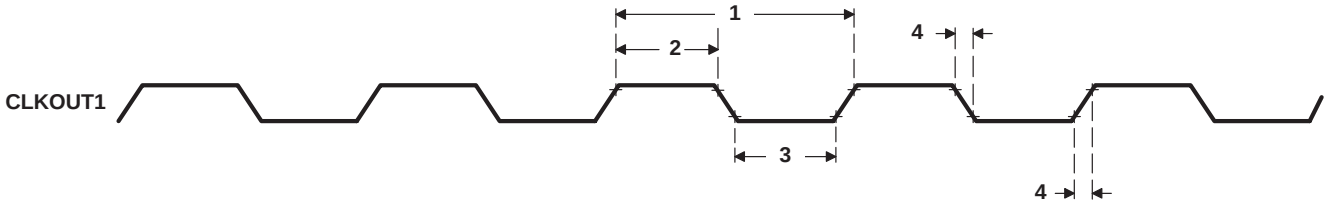


Figure 10. CLKOUT1 Timings

ADVANCE INFORMATION

INPUT AND OUTPUT CLOCKS (CONTINUED)

CLKOUT2 timing parameters† (see Figure 11)

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{c(CKO2)}$	Cycle time, CLKOUT2	2P – 0.7	2P + 0.7	ns
2	$t_{w(CKO2H)}$	Pulse duration, CLKOUT2 high	P – 0.7	P + 0.7	ns
3	$t_{w(CKO2L)}$	Pulse duration, CLKOUT2 low	P – 0.7	P + 0.7	ns
4	$t_{t(CKO2)}$	Transition time, CLKOUT2‡		0.6	ns

† P = 1/CPU clock frequency in ns.
‡ Values specified from characterization data and not tested.

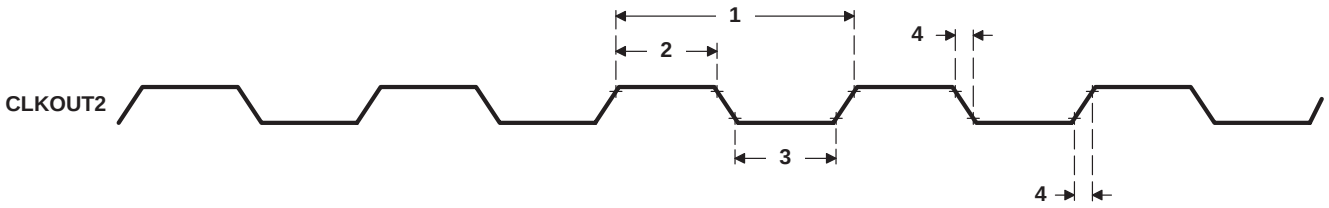


Figure 11. CLKOUT2 Timings

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.

relation of SSCLK, SDCLK, and CLKOUT2 to CLKOUT1 (see Figure 12)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{d(CKO1-SSCLK)}$	Delay time, CLKOUT1 edge to SSCLK edge	–1.2	1.6	ns
2	$t_{d(CKO1-SSCLK1/2)}$	Delay time, CLKOUT1 edge to SSCLK edge (1/2 clock rate)	–1.0	2.4	ns
3	$t_{d(CKO1-CKO2)}$	Delay time, CLKOUT1 edge to CLKOUT2 edge	–1.0	2.4	ns
4	$t_{d(CKO1-SDCLK)}$	Delay time, CLKOUT1 edge to SDCLK edge	–1.0	2.4	ns

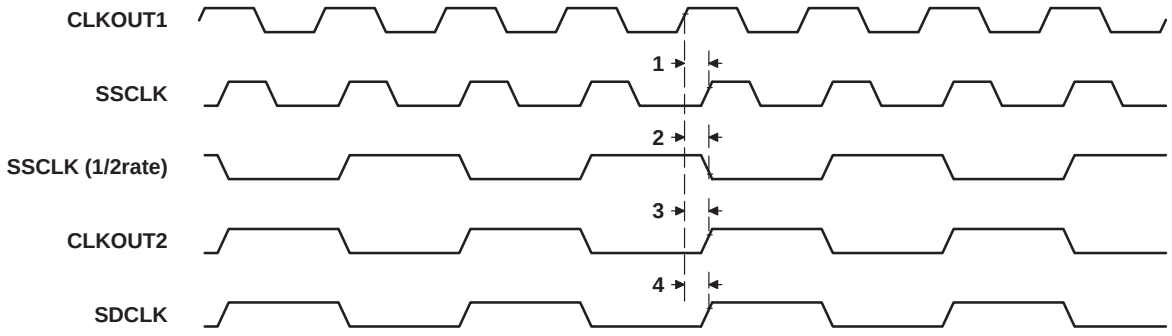


Figure 12. Relation of CLKOUT2, SDCLK and SSCLK to CLKOUT1

ADVANCE INFORMATION

ASYNCHRONOUS MEMORY TIMING

timing requirements for asynchronous memory cycles† (see Figure 13 and Figure 14)

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
6	$t_{su(EDV-CKO1H)}$	Setup time, read EDx valid before CLKOUT1 high	5.0		ns
7	$t_{h(CKO1H-EDV)}$	Hold time, read EDx valid after CLKOUT1 high	0		ns
10	$t_{su(ARDY-CKO1H)}$	Setup time, ARDY valid before CLKOUT1 high	5.0		ns
11	$t_{h(CKO1H-ARDY)}$	Hold time, ARDY valid after CLKOUT1 high	0		ns

† To ensure data setup time, simply program the strobe width wide enough. ARDY is internally synchronized. If it 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‡ (see Figure 13 and Figure 14)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{d(CKO1H-CEV)}$	Delay time, CLKOUT1 high to $\overline{CEx}$ valid	–1.0	5.0	ns
2	$t_{d(CKO1H-BEV)}$	Delay time, CLKOUT1 high to $\overline{BEx}$ valid	–1.0	5.0	ns
3	$t_{d(CKO1H-BEIV)}$	Delay time, CLKOUT1 high to $\overline{BEx}$ invalid§	–1.0	5.0	ns
4	$t_{d(CKO1H-EAV)}$	Delay time, CLKOUT1 high to EAx valid	–1.0	5.0	ns
5	$t_{d(CKO1H-EAIV)}$	Delay time, CLKOUT1 high to EAx invalid§	–1.0	5.0	ns
8	$t_{d(CKO1H-AOEV)}$	Delay time, CLKOUT1 high to $\overline{AOE}$ valid	–1.0	5.0	ns
9	$t_{d(CKO1H-AREV)}$	Delay time, CLKOUT1 high to $\overline{ARE}$ valid	–1.0	5.0	ns
12	$t_{d(CKO1H-EDLZ)}$	Delay time, CLKOUT1 high to EDx low impedance§	–1.0		ns
13	$t_{d(CKO1H-EDV)}$	Delay time, CLKOUT1 high to EDx valid		5.0	ns
14	$t_{d(CKO1H-EDIV)}$	Delay time, CLKOUT1 high to EDx invalid§	–1.0		ns
15	$t_{d(CKO1H-EDHZ)}$	Delay time, CLKOUT1 high to EDx high impedance§		6.5	ns
16	$t_{d(CKO1H-AWEV)}$	Delay time, CLKOUT1 high to $\overline{AWE}$ valid	–1.0	5.0	ns

‡ The minimum delay is also the minimum output hold after CLKOUT1 high.
§ Values specified by design but not tested

ADVANCE INFORMATION

ASYNCHRONOUS MEMORY TIMING (CONTINUED)

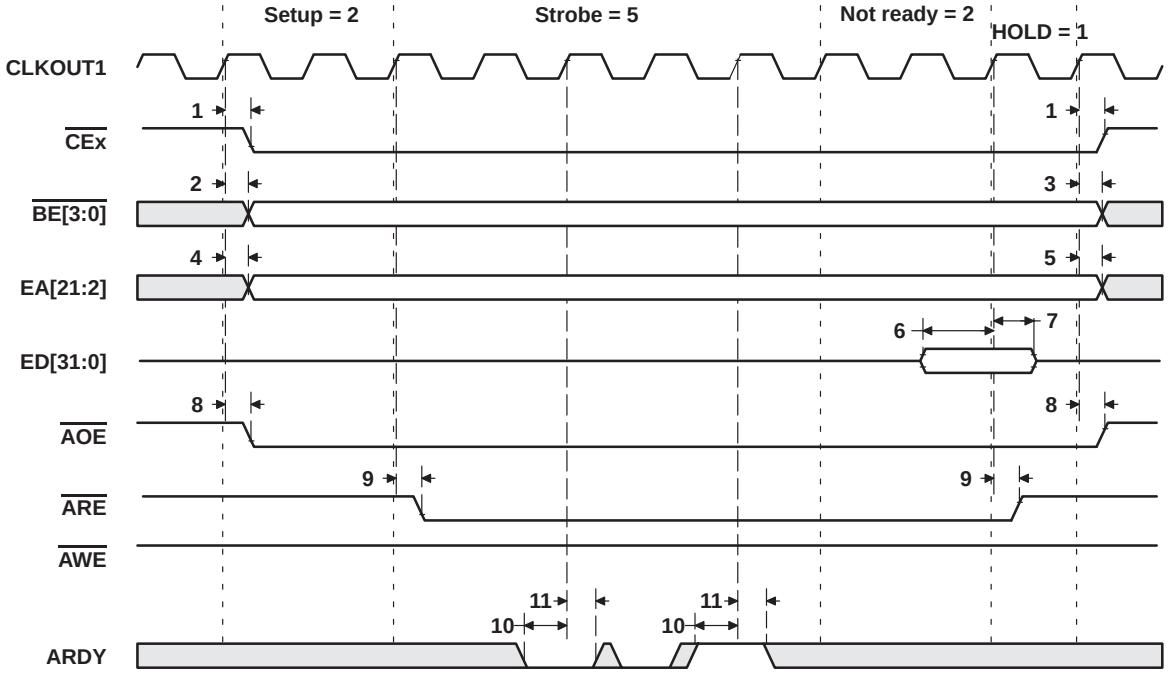


Figure 13. Asynchronous Memory Read Timing

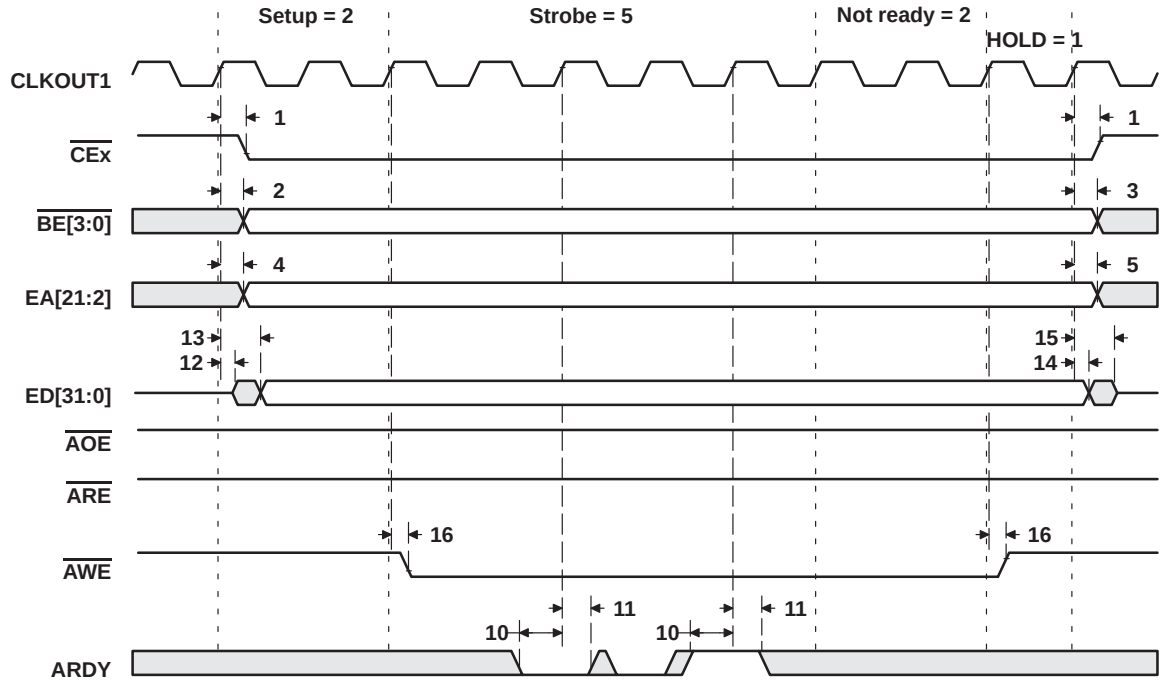


Figure 14. Asynchronous Memory Write Timing

SYNCHRONOUS-BURST MEMORY TIMING

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

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
7	$t_{su(EDV-SSCLKH)}$	Setup time, read EDx valid before SSCLK high	1.5		ns
8	$t_{h(SSCLKH-EDV)}$	Hold time, read EDx valid after SSCLK high	1.2		ns

switching characteristics for synchronous burst SRAM cycles† (full-rate SSCLK) (see Figure 15 and Figure 16)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{su(CEV-SSCLKH)}$	Setup time, $\overline{CEx}$ valid before SSCLK high	P – 4		ns
2	$t_{oh(SSCLKH-CEV)}$	Output hold time, $\overline{CEx}$ valid after SSCLK high	0		ns
3	$t_{su(BEV-SSCLKH)}$	Setup time, $\overline{BEx}$ valid before SSCLK high	P – 4		ns
4	$t_{oh(SSCLKH-BEIV)}$	Output hold time, $\overline{BEx}$ invalid after SSCLK high‡	1		ns
5	$t_{su(EAV-SSCLKH)}$	Setup time, EAx valid before SSCLK high	P – 4		ns
6	$t_{oh(SSCLKH-EAIV)}$	Output hold time, EAx invalid after SSCLK high‡	1		ns
9	$t_{su(ADSV-SSCLKH)}$	Setup time, $\overline{SSADS}$ valid before SSCLK high	P – 4		ns
10	$t_{oh(SSCLKH-ADSV)}$	Output hold time, $\overline{SSADS}$ valid after SSCLK high	1		ns
11	$t_{su(OEV-SSCLKH)}$	Setup time, $\overline{SSOE}$ valid before SSCLK high	P – 4		ns
12	$t_{oh(SSCLKH-OEV)}$	Output hold time, $\overline{SSOE}$ valid after SSCLK high	0		ns
13	$t_{su(EDLZ-SSCLKH)}$	Setup time, EDx low impedance before SSCLK high‡§	P – 4.6		ns
14	$t_{su(EDV-SSCLKH)}$	Setup time, EDx valid before SSCLK high	P – 4		ns
15	$t_{oh(SSCLKH-EDIV)}$	Output hold time, EDx invalid after SSCLK high‡	1		ns
16	$t_{oh(SSCLKH-EDHZ)}$	Output hold time, EDx high impedance after SSCLK high‡		5.8	ns
17	$t_{su(WEV-SSCLKH)}$	Setup time, $\overline{SSWE}$ valid before SSCLK high	P – 4		ns
18	$t_{oh(SSCLKH-WEV)}$	Output hold time, $\overline{SSWE}$ valid after SSCLK high	1		ns

† The effects of internal clock jitter are included at test. There is no need to adjust timing numbers for internal clock jitter or SSCLK duty cycle. P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ Values specified by design but not tested
§ This parameter applies to the first write in a series of one or more consecutive adjacent writes. Note that the write data is generated two CLKOUT1 cycles early to accommodate the ED enable time.

ADVANCE INFORMATION

SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)

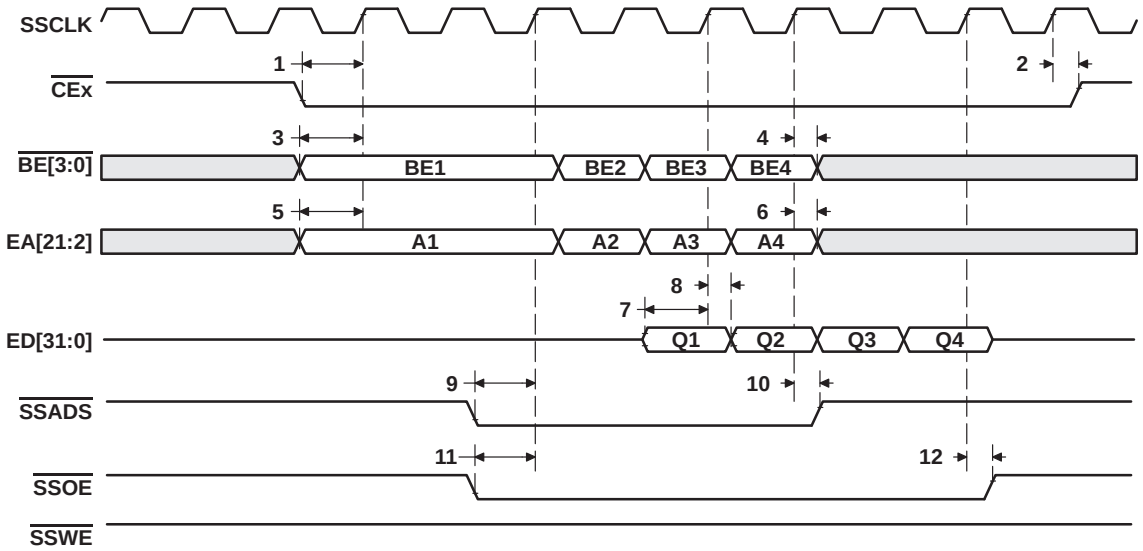


Figure 15. SBSRAM Read Timing (Full-Rate SSCLK)

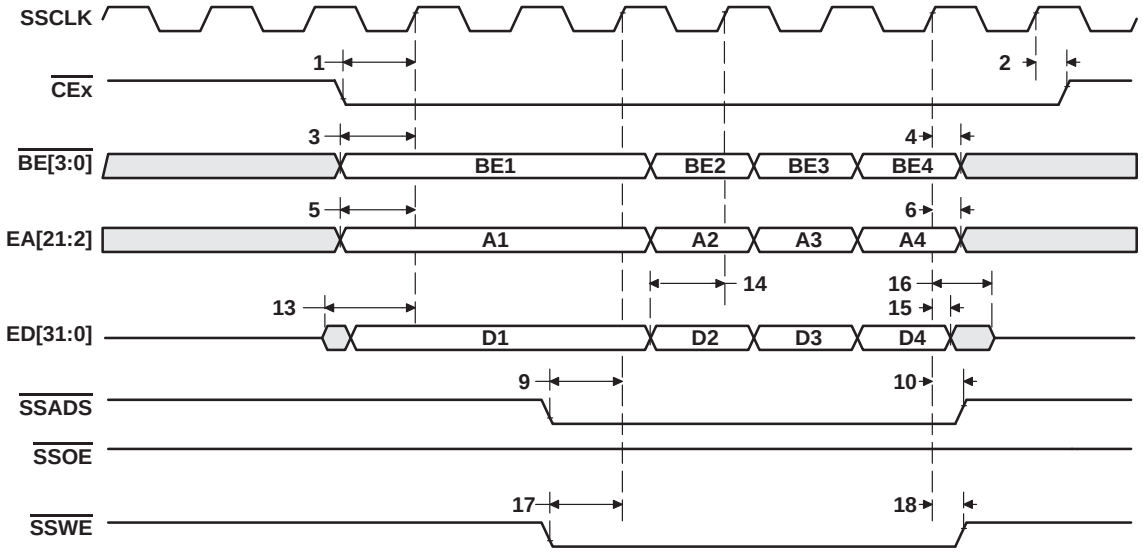


Figure 16. SBSRAM Write Timing (Full-Rate SSCLK)

ADVANCE INFORMATION

SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)

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

NO.			’C6201-167		’C6201-200		UNIT
			MIN	MAX	MIN	MAX	
7	$t_{su(EDV-SSCLKH)}$	Setup time, read EDx valid before SSCLK high	3.6		3.6		ns
8	$t_{h(SSCLKH-EDV)}$	Hold time, read EDx valid after SSCLK high	1.2		1.2		ns

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

NO.	PARAMETER		’C6201-167		’C6201-200		UNIT
			MIN	MAX	MIN	MAX	
1	$t_{su(CEV-SSCLKH)}$	Setup time, $\overline{CEx}$ valid before SSCLK high	P – 3.4		P – 3.4		ns
2	$t_{oh(SSCLKH-CEV)}$	Output hold time, $\overline{CEx}$ valid after SSCLK high	P – 5		P – 4		ns
3	$t_{su(BEV-SSCLKH)}$	Setup time, $\overline{BEx}$ valid before SSCLK high	P – 3.3		P – 2.3		ns
4	$t_{oh(SSCLKH-BEIV)}$	Output hold time, $\overline{BEx}$ invalid after SSCLK high‡	P – 5		P – 4		ns
5	$t_{su(EAV-SSCLKH)}$	Setup time, EAx valid before SSCLK high	P – 3.3		P – 2.3		ns
6	$t_{oh(SSCLKH-EAIV)}$	Output hold time, EAx invalid after SSCLK high‡	P – 5		P – 4		ns
9	$t_{su(ADSV-SSCLKH)}$	Setup time, $\overline{SSADS}$ valid before SSCLK high	P – 3.3		P – 2.3		ns
10	$t_{oh(SSCLKH-ADSV)}$	Output hold time, $\overline{SSADS}$ valid after SSCLK high	P – 5		P – 4		ns
11	$t_{su(OEV-SSCLKH)}$	Setup time, $\overline{SSOE}$ valid before SSCLK high	P – 3.3		P – 3.1		ns
12	$t_{oh(SSCLKH-OEV)}$	Output hold time, $\overline{SSOE}$ valid after SSCLK high	P – 5		P – 4		ns
13	$t_{su(EDLZ-SSCLKH)}$	Setup time, EDx low impedance before SSCLK high‡§	P – 4.4		P – 4.4		ns
14	$t_{su(EDV-SSCLKH)}$	Setup time, EDx valid before SSCLK high	P – 3.3		P – 2.3		ns
15	$t_{oh(SSCLKH-EDIV)}$	Output hold time, EDx invalid after SSCLK high‡	P – 5		P – 4		ns
16	$t_{oh(SSCLKH-EDHZ)}$	Output hold time, EDx high impedance after SSCLK high‡		5.6		5.6	ns
17	$t_{su(WEV-SSCLKH)}$	Setup time, $\overline{SSWE}$ valid before SSCLK high	P – 3.3		P – 2.3		ns
18	$t_{oh(SSCLKH-WEV)}$	Output hold time, $\overline{SSWE}$ valid after SSCLK high	P – 5		P – 4		ns

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

‡ Values specified by design but not tested

§ This parameter applies to the first write in a series of one or more consecutive adjacent writes. Note that the write data is generated two CLKOUT1 cycles early to accommodate the ED enable time.

ADVANCE INFORMATION

SYNCHRONOUS-BURST MEMORY TIMING (CONTINUED)

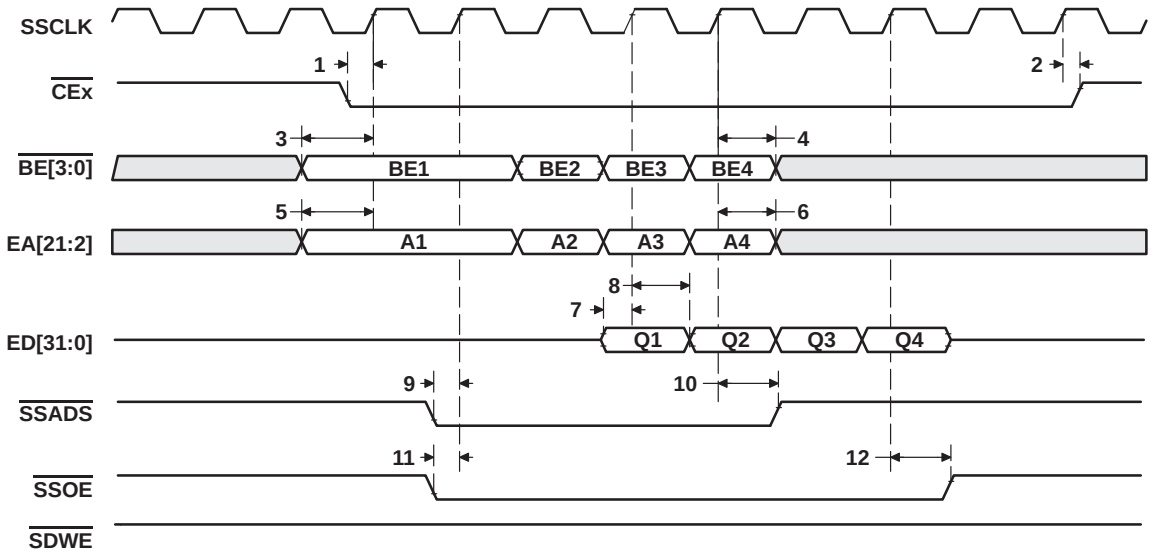


Figure 17. SBSRAM Read Timing (1/2 Rate SSCLK)

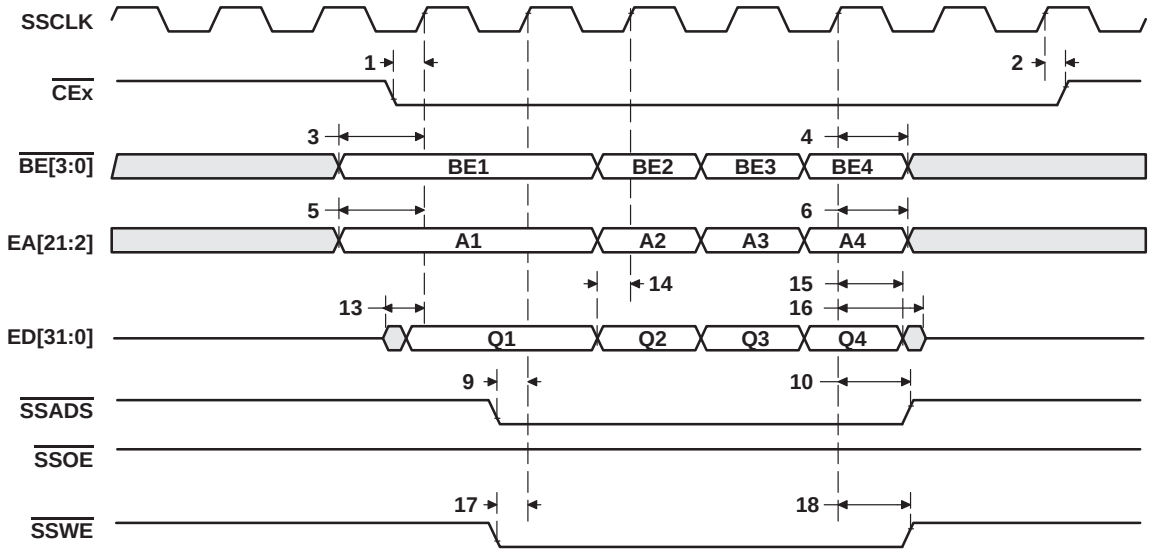


Figure 18. SBSRAM Write Timing (1/2 Rate SSCLK)

ADVANCE INFORMATION

SYNCHRONOUS DRAM TIMING

timing requirements for synchronous DRAM cycles (see Figure 19)

NO.			’C6201-167		’C6201-200		UNIT
			MIN	MAX	MIN	MAX	
7	$t_{su(EDV-SDCLKH)}$	Setup time, read EDx valid before SDCLK high	3.5		1.5		ns
8	$t_{h(SDCLKH-EDV)}$	Hold time, read EDx valid after SDCLK high	1.2		1.2		ns

switching characteristics for synchronous DRAM cycles†‡ (see Figure 19–Figure 24)

NO.	PARAMETER		’C6201-167		’C6201-200		UNIT
			MIN	MAX	MIN	MAX	
1	$t_{su(CEV-SDCLKH)}$	Setup time, $\overline{CEx}$ valid before SDCLK high	P – 3.5		P – 2.5		ns
2	$t_{oh(SDCLKH-CEV)}$	Output hold time, $\overline{CEx}$ valid after SDCLK high	P – 4.5		P – 3.5		ns
3	$t_{su(BEV-SDCLKH)}$	Setup time, $\overline{BEx}$ valid before SDCLK high	P – 3.5		P – 2.5		ns
4	$t_{oh(SDCLKH-BEIV)}$	Output hold time, $\overline{BEx}$ invalid after SDCLK high§	P – 4.5		P – 3.5		ns
5	$t_{su(EAV-SDCLKH)}$	Setup time, EAx valid before SDCLK high	P – 3.5		P – 2.5		ns
6	$t_{oh(SDCLKH-EAIV)}$	Output hold time, EAx invalid after SDCLK high§	P – 4.5		P – 3.5		ns
9	$t_{su(SDCAS-SDCLKH)}$	Setup time, $\overline{SDCAS}$ valid before SDCLK high	P – 3.5		P – 2.5		ns
10	$t_{oh(SDCLKH-SDCAS)}$	Output hold time, $\overline{SDCAS}$ valid after SDCLK high	P – 4.5		P – 3.5		ns
11	$t_{su(EDLZ-SDCLKH)}$	Setup time, EDx low impedance before SDCLK high§	P – 4.4		P – 4.4		ns
12	$t_{su(EDV-SDCLKH)}$	Setup time, EDx valid before SDCLK high	P – 3.5		P – 2.5		ns
13	$t_{oh(SDCLKH-EDIV)}$	Output hold time, EDx invalid after SDCLK high§	P – 4.5		P – 3.5		ns
14	$t_{oh(SDCLKH-EDHZ)}$	Output hold time, EDx high impedance after SDCLK high§		5.6		5.6	ns
15	$t_{su(SDWE-SDCLKH)}$	Setup time, $\overline{SDWE}$ valid before SDCLK high	P – 3.5		P – 2.5		ns
16	$t_{oh(SDCLKH-SDWE)}$	Output hold time, $\overline{SDWE}$ valid after SDCLK high	P – 4.5		P – 3.5		ns
17	$t_{su(SDA10V-SDCLKH)}$	Setup time, SDA10 valid before SDCLK high	P – 3.5		P – 2.5		ns
18	$t_{oh(SDCLKH-SDA10IV)}$	Output hold time, SDA10 invalid after SDCLK high§	P – 4.5		P – 3.5		ns
19	$t_{su(SDRAS-SDCLKH)}$	Setup time, $\overline{SDRAS}$ valid before SDCLK high	P – 3.5		P – 2.5		ns
20	$t_{oh(SDCLKH-SDRAS)}$	Output hold time, $\overline{SDRAS}$ valid after SDCLK high	P – 4.5		P – 3.5		ns

† The effects of internal clock jitter are included at test. There is no need to adjust timing numbers for internal clock jitter or SSCLK duty cycle. P = 1/CPU clock frequency in ns. For example, when running parts at 200 MHz, use P = 5 ns.
‡ The minimum delay also represents the minimum output hold time after SDCLK high.
§ Values specified by design but not tested

ADVANCE INFORMATION

SYNCHRONOUS DRAM TIMING (CONTINUED)

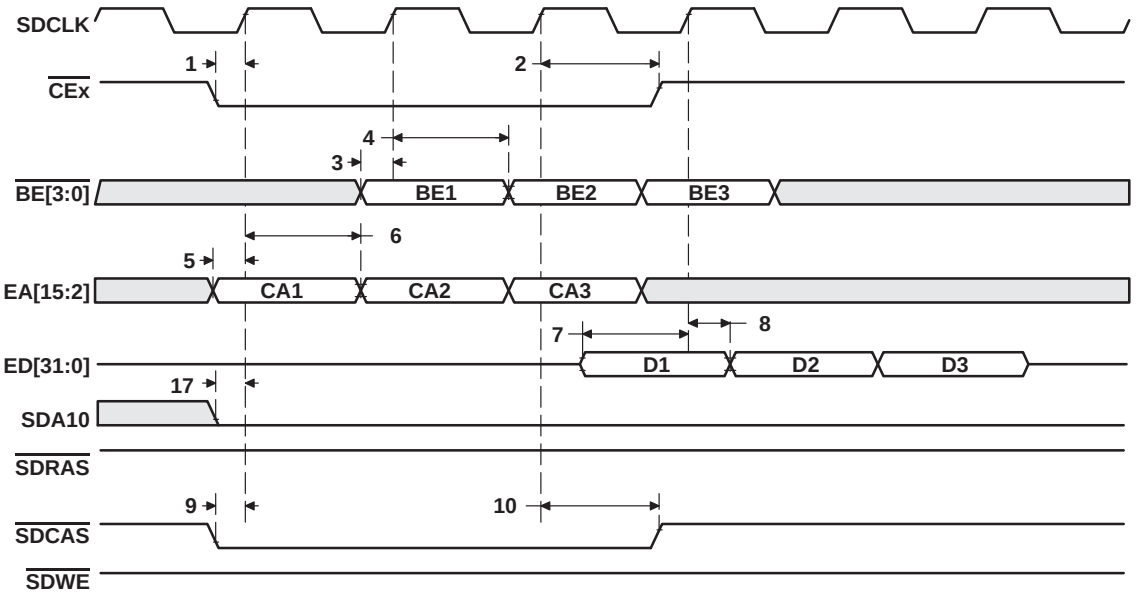


Figure 19. Three SDRAM Read Commands

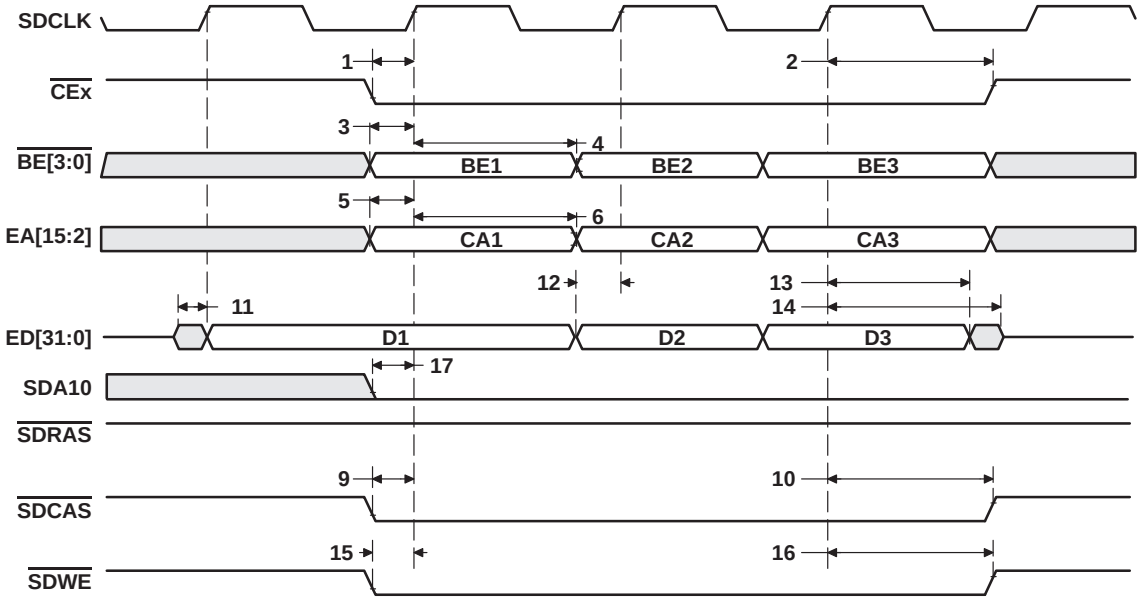


Figure 20. Three SDRAM WRT Commands

ADVANCE INFORMATION

SYNCHRONOUS DRAM TIMING (CONTINUED)

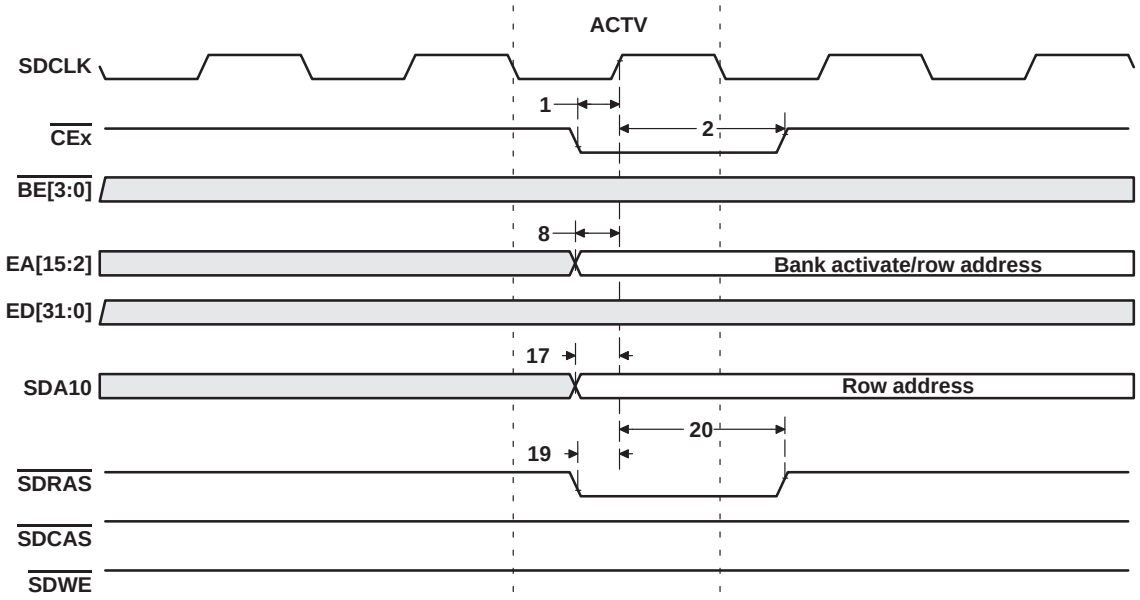


Figure 21. SDRAM ACTV Command

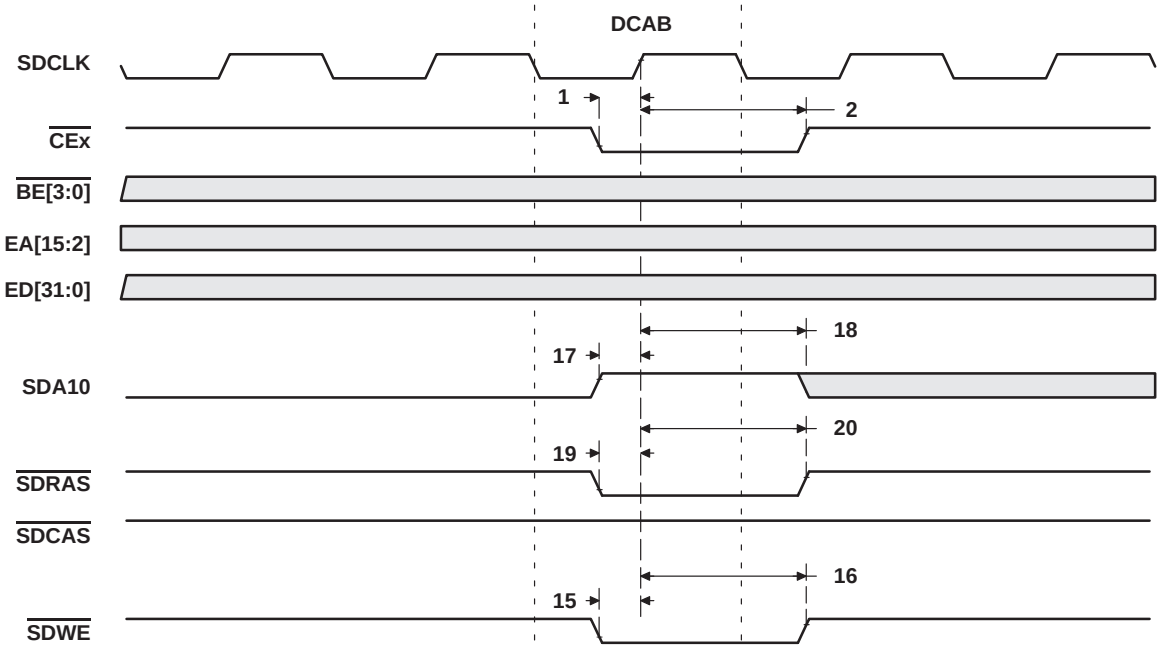


Figure 22. SDRAM DCAB Command

SYNCHRONOUS DRAM TIMING (CONTINUED)

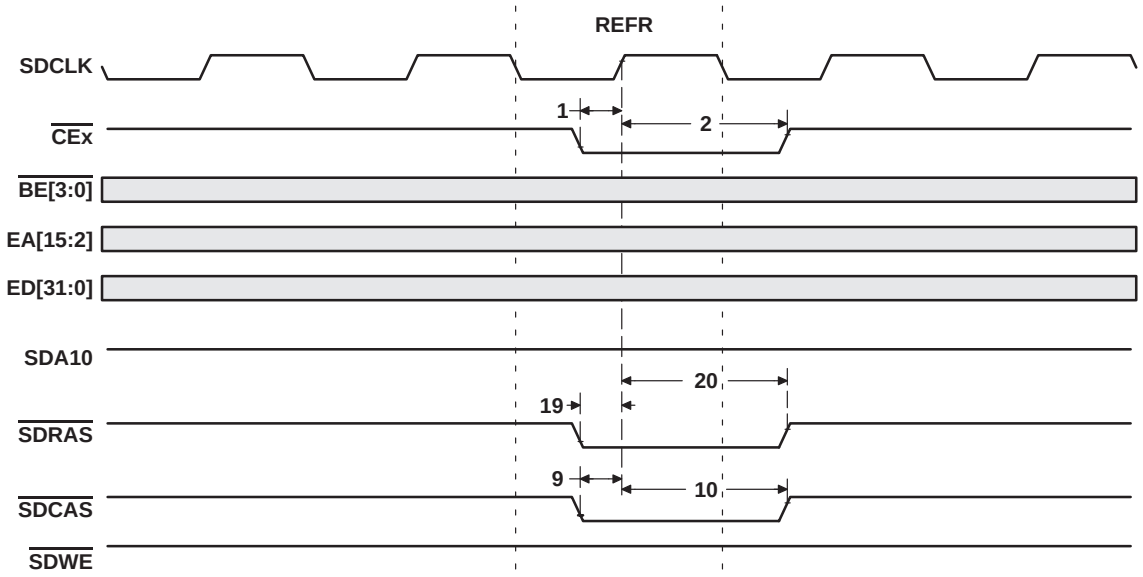


Figure 23. SDRAM REFR Command

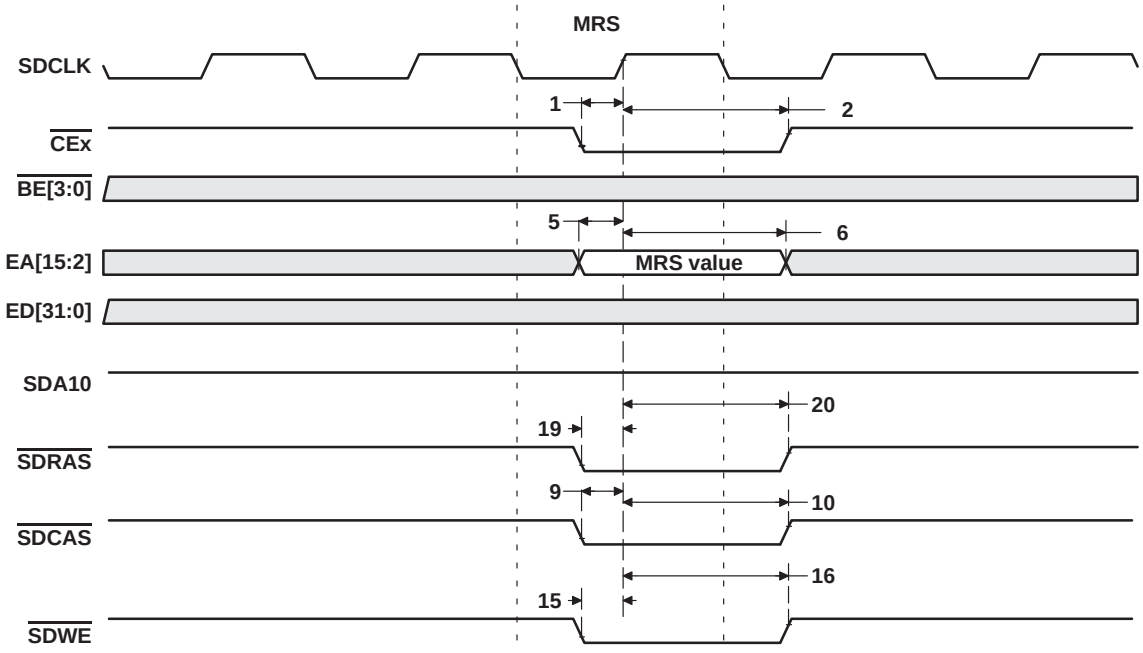


Figure 24. SDRAM MRS Command

ADVANCE INFORMATION

HOLD/HOLDA TIMING

timing requirements for the hold/hold acknowledge cycles† (see Figure 25)

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{su(HOLDH-CKO1H)}$	Setup time, HOLD high before CLKOUT1 high	5		ns
2	$t_{h(CKO1H-HOLDL)}$	Hold time, HOLD low after CLKOUT1 high	0		ns

† Note that 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, HOLD can be an asynchronous input.

switching characteristics for the hold/hold acknowledge cycles (see Figure 25)‡

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
3	$t_{R(HOLDL-EMHZ)}$	Response time, HOLD low to EMIF high impedance§	4P	¶	ns
4	$t_{d(EMHZ-HOLDAL)}$	Delay time, EMIF high impedance to HOLDA low§		2P	ns
5	$t_{R(HOLDH-HOLDAH)}$	Response time, HOLD high to HOLDA high		6P	ns
6	$t_{d(CKO1H-HOLDAL)}$	Delay time, CLKOUT1 high to HOLDA valid	–1	5	ns
7	$t_{d(CKO1H-BHZ)}$	Delay time, CLKOUT1 high to BUS high impedance§#	–1	5	ns
8	$t_{d(CKO1H-BLZ)}$	Delay time, CLKOUT1 high to BUS low impedance§#	–1	5	ns

‡ 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.

§ Values specified by design but not tested

¶ All pending EMIF transactions are allowed to complete before 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 occuring, then the minimum delay time can be achieved. Also, bus hold can be indefinitely delayed by setting the NOHOLD = 1.

BUS consists of CE[3:0], BE[3:0], ED[31:0], EA[21:2], ARE, AOE, AWE, SSADS, SSOE, SSWE, SDA10, SDRAS, SDCAS, and SDWE.

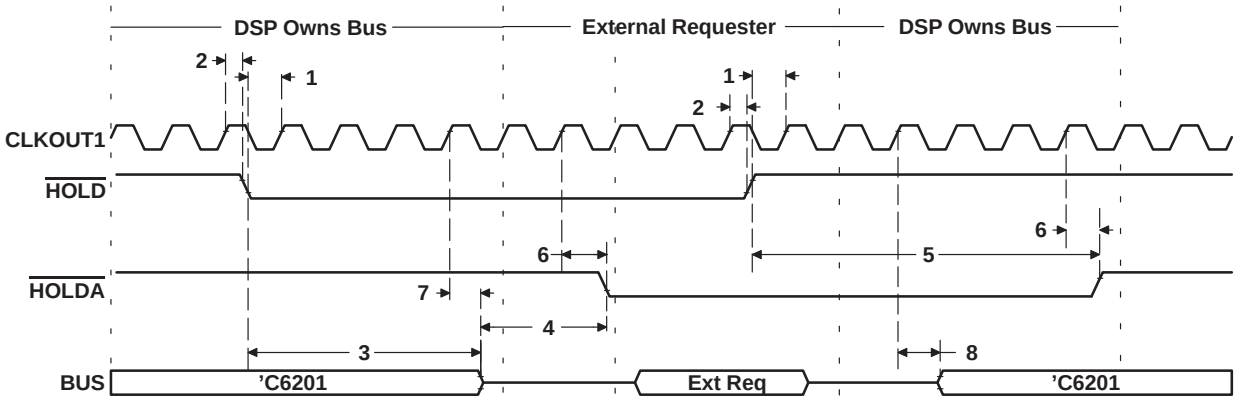


Figure 25. HOLD/HOLDA Timing

ADVANCE INFORMATION

RESET TIMING

timing requirements for reset† (see Figure 26)

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{w(RESET)}$	Width of the $\overline{\text{RESET}}$ pulse (PLL stable)	10‡		CLKOUT1 cycles
		Width of the $\overline{\text{RESET}}$ pulse (PLL needs to sync up)§	250‡		µs

† As long as the $\overline{\text{RESET}}$ pulse satisfies $t_{w(RESET)}$ or $t_{r(RESET)}$ (as applicable), it need not satisfy the setup and hold times. Thus, $\overline{\text{RESET}}$ can be an asynchronous input.
‡ Values specified from characterization data and not tested
§ The $\overline{\text{RESET}}$ 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, $\overline{\text{RESET}}$ must be asserted to ensure proper device operation. See the PLL section for PLL lock times.

switching characteristics during reset¶ (see Figure 26)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
2	$t_{R(RESET)}$	Response time to change of value in $\overline{\text{RESET}}$ signal	2		CLKOUT1 cycles
3	$t_{d(CKO1H-CKO2IV)}$	Delay time, CLKOUT1 high to CLKOUT2 invalid#	–1	10	ns
4	$t_{d(CKO1H-CKO2V)}$	Delay time, CLKOUT1 high to CLKOUT2 valid	–1	10	ns
5	$t_{d(CKO1H-SDCLKIV)}$	Delay time, CLKOUT1 high to SDCLK invalid#	–1	10	ns
6	$t_{d(CKO1H-SDCLKV)}$	Delay time, CLKOUT1 high to SDCLK valid	–1	10	ns
7	$t_{d(CKO1H-SSCKIV)}$	Delay time, CLKOUT1 high to SSCLK invalid#	–1	10	ns
8	$t_{d(CKO1H-SSCKV)}$	Delay time, CLKOUT1 high to SSCLK valid	–1	10	ns
9	$t_{d(CKO1H-LOWIV)}$	Delay time, CLKOUT1 high to low group invalid#	–1	10	ns
10	$t_{d(CKO1H-LOWV)}$	Delay time, CLKOUT1 high to low group valid	–1	10	ns
11	$t_{d(CKO1H-HIGHIV)}$	Delay time, CLKOUT1 high to high group invalid#	–1	10	ns
12	$t_{d(CKO1H-HIGHV)}$	Delay time, CLKOUT1 high to high group valid	–1	10	ns
13	$t_{d(CKO1H-ZHZ)}$	Delay time, CLKOUT1 high to Z group high impedance#	–1	10	ns
14	$t_{d(CKO1H-ZV)}$	Delay time, CLKOUT1 high to Z group valid	–1	10	ns

¶ Low group consists of: $\overline{\text{IACK}}$, INUM[3:0], DMAC[3:0], PD, DMAC[0:3], TOUT0, and TOUT1.
High group consists of: $\overline{\text{HRDY}}$ and $\overline{\text{HINT}}$.
Z group consists of: EA[21:2], ED[31:0], $\overline{\text{CE}}$[3:0], $\overline{\text{BE}}$[3:0], $\overline{\text{ARE}}$, $\overline{\text{AWE}}$, $\overline{\text{AOE}}$, $\overline{\text{SSADS}}$, $\overline{\text{SSOE}}$, $\overline{\text{SSWE}}$, SSCLK, SDA10, $\overline{\text{SDRAS}}$, $\overline{\text{SDCAS}}$, $\overline{\text{SDWE}}$, SDCLK, HD[15:0], CLKX0, CLKX1, FSX0, FSX1, DX0, DX1, CLKR0, CLKR1, FSR0, and FSR1.
Values specified by design but not tested

ADVANCE INFORMATION

RESET TIMING (CONTINUED)

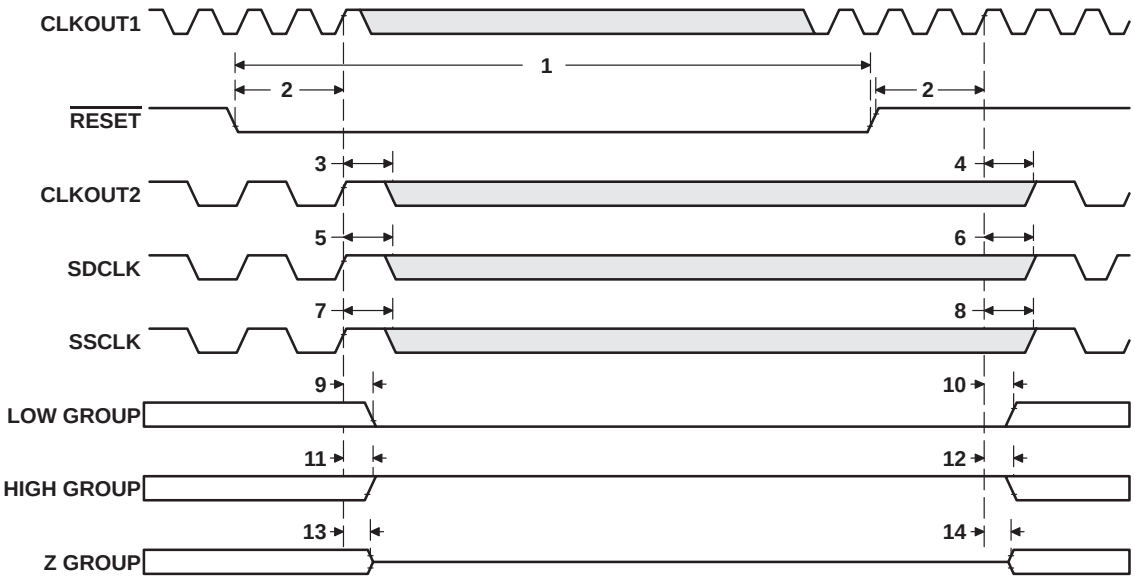


Figure 26. Reset Timing

EXTERNAL INTERRUPT/RESET TIMING

timing requirements for interrupt response cycles† (see Figure 27)

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
3	$t_{w(ILOW)}$	Width of the interrupt pulse low	2		CLKOUT1 cycles
4	$t_{w(IHIGH)}$	Width of the interrupt pulse high	2		CLKOUT1 cycles

† Note interrupt signals are internally synchronized and are potentially recognized one cycle later if setup and hold times are violated. Thus, they can be connected to asynchronous inputs.

switching characteristics during interrupt response cycles (see Figure 27)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{R(IACK)}$	Response time, IACK high after EXT_INTx high	9‡		CLKOUT1 cycles
2	$t_{R(ISFP)}$	Response time, interrupt service fetch packet execution after EXT_INTx high	11‡		CLKOUT1 cycles
5	$t_{d(CKO2L-IACKV)}$	Delay time, CLKOUT2 low to IACK valid	0	10	ns
6	$t_{d(CKO2L-INUMV)}$	Delay time, CLKOUT2 low to INUM valid	0	10	ns
7	$t_{d(CKO2L-INUMIV)}$	Delay time, CLKOUT2 low to INUM invalid§	0	10	ns

‡ Add two CLKOUT1 cycles to this parameter if the interrupt is recognized during the high half of CLKOUT2
§ Values specified by design but not tested

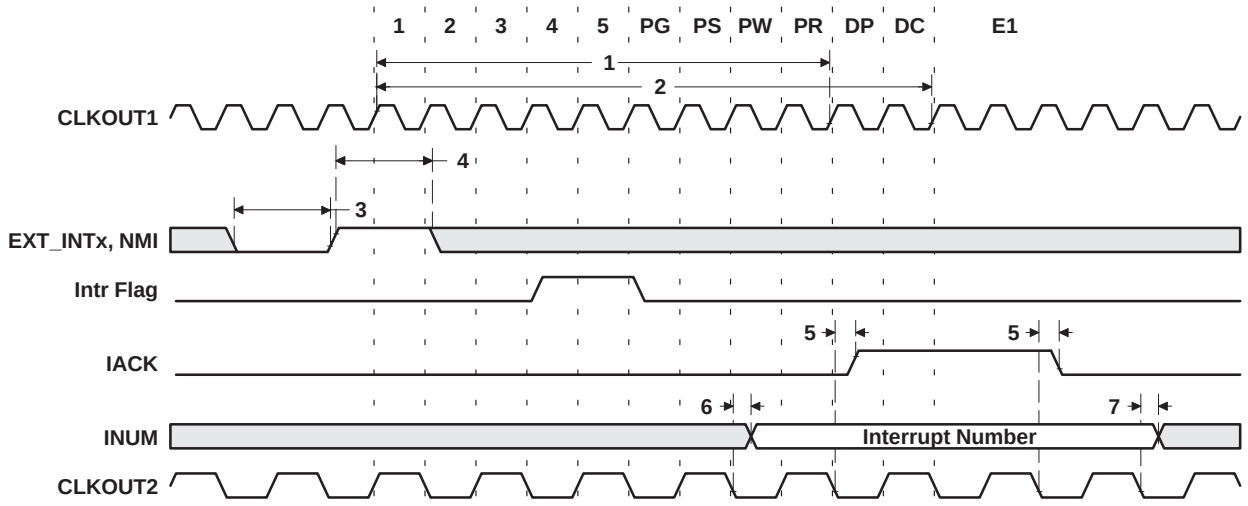


Figure 27. Interrupt Timing

ADVANCE INFORMATION

HOST PORT TIMING

timing requirements for host port interface cycles†‡ (see Figure 28, Figure 29, Figure 30, and Figure 31)

NO.			'C6201-167 'C6201-200		UNIT
			MIN	MAX	
1	$t_{su(SEL-HSTBL)}$	Setup time, select signals§ valid before $\overline{\text{HSTROBE}}$ low	1		ns
2	$t_{h(HSTBL-SEL)}$	Hold time, select signals§ valid after $\overline{\text{HSTROBE}}$ low	2		ns
3	$t_{w(HSTBL)}$	Pulse duration, $\overline{\text{HSTROBE}}$ low	2P		ns
4	$t_{w(HSTBH)}$	Pulse duration, $\overline{\text{HSTROBE}}$ high between consecutive accesses	2P		ns
10	$t_{su(SEL-HASL)}$	Setup time, select signals§ valid before $\overline{\text{HAS}}$ low	1		ns
11	$t_{h(HASL-SEL)}$	Hold time, select signals§ valid after $\overline{\text{HAS}}$ low	2		ns
12	$t_{su(HDV-HSTBH)}$	Setup time, host data valid before $\overline{\text{HSTROBE}}$ high	1		ns
13	$t_{h(HSTBH-HDV)}$	Hold time, host data valid after $\overline{\text{HSTROBE}}$ high	1		ns
14	$t_{h(HRDYL-HSTBL)}$	Hold time, $\overline{\text{HSTROBE}}$ low after $\overline{\text{HRDY}}$ low. This indicates that $\overline{\text{HSTROBE}}$ should not be inactivated until $\overline{\text{HRDY}}$ is active (low). Otherwise, HPI writes will not complete properly.	1		ns

† $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: [NOT($\overline{\text{HDS1}}$ XOR $\overline{\text{HDS2}}$)] OR $\overline{\text{HCS}}$.
‡ 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.
§ Select signals include: HCNTRL[1:0], HR/$\overline{\text{W}}$, and HHWIL.

switching characteristics during host port interface cycles†‡ (see Figure 28, Figure 29, Figure 30, and Figure 31)

NO.	PARAMETER		'C6201-167 'C6201-200		UNIT
			MIN	MAX	
5	$t_{d(HCS-HRDY)}$	Delay time, $\overline{\text{HCS}}$ to $\overline{\text{HRDY}}$¶	1	7	ns
6	$t_{d(HSTBL-HRDYH)}$	Delay time, $\overline{\text{HSTROBE}}$ low to $\overline{\text{HRDY}}$ high#	3	12	ns
7	$t_{oh(HSTBL-HDLZ)}$	Output hold time, HD low impedance after $\overline{\text{HSTROBE}}$ low for an HPI read‖	4		ns
8	$t_{d(HDV-HRDYL)}$	Delay time, HD valid to $\overline{\text{HRDY}}$ low	P – 2	P	ns
9	$t_{oh(HSTBH-HDV)}$	Output hold time, HD valid after $\overline{\text{HSTROBE}}$ high	3	12	ns
15	$t_{d(HSTBH-HDHZ)}$	Delay time, $\overline{\text{HSTROBE}}$ high to HD high impedance‖	3	12	ns
16	$t_{d(HSTBL-HDV)}$	Delay time, $\overline{\text{HSTROBE}}$ low to HD valid	3	12	ns
17	$t_{d(HSTBH-HRDYH)}$	Delay time, $\overline{\text{HSTROBE}}$ high to $\overline{\text{HRDY}}$ high☆	3	12	ns

† $\overline{\text{HSTROBE}}$ refers to the following logical operation on $\overline{\text{HCS}}$, $\overline{\text{HDS1}}$, and $\overline{\text{HDS2}}$: [NOT($\overline{\text{HDS1}}$ XOR $\overline{\text{HDS2}}$)] OR $\overline{\text{HCS}}$.
‡ 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.
¶ $\overline{\text{HCS}}$ enables $\overline{\text{HRDY}}$, and $\overline{\text{HRDY}}$ is always low when $\overline{\text{HCS}}$ is high. The case where $\overline{\text{HRDY}}$ goes high when $\overline{\text{HCS}}$ falls indicates that HPI is busy completing a previous HPID write or READ with autoincrement.
# This parameter is used during an HPID read. At the beginning of the first halfword transfer on the falling edge of $\overline{\text{HSTROBE}}$, the HPI sends the request to the DMA auxiliary channel, and $\overline{\text{HRDY}}$ remains high until the DMA auxiliary channel loads the requested data into HPID.
‖ Values specified by design but not tested
☆ This parameter is used after the second halfword of an HPID write or autoincrement read. $\overline{\text{HRDY}}$ remains low if the access is not an HPID write or autoincrement read. Reading or writing to HPIC or HPIA does not affect the $\overline{\text{HRDY}}$ signal.

ADVANCE INFORMATION

HOST PORT TIMING (CONTINUED)

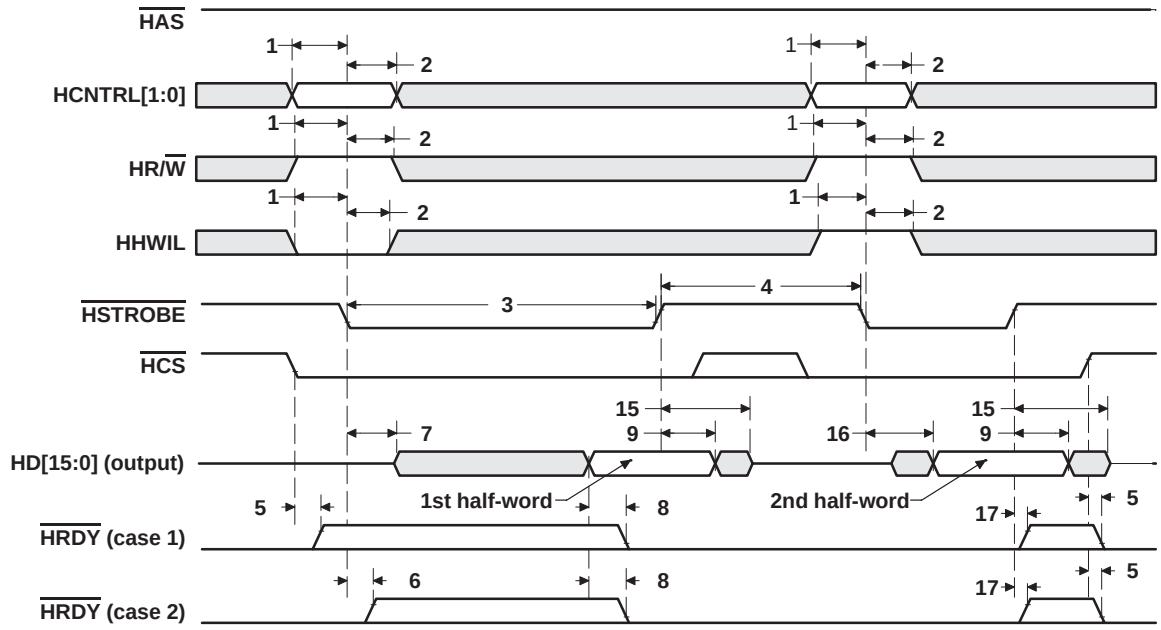


Figure 28. HPI Read Timing (HAS Not Used, Tied High)

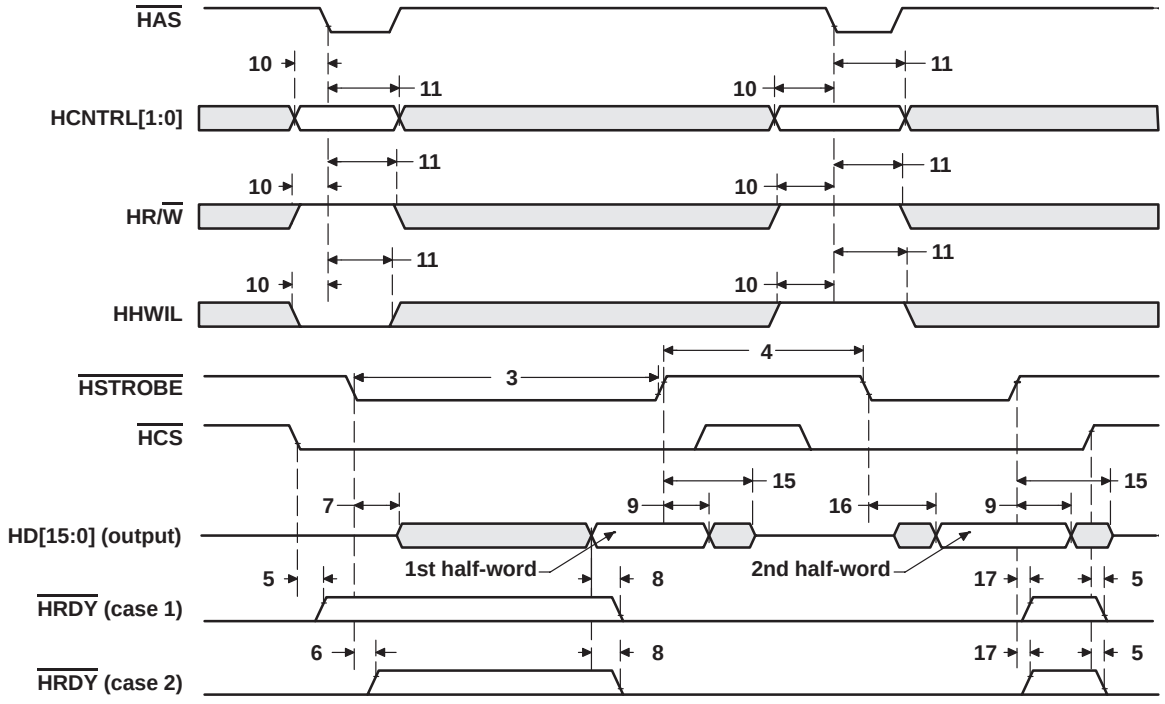


Figure 29. HPI Read Timing (HAS Used)

HOST PORT TIMING (CONTINUED)

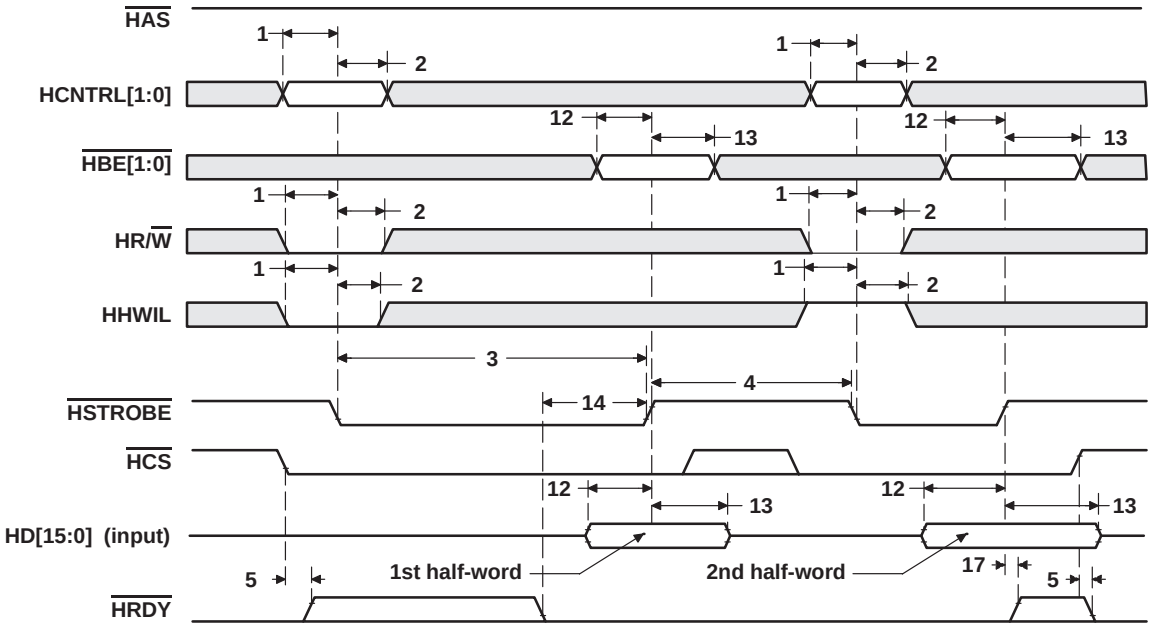


Figure 30. HPI Write Timing ($\overline{\text{HAS}}$ Not Used, Tied High)

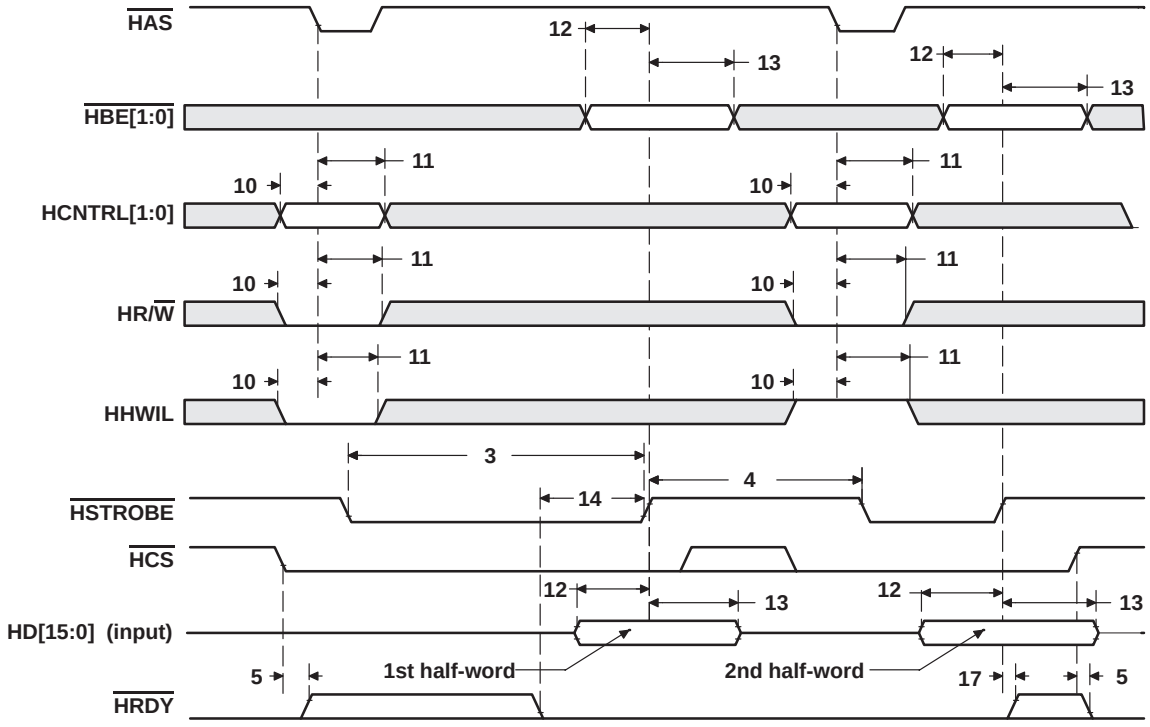


Figure 31. HPI Write Timing ($\overline{\text{HAS}}$ Used)

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING

timing requirements for McBSP† (see Figure 32)

NO.				’C6201-167 ’C6201-200		UNIT
				MIN	MAX	
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X ext	2		CLKOUT1 cycles
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X ext	4		ns
5	$t_{su(FRH-CKRL)}$	Setup time, external FSR high before CLKR low	CLKR int	13		ns
			CLKR ext	4		
6	$t_{h(CKRL-FRH)}$	Hold time, external FSR high after CLKR low	CLKR int	7		ns
			CLKR ext	3		
7	$t_{su(DRV-CKRL)}$	Setup time, DR valid before CLKR low	CLKR int	10		ns
			CLKR ext	1		
8	$t_{h(CKRL-DRV)}$	Hold time, DR valid after CLKR low	CLKR int	4		ns
			CLKR ext	4		
10	$t_{su(FXH-CKXL)}$	Setup time, external FSX high before CLKX low	CLKX int	13		ns
			CLKX ext	4		
11	$t_{h(CKXL-FXH)}$	Hold time, external FSX high after CLKX low	CLKX int	7		ns
			CLKX ext	3		

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

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

switching characteristics for McBSP†‡ (see Figure 32)

NO.	PARAMETER			’C6201-167 ’C6201-200		UNIT
				MIN	MAX	
1	$t_{d(CKSH-CKRXH)}$	Delay time, CLKS high to CLKR/X high for internal CLKR/X generated from CLKS input		4	15	ns
2	$t_{c(CKRX)}$	Cycle time, CLKR/X	CLKR/X int	2P§		CLKOUT1 cycles
3	$t_{w(CKRX)}$	Pulse duration, CLKR/X high or CLKR/X low	CLKR/X int	C – 1¶	C + 1¶	ns
4	$t_{d(CKRH-FRV)}$	Delay time, CLKR high to internal FSR valid	CLKR int	–2	4	ns
9	$t_{d(CKXH-FXV)}$	Delay time, CLKX high to internal FSX valid	CLKX int	0	6	ns
			CLKX ext	3	16	
12	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high#	CLKX int	–2	4	ns
			CLKX ext	3	16	
13	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid. This is also specified by design but not tested to be the delay time for data to be low impedance on the first data bit.	CLKX int	–2	4	ns
			CLKX ext	3	16	
14	$t_{d(FXH-DXV)}$	Delay time, FSX high to DX valid. This is also specified by design but not tested to be the delay time for data to be low impedance on the first data bit. ONLY applies when in data delay 0 (XDATDLY = 00b) mode.	FSX int	–2	4	ns
			FSX ext	3	16	

† 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.
§ 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.
¶ C = H or L
H = CLKX high pulse width = (CLKGDV/2 + 1) * T
L = CLKX low pulse width = (CLKGDV/2) * T
# Values specified by design but not tested

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

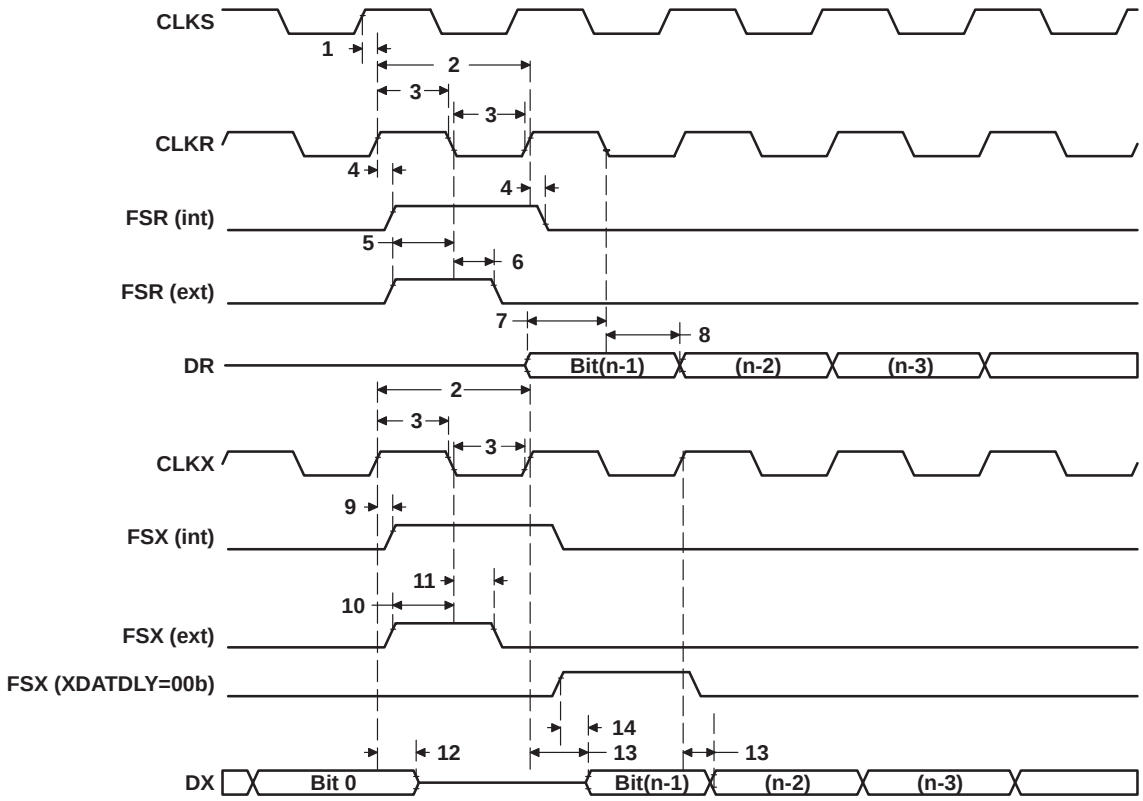


Figure 32. McBSP Timings

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

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

NO.			'C6201-167 'C6201-200		UNIT
			MIN	MAX	
1	$t_{su(FRH-CKSH)}$	Setup time, FSR high to CLKS high	4		ns
2	$t_{h(CKSH-FRH)}$	Hold time, FSR high after CLKS high	4		ns

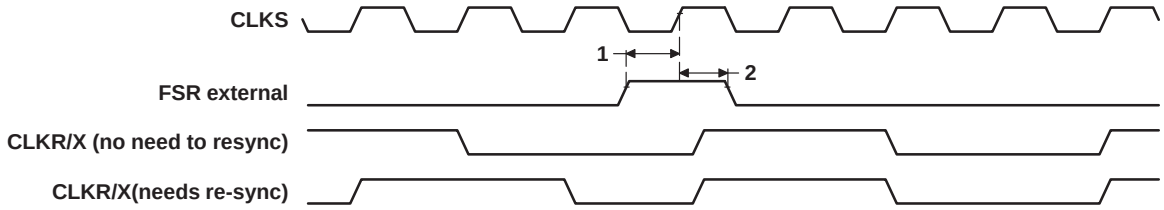


Figure 33. FSR Timing When GSYNC=1

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

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

NO.			'C6201-167 'C6201-200				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 + 3P		ns

† 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.
‡ 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 = 0†‡ (see Figure 34)

NO.	PARAMETER		'C6201-167 'C6201-200				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low¶	T – 2	T + 2			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high#	L – 2	L + 2			ns
3	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid. This is also specified by design but not tested to be the delay time for data to be low impedance on the first data bit.	–2	3	3P + 4	5P + 17	ns
6	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low‖	L – 2	L + 2			ns
7	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high‖			P + 4	3P + 17	ns
8	$t_{d(FXL-DXV)}$	Delay time, DX valid after FSX low			2P + 4	4P + 17	ns

† 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.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1
§ T = CLKX period = (1 + CLKGDV) * P, if CLKSM = 1 then P = 1/CPU clock frequency
= CLKX period = (1 + CLKGDV) * P_clks, if CLKSM = 0, P_clks = CLKS period.
H = CLKX high pulse width = (CLKGDV/2 + 1) * T
L = CLKX low pulse width = (CLKGDV/2) * T
¶ 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
# 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).
‖ Values specified by design but not tested

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

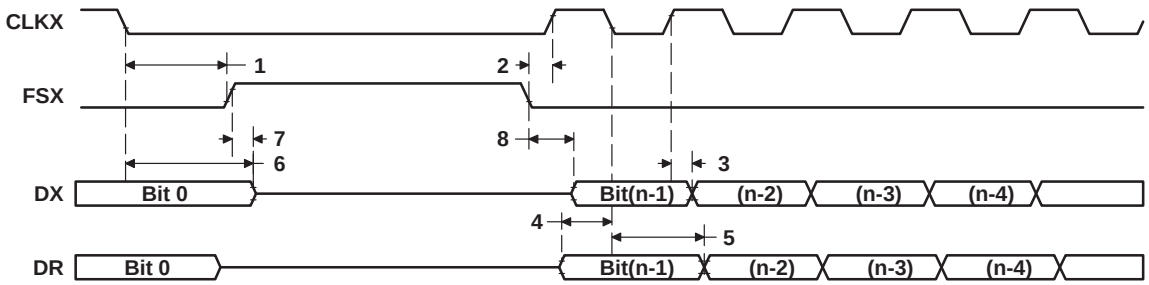


Figure 34. McBSP Timing as SPI Master or Slave: CLKSTP=10b, CLKXP = 0

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

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

NO.			'C6201-167 'C6201-200				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 + 3P		ns

† 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.
‡ 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 = 0†‡ (see Figure 35)

NO.	PARAMETER		'C6201-167 'C6201-200				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXL-FXL)}$	Hold time, FSX low after CLKX low¶	L – 2	L + 2			ns
2	$t_{d(FXL-CKXH)}$	Delay time, FSX low to CLKX high#	T – 2	T + 2			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid	–2	3	3P + 4	5P + 17	ns
6	$t_{dis(CKXL-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX low‖	–2	3	3P + 4	5P + 17	ns
7	$t_{d(FXL-DXV)}$	Delay time, DX valid after FSX low	H – 2	H + 3	2P + 4	4P + 17	ns

† 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.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1
§ T = CLKX period = (1 + CLKGDV) * P, if CLKSM = 1 then P = 1/CPU clock frequency
= CLKX period = (1 + CLKGDV) * P_clks, if CLKSM = 0, P_clks = CLKS period.
H = CLKX high pulse width = (CLKGDV/2 + 1) * T
L = CLKX low pulse width = (CLKGDV/2) * T
¶ 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
# 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).
‖ Values specified by design but not tested

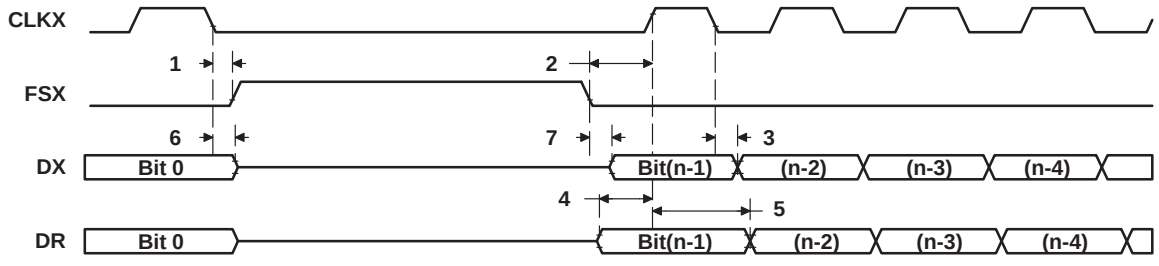


Figure 35. McBSP Timing as SPI Master or Slave: CLKSTP=11b, CLKXP = 0

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

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

NO.			'C6201-167 'C6201-200				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 + 3P		ns

† 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.
‡ 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		'C6201-167 'C6201-200				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high¶	T – 2	T + 2			ns
2	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low#	H – 2	H + 2			ns
3	$t_{d(CKXL-DXV)}$	Delay time, CLKX low to DX valid. This is also specified by design but not tested to be the delay time for data to be low impedance on the first data bit.	–2	3	3P + 4	5P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high‖	H – 2	H + 2			ns
7	$t_{dis(FXH-DXHZ)}$	Disable time, DX high impedance following last data bit from FSX high‖			P + 4	3P + 17	ns
8	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid			2P + 4	4P + 17	ns

† 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.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1
§ T = CLKX period = (1 + CLKGDV) * P, if CLKSM = 1 then P = 1/CPU clock frequency
= CLKX period = (1 + CLKGDV) * P_clks, if CLKSM = 0, P_clks = CLKS period.
H = CLKX high pulse width = (CLKGDV/2 + 1) * T
L = CLKX low pulse width = (CLKGDV/2) * T
¶ 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
# 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).
‖ Values specified by design but not tested

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

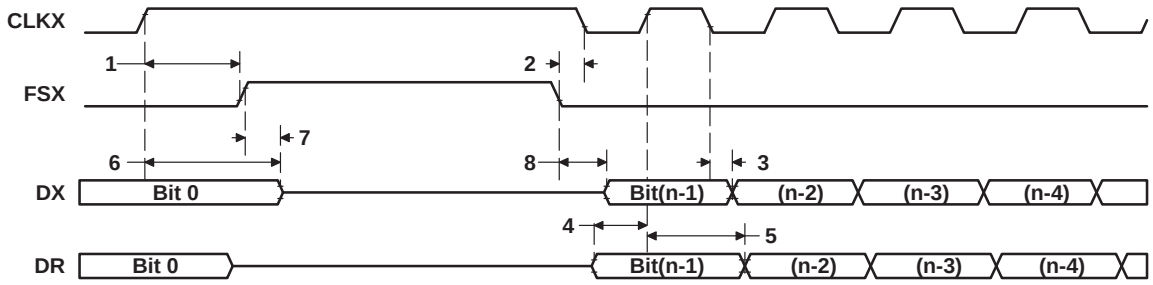


Figure 36. McBSP Timing as SPI Master or Slave: CLKSTP=10b, CLKXP=1

ADVANCE INFORMATION

MULTICHANNEL BUFFERED SERIAL PORT TIMING (CONTINUED)

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

NO.			'C6201-167 'C6201-200				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 + 3P		ns

† 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.
‡ 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		'C6201-167 'C6201-200				UNIT
			MASTER§		SLAVE		
			MIN	MAX	MIN	MAX	
1	$t_{h(CKXH-FXL)}$	Hold time, FSX low after CLKX high¶	H – 2	H + 2			ns
2	$t_{d(FXL-CKXL)}$	Delay time, FSX low to CLKX low#	T – 2	T			ns
3	$t_{d(CKXH-DXV)}$	Delay time, CLKX high to DX valid	–2	3	3P + 4	5P + 17	ns
6	$t_{dis(CKXH-DXHZ)}$	Disable time, DX high impedance following last data bit from CLKX high‖	–2	3	3P + 4	5P + 17	ns
7	$t_{d(FXL-DXV)}$	Delay time, FSX low to DX valid	L – 2	L + 3	2P + 4	4P + 17	ns

† 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.
‡ For all SPI slave modes, CLKG is programmed as 1/2 of the CPU clock by setting CLKSM = CLKGDV = 1
§ T = CLKX period = (1 + CLKGDV) * P, if CLKSM = 1 then P = 1/CPU clock frequency
= CLKX period = (1 + CLKGDV) * P_clks, if CLKSM = 0, P_clks = CLKS period.
H = CLKX high pulse width = (CLKGDV/2 + 1) * T
L = CLKX low pulse width = (CLKGDV/2) * T
¶ 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
# 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).
‖ Values specified by design but not tested

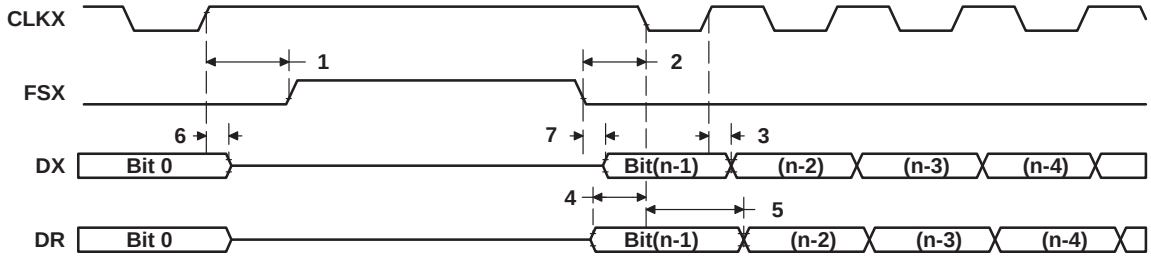


Figure 37. McBSP Timing as SPI Master or Slave: CLKSTP=11b, CLKXP=1

ADVANCE INFORMATION

DMAC, TIMER, POWER-DOWN TIMING

switching characteristics for DMAC outputs (see Figure 38)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{d(CKO1H-DMACV)}$	Delay time, CLKOUT1 high to DMAC valid	3	8	ns

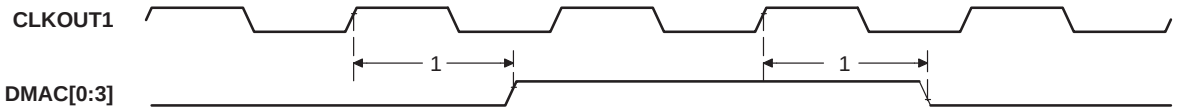


Figure 38. DMAC Timing

timing requirements for timer inputs (see Figure 39)

NO.			’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{w(TINPH)}$	Pulse duration, TINP high	2		CLKOUT1 cycles

switching characteristics for timer outputs (see Figure 39)

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
2	$t_{d(CKO1H-TOUTV)}$	Delay time, CLKOUT1 high to TOUT valid	3	9	ns

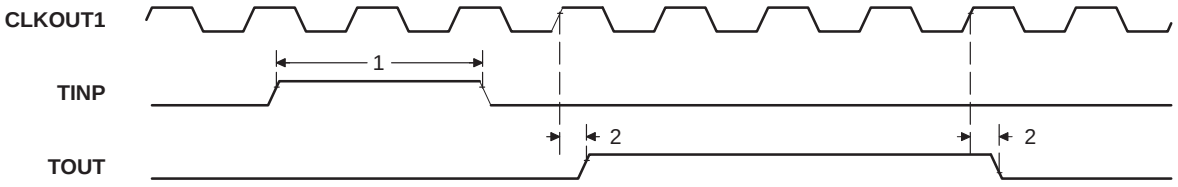


Figure 39. Timer Timing

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

NO.	PARAMETER		’C6201-167 ’C6201-200		UNIT
			MIN	MAX	
1	$t_{d(CKO1H-PDV)}$	Delay time, CLKOUT1 high to PD valid	3	5	ns

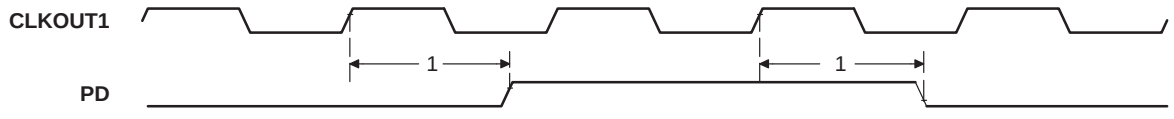
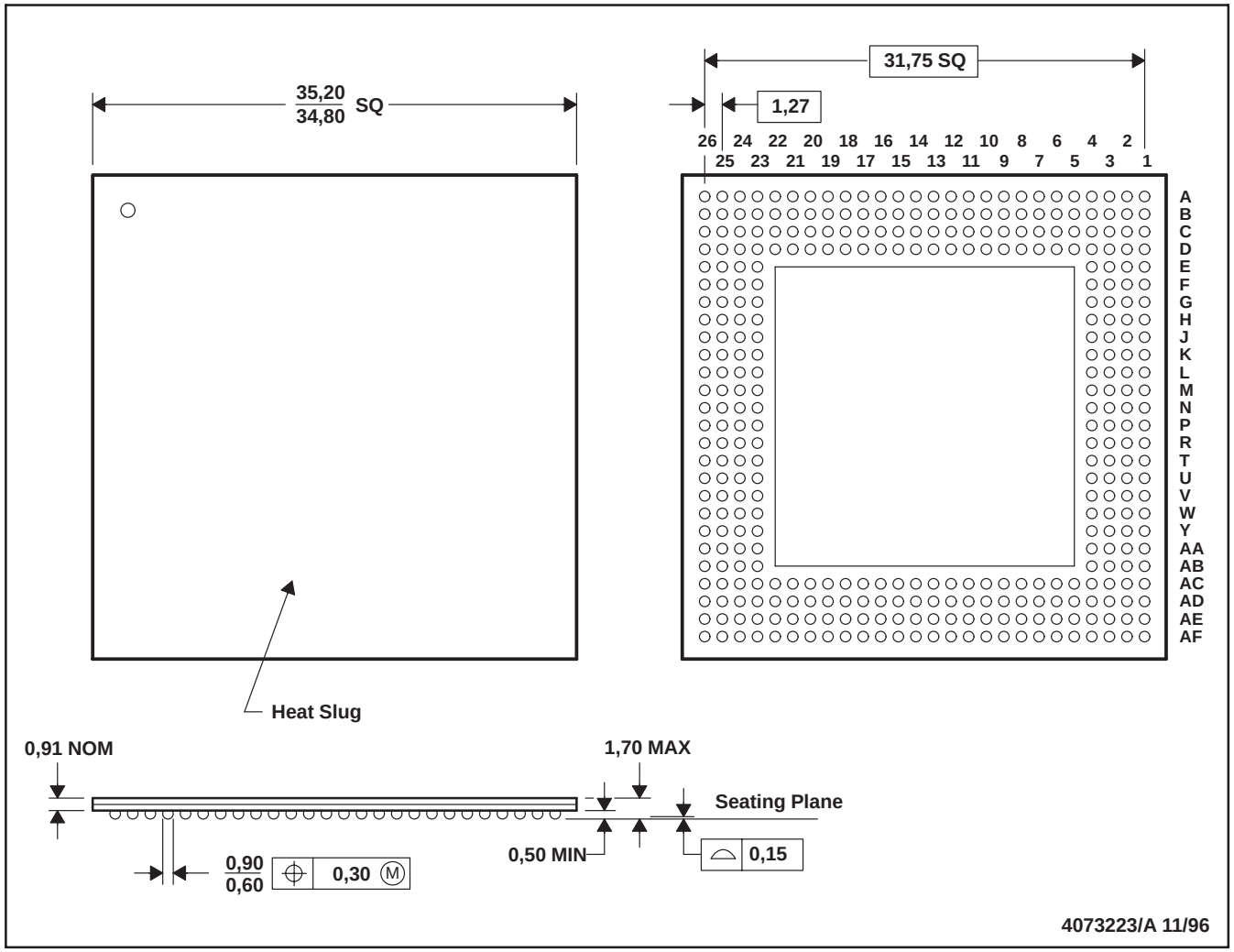


Figure 40. Power-Down Timing

MECHANICAL DATA

GGP (S-PBGA-N352) **PLASTIC BALL GRID ARRAY (CAVITY DOWN) PACKAGE**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Thermally enhanced die down plastic package with top surface metal heat slug.

ADVANCE INFORMATION

thermal resistance characteristics (S-PBGA package)

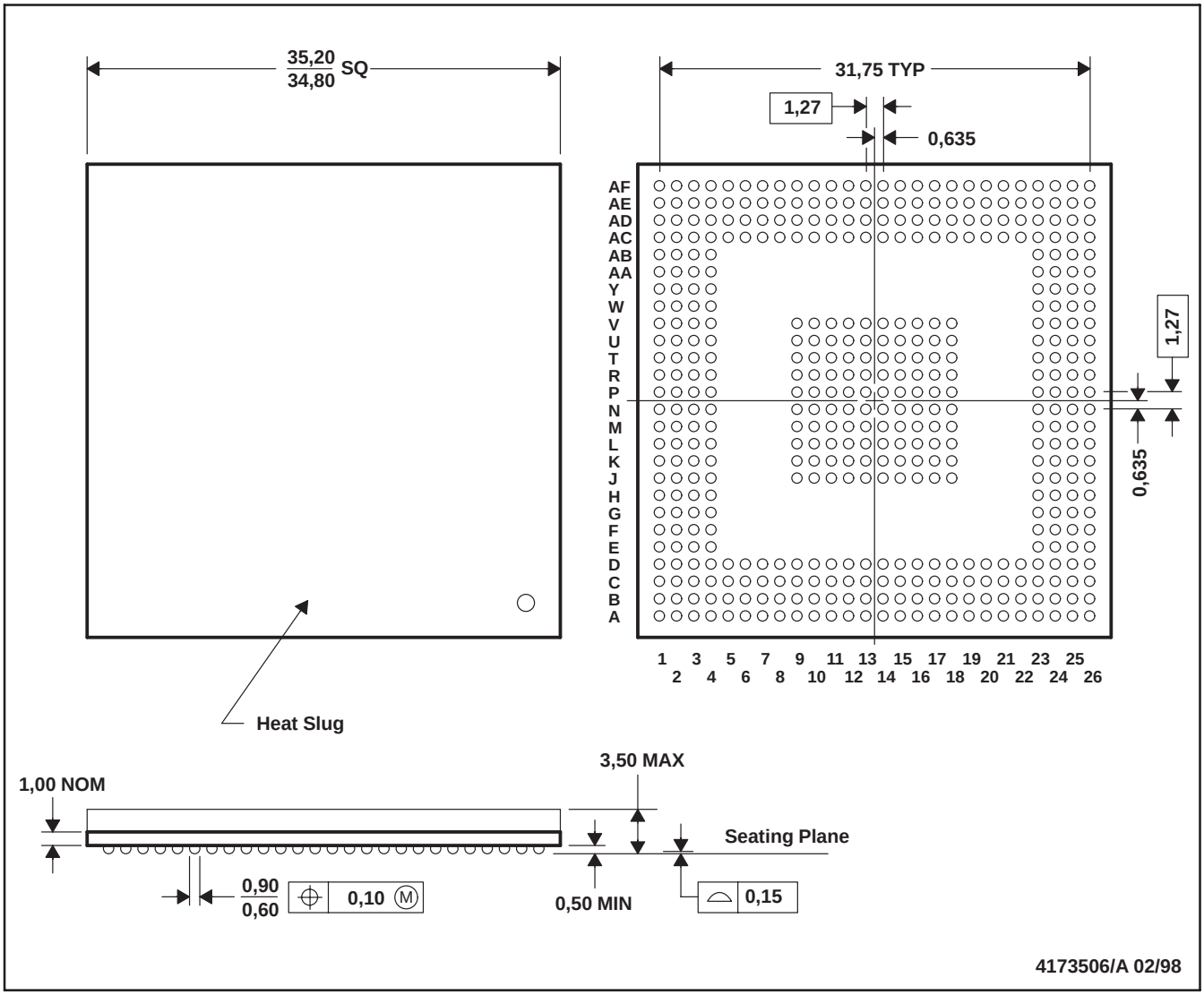
NO		°C/W	Air Flow LFPM†
1	$R\Theta_{JC}$ Junction-to-case	0.94	N/A
2	$R\Theta_{JA}$ Junction-to-free air	11.11	0
3	$R\Theta_{JA}$ Junction-to-free air	9.61	100
4	$R\Theta_{JA}$ Junction-to-free air	8.24	250
5	$R\Theta_{JA}$ Junction-to-free air	7.10	500

† LFPM = Linear Feet Per Minute

ADVANCE INFORMATION

MECHANICAL DATA

GJC (S-PBGA-N452) **PLASTIC BALL GRID ARRAY**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.
C. Thermally enhanced plastic package with heat slug (HSL).
D. Falls within JEDEC MO-151/BAR-2
E. Flip chip application only.

thermal resistance characteristics (S-PBGA package)

NO			°C/W	Air Flow LFPM†
1	$R\Theta_{JC}$	Junction-to-case	0.94	N/A
2	$R\Theta_{JA}$	Junction-to-free air	11.11	0
3	$R\Theta_{JA}$	Junction-to-free air	9.61	100
4	$R\Theta_{JA}$	Junction-to-free air	8.24	250
5	$R\Theta_{JA}$	Junction-to-free air	7.10	500

† LFPM = Linear Feet Per Minute

IMPORTANT NOTICE

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

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

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

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

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.