MOTOROLA
Semiconductor Products Sector

Order Number: DSP56311/D
Rev. 0.5, 6/2000

DSP56311

Advance Information

24-BIT DIGITAL SIGNAL PROCESSOR

The Motorola DSP56311, a member of the DSP56300 family of programmable digital signal processors (DSPs), supports wireless infrastructure applications with general filtering operations. The on-chip enhanced filter coprocessor (EFCOP) processes filter algorithms in parallel with core operation, thus increasing overall DSP performance and efficiency. Like the other family members, the DSP56311 uses a high-performance, single-clock-cycle-per-instruction engine (code-compatible with Motorola's popular DSP56000 core family), a barrel shifter, 24-bit addressing, an instruction cache, and a direct memory access (DMA) controller, as shown in **Figure 1**. The DSP56311 performs at 150 million instructions per second (MIPS), attaining 255 MIPS when the EFCOP is used in filtering applications. It has an internal 150 MHz clock with a 1.8 volt core and independent 3.3 volt input/output (I/O) power.

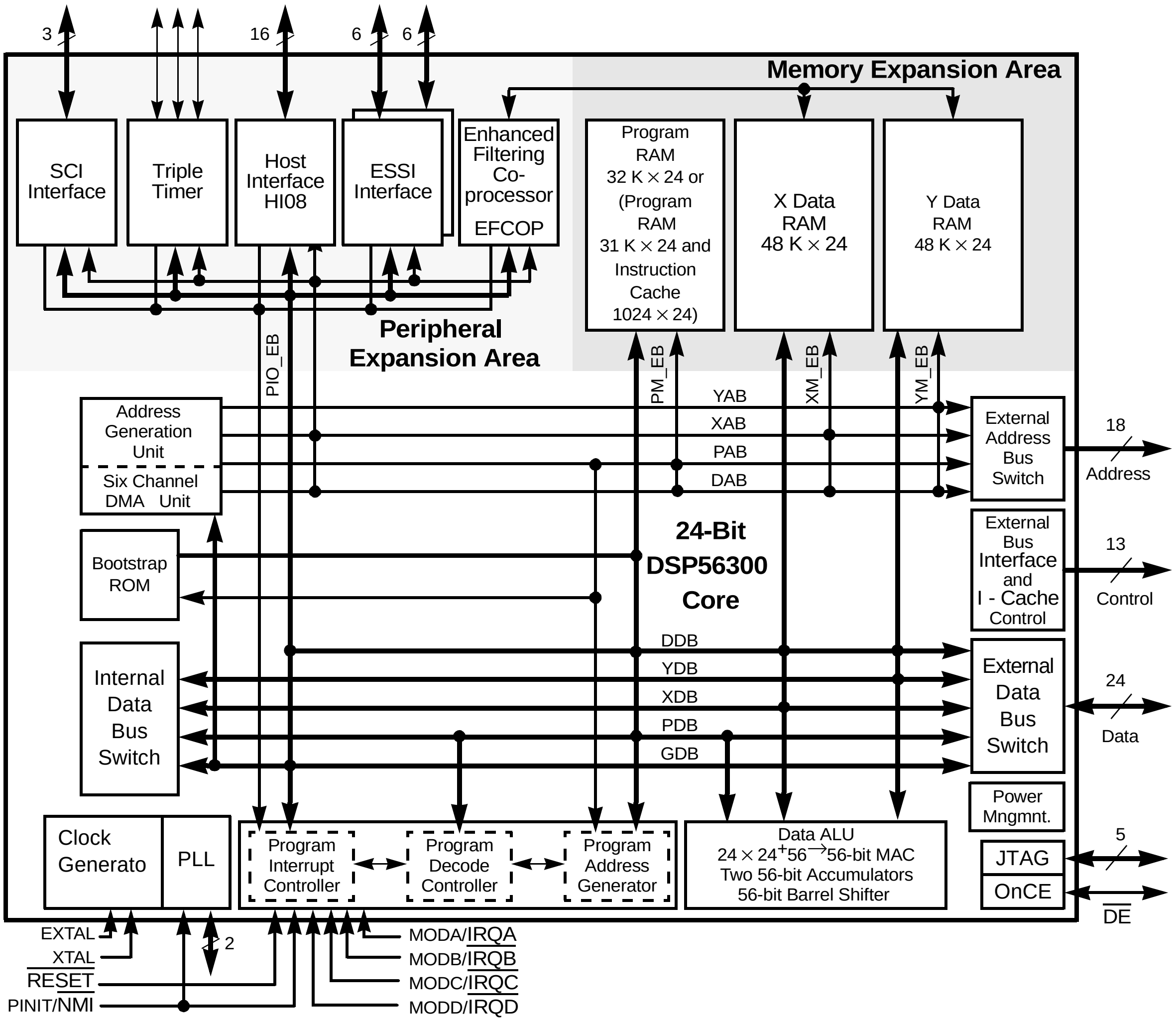


Figure 1. DSP56311 Block Diagram

CONTENTS

FOR TECHNICAL ASSISTANCE:

Telephone:	1-800-521-6274
Email:	dsphelp@dsp.sps.mot.com
Internet:	http://www.mot.com/SPS/DSP/

Data Sheet Conventions

$\overline{\text{OVERBAR}}$	Indicates a signal that is active when pulled low (For example, the $\overline{\text{RESET}}$ pin is active when low.)
“asserted”	A high true (active high) signal is high or a low true (active low) signal is low
“deasserted”	A high true (active high) signal is low or a low true (active low) signal is high

Examples:

Signal/Symbol	Logic State	Signal State	Voltage*
$\overline{\text{PIN}}$	True	Asserted	V_{IL}/V_{OL}
$\overline{\text{PIN}}$	False	Deasserted	V_{IH}/V_{OH}
PIN	True	Asserted	V_{IH}/V_{OH}
PIN	False	Deasserted	V_{IL}/V_{OL}

Note: *Values for V_{IL}, V_{OL}, V_{IH}, and V_{OH} are defined by individual product specifications.

FEATURES

High-Performance DSP56300 Core

- 150 million instructions per second (MIPS) (255 MIPS using the EFCOP in filtering applications) with a 150 MHz clock at 1.8 V core and 3.3 V I/O
- Object code compatible with the DSP56000 core
- Highly parallel instruction set
- Data arithmetic logic unit (ALU)
  - Fully pipelined 24 x 24-bit parallel multiplier-accumulator
  - 56-bit parallel barrel shifter (fast shift and normalization; bit stream generation and parsing)
  - Conditional ALU instructions
  - 24-bit or 16-bit arithmetic support under software control
- Program control unit (PCU)
  - Position independent code (PIC) support
  - Addressing modes optimized for DSP applications (including immediate offsets)
  - On-chip instruction cache controller
  - On-chip memory-expandable hardware stack
  - Nested hardware DO loops
  - Fast auto-return interrupts
- Direct memory access (DMA)
  - Six DMA channels supporting internal and external accesses
  - One-, two-, and three- dimensional transfers (including circular buffering)
  - End-of-block-transfer interrupts
  - Triggering from interrupt lines and all peripherals
- Phase-locked loop (PLL)
  - Allows change of low-power divide factor (DF) without loss of lock
  - ~~Output clock with skew elimination~~
- Hardware debugging support
  - On-Chip Emulation (OnCE™) module
  - Joint test action group (JTAG) test access port (TAP)
  - ~~Address trace mode reflects internal Program RAM accesses at the external port~~

Enhanced Filtering Coprocessor (EFCOP)

The on-chip filtering and echo-cancellation coprocessor (EFCOP) runs in parallel to the DSP core.

On-Chip Memories

- 128 K on-chip RAM total
- Program RAM, Instruction Cache, X data RAM, and Y data RAM sizes are programmable:

Program RAM Size	Instruction Cache Size	X Data RAM Size*	Y Data RAM Size*	Instruction Cache	Switch Mode	MSW1	MSW0
32K × 24-bit	0	48K × 24-bit	48K × 24-bit	disabled	disabled	0/1	0/1
31K × 24-bit	1024 × 24-bit	48K × 24-bit	48K × 24-bit	enabled	disabled	0/1	0/1
96K × 24-bit	0	16K × 24-bit	16K × 24-bit	disabled	enabled	0	0
95K × 24-bit	1024 × 24-bit	16K × 24-bit	16K × 24-bit	enabled	enabled	0	0
80K × 24-bit	0	24K × 24-bit	24K × 24-bit	disabled	enabled	0	1
79K × 24-bit	1024 × 24-bit	24K × 24-bit	24K × 24-bit	enabled	enabled	0	1
64K × 24-bit	0	32K × 24-bit	32K × 24-bit	disabled	enabled	1	0
63K × 24-bit	1024 × 24-bit	32K × 24-bit	32K × 24-bit	enabled	enabled	1	0
48K × 24-bit	0	40K × 24-bit	40K × 24-bit	disabled	enabled	1	1
47K × 24-bit	1024 × 24-bit	40K × 24-bit	40K × 24-bit	enabled	enabled	1	1

*Includes 10K × 24-bit shared memory (that is, memory shared by the core and the EFCOP)

- 192 x 24-bit bootstrap ROM

Off-Chip Memory Expansion

- Data memory expansion to two 256K × 24-bit word memory spaces (or up to two 4 M × 24-bit word memory spaces by using the address attribute AA0–AA3 signals)
- Program memory expansion to one 256K × 24-bit words memory space (or up to one 4 M × 24-bit word memory space by using the address attribute AA0–AA3 signals)
- External memory expansion port
- Chip Select Logic for glueless interface to static random access memory (SRAMs)
- ~~On-chip DRAM Controller for glueless interface to dynamic random access memory (DRAMs)~~

DRAM Access Support

DRAM accesses are not supported in the DSP56311 at 150 MHz.

On-Chip Peripherals

- Enhanced DSP56000-like 8-bit parallel host interface (HI08) supports a variety of buses (for example, ISA) and provides glueless connection to a number of industry-standard microcomputers, microprocessors, and DSPs
- Two enhanced synchronous serial interfaces (ESSI), each with one receiver and three transmitters (allows six-channel home theater)
- Serial communications interface (SCI) with baud rate generator
- Triple timer module
- Up to 34 programmable general-purpose input/output (GPIO) pins, depending on which peripherals are enabled

Reduced Power Dissipation

- Very low-power CMOS design
- Wait and Stop low-power standby modes
- Fully static design specified to operate at 0 Hz (dc)
- Optimized power management circuitry (instruction-dependent, peripheral-dependent, and mode-dependent)

Packaging

The DSP56311 is available in a 196-pin PBGA package.

TARGET APPLICATIONS

The DSP56311 is intended for applications requiring a large amount of on-chip memory, such as wireless infrastructure applications. The EFCOP can accelerate general filtering applications, such as echo-cancellation applications, correlation, and general-purpose convolution-based algorithms.

PRODUCT DOCUMENTATION

The three documents listed in the following table are required for a complete description of the DSP56311 and are necessary to design properly with the part. Documentation is available from the following sources. (See the back cover for detailed information.)

- A local Motorola distributor
- A Motorola semiconductor sales office
- A Motorola Literature Distribution Center
- The World Wide Web (WWW)

DSP56311 Documentation

Name	Description	Order Number
DSP56300 Family Manual	Detailed description of the DSP56300 family processor core and instruction set	DSP56300FM/AD
DSP56311 User's Manual	Detailed functional description of the DSP56311 memory configuration, operation, and register programming	DSP56311UM/D
DSP56311 Technical Data	DSP56311 features list and physical, electrical, timing, and package specifications	DSP56311DS/D

SECTION 1

SIGNALS/CONNECTIONS

SIGNAL GROUPINGS

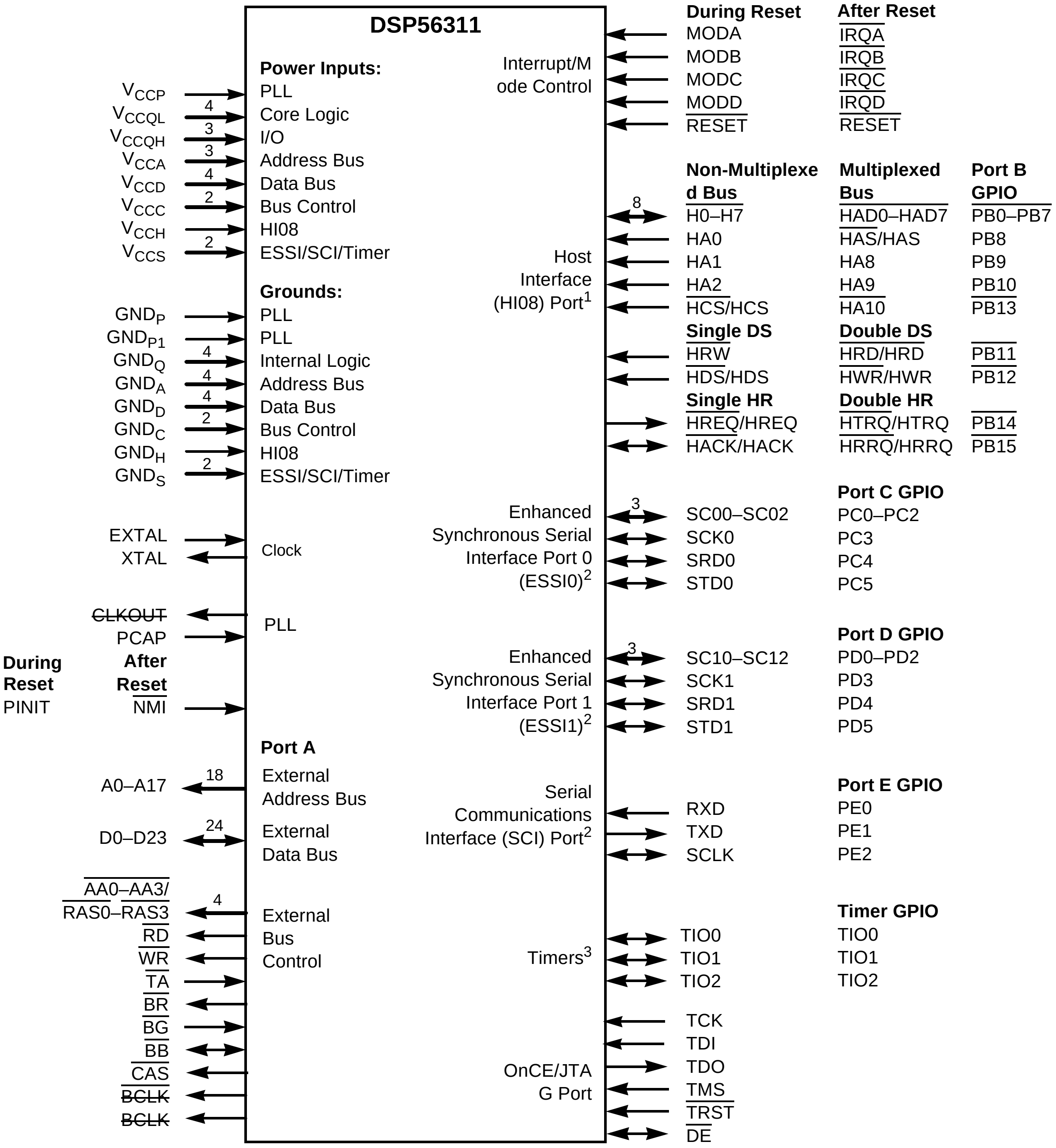
The DSP56311 input and output signals are organized into functional groups as shown in **Table 1-1.** **Figure 1-1.** diagrams the DSP56311 signals by functional group. The remainder of this chapter describes the signal pins in each functional group.

Table 1-1. DSP56311 Functional Signal Groupings

Functional Group		Number of Signals
Power (V_{CC})		20
Ground (GND)		19
Clock		2
PLL		3
Address bus	Port A[1]	18
Data bus		24
Bus control		13
Interrupt and mode control		5
Host interface (HI08)	Port B[2]	16
Enhanced synchronous serial interface (ESSI)	Ports C and D[3]	12
Serial communication interface (SCI)	Port E[4]	3
Timer		3
OnCE/JTAG Port		6

Notes:
1. Port A signals define the external memory interface port, including the external address bus, data bus, and control signals.
2. Port B signals are the HI08 port signals multiplexed with the GPIO signals.
3. Port C and D signals are the two ESSI port signals multiplexed with the GPIO signals.
4. Port E signals are the SCI port signals multiplexed with the GPIO signals.

Note: The Clock Output (CLKOUT) is not functional in the DSP56311. The CLKOUT output pin provides a 50 percent duty cycle output clock synchronized to the internal processor clock when the Phase Lock Loop (PLL) is enabled and locked. At 150 MHz and above, CLKOUT produces a low-amplitude waveform that is not usable externally by other devices. Several alternatives to using CLKOUT exist, such as enabling bus arbitration by setting the Asynchronous Bus Arbitration Enable BIt (ABE) in the Operating Mode register. When set, the ABE bit eliminates the setup and hold time requirements with respect to CLKOUT for $\overline{\text{BB}}$ and $\overline{\text{BG}}$.



Note:
1. The HI08 port supports a non-multiplexed or a multiplexed bus, single or double Data Strobe (DS), and single or double Host Request (HR) configurations. Since each of these modes is configured independently, any combination of these modes is possible. These HI08 signals can also be configured alternately as GPIO signals (PB0–PB15). Signals with dual designations (for example, HAS/HAS) have configurable polarity.
2. The ESSI0, ESSI1, and SCI signals are multiplexed with the Port C GPIO signals (PC0–PC5), Port D GPIO signals (PD0–PD5), and Port E GPIO signals (PE0–PE2), respectively.
3. TIO0–TIO2 can be configured as GPIO signals.

Figure 1-1. Signals Identified by Functional Group

POWER

Table 1-2. Power Inputs

Power Name	Description
V_{CCP}	**PLL Power**—V_{CCP} is V_{CC} dedicated for PLL use. The voltage should be well-regulated and the input should be provided with an extremely low impedance path to the V_{CC} power rail.
V_{CCQL}	**Quiet Core (Low) Power**—V_{CCQL} is an isolated power for the core processing logic. This input must be isolated externally from all other chip power inputs. The user must provide adequate external decoupling capacitors.
V_{CCQH}	**Quiet External (High) Power**—V_{CCQH} is a quiet power source for I/O lines. This input must be tied externally to all other chip power inputs, *except* V_{CCQL}. The user must provide adequate decoupling capacitors.
V_{CCA}	**Address Bus Power**—V_{CCA} is an isolated power for sections of the address bus I/O drivers. This input must be tied externally to all other chip power inputs, *except* V_{CCQL}. The user must provide adequate external decoupling capacitors.
V_{CCD}	**Data Bus Power**—V_{CCD} is an isolated power for sections of the data bus I/O drivers. This input must be tied externally to all other chip power inputs, *except* V_{CCQL}. The user must provide adequate external decoupling capacitors.
V_{CCC}	**Bus Control Power**—V_{CCC} is an isolated power for the bus control I/O drivers. This input must be tied externally to all other chip power inputs**,** *except* V_{CCQL}. The user must provide adequate external decoupling capacitors.
V_{CCH}	**Host Power**—V_{CCH} is an isolated power for the HI08 I/O drivers. This input must be tied externally to all other chip power inputs**,** *except* V_{CCQL}. The user must provide adequate external decoupling capacitors.
V_{CCS}	**ESSI, SCI, and Timer Power**—V_{CCS} is an isolated power for the ESSI, SCI, and timer I/O drivers. This input must be tied externally to all other chip power inputs, *except* V_{CCQL}. The user must provide adequate external decoupling capacitors.

GROUND

Table 1-3. Grounds

Ground Name	Description
GND_P	**PLL Ground**—GND_P is ground-dedicated for PLL use. The connection should be provided with an extremely low-impedance path to ground. V_{CCP} should be bypassed to GND_P by a 0.47 mF capacitor located as close as possible to the chip package.

Table 1-3. Grounds

Ground Name	Description
GND_{P1}	**PLL Ground 1**—GND_{P1} is ground-dedicated for PLL use. The connection should be provided with an extremely low-impedance path to ground.
GND_Q	**Quiet Ground**—GND_Q is an isolated ground for the internal processing logic. This connection must be tied externally to all other chip ground connections. The user must provide adequate external decoupling capacitors.
GND_A	**Address Bus Ground**—GND_A is an isolated ground for sections of the address bus I/O drivers. This connection must be tied externally to all other chip ground connections. The user must provide adequate external decoupling capacitors. There are four GND_A connections.
GND_D	**Data Bus Ground**—GND_D is an isolated ground for sections of the data bus I/O drivers. This connection must be tied externally to all other chip ground connections. The user must provide adequate external decoupling capacitors.
GND_C	**Bus Control Ground**—GND_C is an isolated ground for the bus control I/O drivers. This connection must be tied externally to all other chip ground connections. The user must provide adequate external decoupling capacitors.
GND_H	**Host Ground**—GND_H is an isolated ground for the HI08 I/O drivers. This connection must be tied externally to all other chip ground connections. The user must provide adequate external decoupling capacitors.
GND_S	**ESSI, SCI, and Timer Ground**—GND_S is an isolated ground for the ESSI, SCI, and timer I/O drivers. This connection must be tied externally to all other chip ground connections. The user must provide adequate external decoupling capacitors.

CLOCK

Table 1-4. Clock Signals

Signal Name	Type	State During Reset	Signal Description
EXTAL	Input	Input	**External Clock/Crystal Input**—EXTAL interfaces the internal crystal oscillator input to an external crystal or an external clock.
XTAL	Output	Chip-driven	**Crystal Output**—XTAL connects the internal crystal oscillator output to an external crystal. If an external clock is used, leave XTAL unconnected.

PLL

Table 1-5. Phase-Locked Loop Signals

Signal Name	Type	State During Reset	Signal Description
PCAP	Input	Input	**PLL Capacitor**—PCAP is an input connecting an off-chip capacitor to the PLL filter. Connect one capacitor terminal to PCAP and the other terminal to V_{CCP}. If the PLL is not used, PCAP may be tied to V_{CC}, GND, or left floating.
~~CLKOUT~~	~~Output~~	~~Chip-driven~~	At speeds made possible by HiP4 process technology, CLKOUT produces a low-amplitude waveform that is not usable externally by other devices. Alternatives to using CLKOUT exist. One example is the use of the Asynchronous Bus Arbitration Enable Bit (ABE) in the Operating Mode register. When set, the ABE bit eliminates the setup and hold time requirements with respect to CLKOUT for $\overline{BB}$ and $\overline{BG}$. Future changes in process technology may continue to produce alternatives to CLKOUT.
PINIT $\overline{NMI}$	Input Input	Input	**PLL Initial**—During assertion of $\overline{RESET}$, the value of PINIT is written into the PLL enable (PEN) bit of the PLL control (PCTL) register, determining whether the PLL is enabled or disabled. **Nonmaskable Interrupt**—After $\overline{RESET}$ deassertion and during normal instruction processing, this Schmitt-trigger input is the negative-edge-triggered NMI request internally synchronized to CLKOUT.

EXTERNAL MEMORY EXPANSION PORT (PORT A)

Note: When the DSP56311 enters a low-power standby mode (stop or wait), it releases bus mastership and tri-states the relevant Port A signals: A0–A17, D0–D23, AA0/$\overline{RAS0}$–AA3/$\overline{RAS3}$, $\overline{RD}$, $\overline{WR}$, $\overline{BB}$, $\overline{CAS}$, BCLK, $\overline{BCLK}$.

External Address Bus

Table 1-6. External Address Bus Signals

Signal Name	Type	State During Reset	Signal Description
A0–A17	Output	Tri-stated	**Address Bus**—When the DSP is the bus master, A0–A17 are active-high outputs that specify the address for external program and data memory accesses. Otherwise, the signals are tri-stated. To minimize power dissipation, A0–A17 do not change state when external memory spaces are not being accessed.

External Data Bus

Table 1-7. External Data Bus Signals

Signal Name	Type	State During Reset	Signal Description
D0–D23	Input/ Output	Tri-stated	**Data Bus**—When the DSP is the bus master, D0–D23 are active-high, bidirectional input/outputs that provide the bidirectional data bus for external program and data memory accesses. Otherwise, D0–D23 are tri-stated. These lines have weak keepers to maintain the last state even if all drivers are tri-stated.

External Bus Control

Table 1-8. External Bus Control Signals

Signal Name	Type	State During Reset	Signal Description
AA0–AA3 $\overline{\text{RAS0}}$–$\overline{\text{RAS3}}$	Output Output	Tri-stated	**Address Attribute**—When defined as AA, these signals can be used as chip selects or additional address lines. The default use defines a priority scheme under which only one AA signal can be asserted at a time. Setting the AA priority disable (APD) bit (Bit 14) of the OMR, the priority mechanism is disabled and the lines can be used together as four external lines that can be decoded externally into 16 chip select signals. **Row Address Strobe**—When defined as $\overline{\text{RAS}}$, these signals can be used as $\overline{\text{RAS}}$ for DRAM interface. These signals are tri-statable outputs with programmable polarity.
$\overline{\text{RD}}$	Output	Tri-stated	**Read Enable**—When the DSP is the bus master, $\overline{\text{RD}}$ is an active-low output that is asserted to read external memory on the data bus (D0–D23). Otherwise, $\overline{\text{RD}}$ is tri-stated.
$\overline{\text{WR}}$	Output	Tri-stated	**Write Enable**—When the DSP is the bus master, $\overline{\text{WR}}$ is an active-low output that is asserted to write external memory on the data bus (D0–D23). Otherwise, the signals are tri-stated.
$\overline{\text{TA}}$	Input	Ignored Input	**Transfer Acknowledge**—If the DSP56311 is the bus master and there is no external bus activity, or the DSP56311 is not the bus master, the $\overline{\text{TA}}$ input is ignored. The $\overline{\text{TA}}$ input is a data transfer acknowledge (DTACK) function that can extend an external bus cycle indefinitely. Any number of wait states (1, 2. . .infinity) may be added to the wait states inserted by the bus control register (BCR) by keeping $\overline{\text{TA}}$ deasserted. In typical operation, $\overline{\text{TA}}$ is deasserted at the start of a bus cycle, is asserted to enable completion of the bus cycle, and is deasserted before the next bus cycle. The current bus cycle completes one clock period after $\overline{\text{TA}}$ is asserted synchronous to CLKOUT. The number of wait states is determined by the $\overline{\text{TA}}$ input or by the BCR, whichever is longer. The BCR can be used to set the minimum number of wait states in external bus cycles. In order to use the $\overline{\text{TA}}$ functionality, the BCR must be programmed to at least one wait state. A zero wait state access cannot be extended by $\overline{\text{TA}}$ deassertion; otherwise, improper operation may result. $\overline{\text{TA}}$ can operate synchronously or asynchronously depending on the setting of the TAS bit in the OMR. $\overline{\text{TA}}$ functionality may not be used while performing DRAM type accesses; otherwise, improper operation may result.

Table 1-8. External Bus Control Signals (Continued)

Signal Name	Type	State During Reset	Signal Description
$\overline{\text{BR}}$	Output	Output (deasserted)	**Bus Request**—$\overline{\text{BR}}$ is an active-low output, never tri-stated. $\overline{\text{BR}}$ is asserted when the DSP requests bus mastership. $\overline{\text{BR}}$ is deasserted when the DSP no longer needs the bus. $\overline{\text{BR}}$ may be asserted or deasserted independently of whether the DSP56311 is a bus master or a bus slave. Bus “parking” allows $\overline{\text{BR}}$ to be deasserted even though the DSP56311 is the bus master. (See the description of bus “parking” in the $\overline{\text{BB}}$ signal description.) The bus request hole (BRH) bit in the BCR allows $\overline{\text{BR}}$ to be asserted under software control even though the DSP does not need the bus. $\overline{\text{BR}}$ is typically sent to an external bus arbitrator that controls the priority, parking, and tenure of each master on the same external bus. $\overline{\text{BR}}$ is only affected by DSP requests for the external bus, never for the internal bus. During hardware reset, $\overline{\text{BR}}$ is deasserted and the arbitration is reset to the bus slave state.
$\overline{\text{BG}}$	Input	Ignored Input	**Bus Grant**—$\overline{\text{BG}}$ is an active-low input. ~~$\overline{\text{BG}}$ must be asserted/deasserted synchronous to CLKOUT for proper operation.~~ $\overline{\text{BG}}$ is asserted by an external bus arbitration circuit when the DSP56311 becomes the next bus master. When $\overline{\text{BG}}$ is asserted, the DSP56311 must wait until $\overline{\text{BB}}$ is deasserted before taking bus mastership. When $\overline{\text{BG}}$ is deasserted, bus mastership is typically given up at the end of the current bus cycle. This may occur in the middle of an instruction that requires more than one external bus cycle for execution. The default operation of this bit requires a setup and hold time as specified in *DSP56311 Technical Data* (the data sheet). An alternate mode can be invoked: set the asynchronous bus arbitration enable (ABE) bit (Bit 13) in the OMR. When this bit is set, $\overline{\text{BG}}$ and $\overline{\text{BB}}$ are synchronized internally. This eliminates the respective setup and hold time requirements but adds a required delay between the deassertion of an initial $\overline{\text{BG}}$ input and the assertion of a subsequent $\overline{\text{BG}}$ input.

Table 1-8. External Bus Control Signals (Continued)

Signal Name	Type	State During Reset	Signal Description
$\overline{BB}$	Input/Output	Input	**Bus Busy**—$\overline{BB}$ is a bidirectional active-low input/output ~~and must be asserted and deasserted synchronous to CLKOUT~~.$\overline{BB}$ indicates that the bus is active. Only after $\overline{BB}$ is deasserted can the pending bus master become the bus master (and then assert the signal again). The bus master may keep $\overline{BB}$ asserted after ceasing bus activity regardless of whether $\overline{BR}$ is asserted or deasserted. Called "bus parking," this allows the current bus master to reuse the bus without rearbitration until another device requires the bus. The deassertion of $\overline{BB}$ is done by an "active pull-up" method (that is, $\overline{BB}$ is driven high and then released and held high by an external pull-up resistor). The default operation of this bit requires a setup and hold time as specified in the *DSP56311 Technical Data sheet*. An alternate mode can be invoked: set the ABE bit (Bit 13) in the OMR. When this bit is set, $\overline{BG}$ and $\overline{BB}$ are synchronized internally. See $\overline{BG}$ for additional information. $\overline{BB}$ requires an external pull-up resistor.
$\overline{CAS}$	Output	Tri-stated	**Column Address Strobe**—When the DSP is the bus master, $\overline{CAS}$ is an active-low output used by DRAM to strobe the column address. Otherwise, if the bus mastership enable (BME) bit in the DRAM control register is cleared, the signal is tri-stated.
~~BCLK~~	~~Output~~	~~Tri-stated~~	**Bus Clock**—With the application of HiP4 process technology, BCLK does not function. Without BCLK functioning, no signal exists to initiate the sampling process, and the DSP does not output any addresses. Therefore, Address Trace mode is not supported under the HiP4 process.
~~$\overline{BCLK}$~~	~~Output~~	~~Tri-stated~~	~~**Bus Clock Not**—When the DSP is the bus master, $\overline{BCLK}$ is an active-low output and is the inverse of the BCLK signal. Otherwise, the signal is tri-stated.~~

INTERRUPT AND MODE CONTROL

The interrupt and mode control signals select the chip's operating mode as it comes out of hardware reset. After $\overline{RESET}$ is deasserted, these inputs are hardware interrupt request lines.

Table 1-9. Interrupt and Mode Control

Signal Name	Type	State During Reset	Signal Description
RESET	Input	Input	**Reset**—RESET is an active-low, Schmitt-trigger input. ~~Deassertion of RESET is internally synchronized to CLKOUT~~. When asserted, the chip is placed in the Reset state and the internal phase generator is reset. The Schmitt-trigger input allows a slowly rising input (such as a capacitor charging) to reset the chip reliably. ~~If RESET is deasserted synchronous to CLKOUT, exact start-up timing is guaranteed, allowing multiple processors to start synchronously and operate together in “lock-step.”~~ When the RESET signal is deasserted, the initial chip operating mode is latched from the MODA, MODB, MODC, and MODD inputs. The RESET signal must be asserted after power up.
MODA IRQA	Input Input	Input	**Mode Select A**—MODA is an active-low Schmitt-trigger input~~,~~ ~~internally synchronized to CLKOUT~~. MODA, MODB, MODC, and MODD select one of 16 initial chip operating modes, latched into the OMR when the RESET signal is deasserted. **External Interrupt Request A**—After reset, this input becomes a level-sensitive or negative-edge-triggered, maskable interrupt request input during normal instruction processing. ~~If IRQA is asserted synchronous to CLKOUT, multiple processors can be resynchronized using the WAIT instruction and asserting IRQA to exit the wait state.~~ If the processor is in the stop standby state and IRQA is asserted, the processor will exit the stop state.
MODB IRQB	Input Input	Input	**Mode Select B**—MODB is an active-low Schmitt-trigger input~~,~~ ~~internally synchronized to CLKOUT~~. MODA, MODB, MODC, and MODD select one of 16 initial chip operating modes, latched into the OMR when the RESET signal is deasserted. **External Interrupt Request B**—After reset, this input becomes a level-sensitive or negative-edge-triggered, maskable interrupt request input during normal instruction processing. ~~If IRQB is asserted synchronous to CLKOUT, multiple processors can be resynchronized using the WAIT instruction and asserting IRQB to exit the wait state.~~ If the processor is in the stop standby state and IRQB is asserted, the processor will exit the stop state.
MODC IRQC	Input Input	Input	**Mode Select C**—MODC is an active-low Schmitt-trigger input~~,~~ ~~internally synchronized to CLKOUT~~. MODA, MODB, MODC, and MODD select one of 16 initial chip operating modes, latched into the OMR when the RESET signal is deasserted. **External Interrupt Request C**—After reset, this input becomes a level-sensitive or negative-edge-triggered, maskable interrupt request input during normal instruction processing. ~~If IRQC is asserted synchronous to CLKOUT, multiple processors can be resynchronized using the WAIT instruction and asserting IRQC to exit the wait state.~~ If the processor is in the stop standby state and IRQC is asserted, the processor will exit the stop state.

Table 1-9. Interrupt and Mode Control (Continued)

Signal Name	Type	State During Reset	Signal Description
MODD	Input	Input	**Mode Select D**—MODD is an active-low Schmitt-trigger input~~, internally synchronized to CLKOUT~~. MODA, MODB, MODC, and MODD select one of 16 initial chip operating modes, latched into the OMR when the $\overline{\text{RESET}}$ signal is deasserted.
$\overline{\text{IRQD}}$	Input		**External Interrupt Request D**—After reset, this input becomes a level-sensitive or negative-edge-triggered, maskable interrupt request input during normal instruction processing. ~~If $\overline{\text{IRQD}}$ is asserted synchronous to CLKOUT, multiple processors can be resynchronized using the WAIT instruction and asserting $\overline{\text{IRQD}}$ to exit the wait state.~~ If the processor is in the stop standby state and $\overline{\text{IRQD}}$ is asserted, the processor will exit the stop state.

HI08

The HI08 provides a fast, 8-bit, parallel data port that may be connected directly to the host bus. The HI08 supports a variety of standard buses and can be directly connected to a number of industry standard microcomputers, microprocessors, DSPs, and DMA hardware.

Table 1-10. Host Interface

Signal Name	Type	State During Reset	Signal Description
H0–H7	Input/ Output	Tri-stated	**Host Data**—When the HI08 is programmed to interface a nonmultiplexed host bus and the host interface (HI) function is selected, these signals are lines 0–7 of the data bidirectional, tri-state bus.
HAD0–HAD7	Input/ Output		**Host Address—**When HI08 is programmed to interface a multiplexed host bus and the HI function is selected, these signals are lines 0–7 of the address/data bidirectional, multiplexed, tri-state bus.
PB0–PB7	Input or Output		**Port B 0–7**—When the HI08 is configured as GPIO through the host port control register (HPCR), these signals are individually programmed as inputs or outputs through the HI08 data direction register (HDDR). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
HA0	Input	Input	**Host Address Input 0**—When the HI08 is programmed to interface a nonmultiplexed host bus and the HI function is selected, this signal is line 0 of the host address input bus.
$\overline{HAS}$/HAS	Input		**Host Address Strobe—**When HI08 is programmed to interface a multiplexed host bus and the HI function is selected, this signal is the host address strobe (HAS) Schmitt-trigger input. The polarity of the address strobe is programmable but is configured active-low ($\overline{HAS}$) following reset.
PB8	Input or Output		**Port B 8**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

Table 1-10. Host Interface (Continued)

Signal Name	Type	State During Reset	Signal Description
HA1	Input	Input	**Host Address Input 1**—When the HI08 is programmed to interface a nonmultiplexed host bus and the HI function is selected, this signal is line 1 of the host address (HA1) input bus.
HA8	Input		**Host Address 8**—When HI08 is programmed to interface a multiplexed host bus and the HI function is selected, this signal is line 8 of the host address (HA8) input bus.
PB9	Input or Output		**Port B 9**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
HA2	Input	Input	**Host Address Input 2**—When the HI08 is programmed to interface a nonmultiplexed host bus and the HI function is selected, this signal is line 2 of the host address (HA2) input bus.
HA9	Input		**Host Address 9**—When HI08 is programmed to interface a multiplexed host bus and the HI function is selected, this signal is line 9 of the host address (HA9) input bus.
PB10	Input or Output		**Port B 10**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
HRW	Input	Input	**Host Read/Write**—When HI08 is programmed to interface a single-data-strobe host bus and the HI function is selected, this signal is the Host Read/$\overline{\text{Write}}$ (HRW) input.
$\overline{\text{HRD}}$/HRD	Input		**Host Read Data**—When HI08 is programmed to interface a double-data-strobe host bus and the HI function is selected, this signal is the HRD strobe Schmitt-trigger input. The polarity of the data strobe is programmable, but is configured as active-low ($\overline{\text{HRD}}$) after reset.
PB11	Input or Output		**Port B 11**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

Table 1-10. Host Interface (Continued)

Signal Name	Type	State During Reset	Signal Description
$\overline{\mathrm{HDS}}$/HDS	Input	Input	**Host Data Strobe**—When HI08 is programmed to interface a single-data-strobe host bus and the HI function is selected, this signal is the host data strobe (HDS) Schmitt-trigger input. The polarity of the data strobe is programmable, but is configured as active-low ($\overline{\mathrm{HDS}}$) following reset.
$\overline{\mathrm{HWR}}$/HWR	Input		**Host Write Data**—When HI08 is programmed to interface a double-data-strobe host bus and the HI function is selected, this signal is the host write data strobe (HWR) Schmitt-trigger input. The polarity of the data strobe is programmable, but is configured as active-low ($\overline{\mathrm{HWR}}$) following reset.
PB12	Input or Output		**Port B 12**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
HCS	Input	Input	**Host Chip Select**—When HI08 is programmed to interface a nonmultiplexed host bus and the HI function is selected, this signal is the host chip select (HCS) input. The polarity of the chip select is programmable, but is configured active-low ($\overline{\mathrm{HCS}}$) after reset.
HA10	Input		**Host Address 10**—When HI08 is programmed to interface a multiplexed host bus and the HI function is selected, this signal is line 10 of the host address (HA10) input bus.
PB13	Input or Output		**Port B 13**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

Table 1-10. Host Interface (Continued)

Signal Name	Type	State During Reset	Signal Description
$\overline{\text{HREQ}}$/HREQ	Output	Input	**Host Request**—When HI08 is programmed to interface a single host request host bus and the HI function is selected, this signal is the host request (HREQ) output. The polarity of the host request is programmable, but is configured as active-low ($\overline{\text{HREQ}}$) following reset. The host request may be programmed as a driven or open-drain output.
$\overline{\text{HTRQ}}$/HTRQ	Output		**Transmit Host Request**—When HI08 is programmed to interface a double host request host bus and the HI function is selected, this signal is the transmit host request (HTRQ) output. The polarity of the host request is programmable, but is configured as active-low ($\overline{\text{HTRQ}}$) following reset. The host request may be programmed as a driven or open-drain output.
PB14	Input or Output		**Port B 14**—When the HI08 is programmed to interface a multiplexed host bus and the signal is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
$\overline{\text{HACK}}$/ HACK	Input	Input	**Host Acknowledge**—When HI08 is programmed to interface a single host request host bus and the HI function is selected, this signal is the host acknowledge (HACK) Schmitt-trigger input. The polarity of the host acknowledge is programmable, but is configured as active-low ($\overline{\text{HACK}}$) after reset.
$\overline{\text{HRRQ}}$/ HRRQ	Output		**Receive Host Request**—When HI08 is programmed to interface a double host request host bus and the HI function is selected, this signal is the receive host request (HRRQ) output. The polarity of the host request is programmable, but is configured as active-low ($\overline{\text{HRRQ}}$) after reset. The host request may be programmed as a driven or open-drain output.
PB15	Input or Output		**Port B 15**—When the HI08 is configured as GPIO through the HPCR, this signal is individually programmed as an input or output through the HDDR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

ENHANCED SYNCHRONOUS SERIAL INTERFACE 0

There are two synchronous serial interfaces (ESSI0 and ESSI1) that provide a full-duplex serial port for serial communication with a variety of serial devices, including one or more industry-standard codecs, other DSPs, microprocessors, and peripherals which implement the Motorola serial peripheral interface (SPI).

Table 1-11. Enhanced Synchronous Serial Interface 0

Signal Name	Type	State During Reset	Signal Description
SC00 PC0	Input or Output	Input	**Serial Control 0**—The function of SC00 is determined by the selection of either synchronous or asynchronous mode. For asynchronous mode, this signal will be used for the receive clock I/O (Schmitt-trigger input). For synchronous mode, this signal is used either for transmitter 1 output or for serial I/O flag 0. **Port C 0**—The default configuration following reset is GPIO input PC0. When configured as PC0, signal direction is controlled through the port directions register (PRR0). The signal can be configured as ESSI signal SC00 through the port control register (PCR0). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SC01 PC1	Input/ Output Input or Output	Input	**Serial Control 1**—The function of this signal is determined by the selection of either synchronous or asynchronous mode. For asynchronous mode, this signal is the receiver frame sync I/O. For synchronous mode, this signal is used either for transmitter 2 output or for serial I/O flag 1. **Port C 1**—The default configuration following reset is GPIO input PC1. When configured as PC1, signal direction is controlled through PRR0. The signal can be configured as an ESSI signal SC01 through PCR0. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SC02 PC2	Input/ Output Input or Output	Input	**Serial Control Signal 2**—SC02 is used for frame sync I/O. SC02 is the frame sync for both the transmitter and receiver in synchronous mode, and for the transmitter only in asynchronous mode. When configured as an output, this signal is the internally generated frame sync signal. When configured as an input, this signal receives an external frame sync signal for the transmitter (and the receiver in synchronous operation). **Port C 2**—The default configuration following reset is GPIO input PC2. When configured as PC2, signal direction is controlled through PRR0. The signal can be configured as an ESSI signal SC02 through PCR0. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

Table 1-11. Enhanced Synchronous Serial Interface 0 (Continued)

Signal Name	Type	State During Reset	Signal Description
SCK0	Input/ Output	Input	**Serial Clock**—SCK0 is a bidirectional Schmitt-trigger input signal providing the serial bit rate clock for the ESSI. The SCK0 is a clock input or output, used by both the transmitter and receiver in synchronous modes or by the transmitter in asynchronous modes. Although an external serial clock can be independent of and asynchronous to the DSP system clock, it must exceed the minimum clock cycle time of 6T (that is, the system clock frequency must be at least three times the external ESSI clock frequency). The ESSI needs at least three DSP phases inside each half of the serial clock.
PC3	Input or Output		**Port C 3**—The default configuration following reset is GPIO input PC3. When configured as PC3, signal direction is controlled through PRR0. The signal can be configured as an ESSI signal SCK0 through PCR0. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SRD0	Input/ Output	Input	**Serial Receive Data**—SRD0 receives serial data and transfers the data to the ESSI receive shift register. SRD0 is an input when data is being received.
PC4	Input or Output		**Port C 4**—The default configuration following reset is GPIO input PC4. When configured as PC4, signal direction is controlled through PRR0. The signal can be configured as an ESSI signal SRD0 through PCR0. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
STD0	Input/ Output	Input	**Serial Transmit Data**—STD0 is used for transmitting data from the serial transmit shift register. STD0 is an output when data is being transmitted.
PC5	Input or Output		**Port C 5**—The default configuration following reset is GPIO input PC5. When configured as PC5, signal direction is controlled through PRR0. The signal can be configured as an ESSI signal STD0 through PCR0. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

ENHANCED SYNCHRONOUS SERIAL INTERFACE 1

Table 1-12. Enhanced Serial Synchronous Interface 1

Signal Name	Type	State During Reset	Signal Description
SC10 PD0	Input or Output Input or Output	Input	**Serial Control 0**—The function of SC10 is determined by the selection of either synchronous or asynchronous mode. For asynchronous mode, this signal will be used for the receive clock I/O (Schmitt-trigger input). For synchronous mode, this signal is used either for transmitter 1 output or for serial I/O flag 0. **Port D 0**—The default configuration following reset is GPIO input PD0. When configured as PD0, signal direction is controlled through the port directions register (PRR1). The signal can be configured as an ESSI signal SC10 through the port control register (PCR1). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SC11 PD1	Input/ Output Input or Output	Input	**Serial Control 1**—The function of this signal is determined by the selection of either synchronous or asynchronous mode. For asynchronous mode, this signal is the receiver frame sync I/O. For synchronous mode, this signal is used either for Transmitter 2 output or for Serial I/O Flag 1. **Port D 1**—The default configuration following reset is GPIO input PD1. When configured as PD1, signal direction is controlled through PRR1. The signal can be configured as an ESSI signal SC11 through PCR1. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SC12 PD2	Input/ Output Input or Output	Input	**Serial Control Signal 2**—SC12 is used for frame sync I/O. SC12 is the frame sync for both the transmitter and receiver in synchronous mode, and for the transmitter only in asynchronous mode. When configured as an output, this signal is the internally generated frame sync signal. When configured as an input, this signal receives an external frame sync signal for the transmitter (and the receiver in synchronous operation). **Port D 2**—The default configuration following reset is GPIO input PD2. When configured as PD2, signal direction is controlled through PRR1. The signal can be configured as an ESSI signal SC12 through PCR1. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

Table 1-12. Enhanced Serial Synchronous Interface 1 (Continued)

Signal Name	Type	State During Reset	Signal Description
SCK1	Input/ Output	Input	**Serial Clock**—SCK1 is a bidirectional Schmitt-trigger input signal providing the serial bit rate clock for the ESSI. The SCK1 is a clock input or output used by both the transmitter and receiver in synchronous modes, or by the transmitter in asynchronous modes. Although an external serial clock can be independent of and asynchronous to the DSP system clock, it must exceed the minimum clock cycle time of 6T (that is, the system clock frequency must be at least three times the external ESSI clock frequency). The ESSI needs at least three DSP phases inside each half of the serial clock.
PD3	Input or Output		**Port D 3**—The default configuration following reset is GPIO input PD3. When configured as PD3, signal direction is controlled through PRR1. The signal can be configured as an ESSI signal SCK1 through PCR1. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SRD1	Input/ Output	Input	**Serial Receive Data**—SRD1 receives serial data and transfers the data to the ESSI receive shift register. SRD1 is an input when data is being received.
PD4	Input or Output		**Port D 4**—The default configuration following reset is GPIO input PD4. When configured as PD4, signal direction is controlled through PRR1. The signal can be configured as an ESSI signal SRD1 through PCR1. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
STD1	Input/ Output	Input	**Serial Transmit Data**—STD1 is used for transmitting data from the serial transmit shift register. STD1 is an output when data is being transmitted.
PD5	Input or Output		**Port D 5**—The default configuration following reset is GPIO input PD5. When configured as PD5, signal direction is controlled through PRR1. The signal can be configured as an ESSI signal STD1 through PCR1. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

SCI

The SCI provides a full duplex port for serial communication to other DSPs, microprocessors, or peripherals such as modems.

Table 1-13. Serial Communication Interface

Signal Name	Type	State During Reset	Signal Description
RXD PE0	Input Input or Output	Input	**Serial Receive Data**—This input receives byte oriented serial data and transfers it to the SCI receive shift register. **Port E 0**—The default configuration following reset is GPIO input PE0. When configured as PE0, signal direction is controlled through the SCI port directions register (PRR). The signal can be configured as an SCI signal RXD through the SCI port control register (PCR). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
TXD PE1	Output Input or Output	Input	**Serial Transmit Data**—This signal transmits data from SCI transmit data register. **Port E 1**—The default configuration following reset is GPIO input PE1. When configured as PE1, signal direction is controlled through the SCI PRR. The signal can be configured as an SCI signal TXD through the SCI PCR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
SCLK PE2	Input/ Output Input or Output	Input	**Serial Clock**—This is the bidirectional Schmitt-trigger input signal providing the input or output clock used by the transmitter and/or the receiver. **Port E 2**—The default configuration following reset is GPIO input PE2. When configured as PE2, signal direction is controlled through the SCI PRR. The signal can be configured as an SCI signal SCLK through the SCI PCR. Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

TIMERS

Three identical and independent timers are implemented in the DSP56311. Each timer can use internal or external clocking and can either interrupt the DSP56311 after a specified number of events (clocks) or signal an external device after counting a specific number of internal events.

Table 1-14. Triple Timer Signals

Signal Name	Type	State During Reset	Signal Description
TIO0	Input or Output	Input	**Timer 0 Schmitt-Trigger Input/Output**— When Timer 0 functions as an external event counter or in measurement mode, TIO0 is used as input. When Timer 0 functions in watchdog, timer, or pulse modulation mode, TIO0 is used as output. The default mode after reset is GPIO input. This can be changed to output or configured as a timer I/O through the timer 0 control/status register (TCSR0). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
TIO1	Input or Output	Input	**Timer 1 Schmitt-Trigger Input/Output**— When Timer 1 functions as an external event counter or in measurement mode, TIO1 is used as input. When Timer 1 functions in watchdog, timer, or pulse modulation mode, TIO1 is used as output. The default mode after reset is GPIO input. This can be changed to output or configured as a timer I/O through the timer 1 control/status register (TCSR1). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.
TIO2	Input or Output	Input	**Timer 2 Schmitt-Trigger Input/Output**— When Timer 2 functions as an external event counter or in measurement mode, TIO2 is used as input. When Timer 2 functions in watchdog, timer, or pulse modulation mode, TIO2 is used as output. The default mode after reset is GPIO input. This can be changed to output or configured as a timer I/O through the timer 2 control/status register (TCSR2). Note: This signal has a weak keeper to maintain the last state even if all drivers are tri-stated.

JTAG AND OnCE INTERFACE

The DSP56300 family and in particular the DSP56311 support circuit-board test strategies based on the *IEEE 1149.1 Standard Test Access Port and Boundary Scan Architecture*, the industry standard developed under the sponsorship of the Test Technology Committee of IEEE and the JTAG.

The OnCE module provides a means to interface nonintrusively with the DSP56300 core and its peripherals so that you can examine registers, memory, or on-chip peripherals. Functions of the OnCE module are provided through the JTAG TAP signals.

For programming models, see the chapter on debugging support in the *DSP56300 Family Manual*.

Table 1-15. OnCE/JTAG Interface

Signal Name	Type	State During Reset	Signal Description
TCK	Input	Input	**Test Clock**—TCK is a test clock input signal used to synchronize the JTAG test logic.
TDI	Input	Input	**Test Data Input**—TDI is a test data serial input signal used for test instructions and data. TDI is sampled on the rising edge of TCK and has an internal pull-up resistor.
TDO	Output	Tri-stated	**Test Data Output**—TDO is a test data serial output signal used for test instructions and data. TDO is tri-statable and is actively driven in the shift-IR and shift-DR controller states. TDO changes on the falling edge of TCK.
TMS	Input	Input	**Test Mode Select**—TMS is an input signal used to sequence the test controller’s state machine. TMS is sampled on the rising edge of TCK and has an internal pull-up resistor.
$\overline{\text{TRST}}$	Input	Input	**Test Reset**—$\overline{\text{TRST}}$ is an active-low Schmitt-trigger input signal used to asynchronously initialize the test controller. $\overline{\text{TRST}}$ has an internal pull-up resistor. $\overline{\text{TRST}}$ must be asserted after power up.

Table 1-15. OnCE/JTAG Interface (Continued)

Signal Name	Type	State During Reset	Signal Description
$\overline{\text{DE}}$	Input/ Output	Input	**Debug Event**—$\overline{\text{DE}}$ is an open-drain, bidirectional, active-low signal that provides, as an input, a means of entering the debug mode of operation from an external command controller, and, as an output, a means of acknowledging that the chip has entered the debug mode. This signal, when asserted as an input, causes the DSP56300 core to finish the current instruction being executed, save the instruction pipeline information, enter the debug mode, and wait for commands to be entered from the debug serial input line. This signal is asserted as an output for three clock cycles when the chip enters the debug mode as a result of a debug request or as a result of meeting a breakpoint condition. The $\overline{\text{DE}}$ has an internal pull-up resistor. This is not a standard part of the JTAG TAP controller. The signal connects directly to the OnCE module to initiate debug mode directly or to provide a direct external indication that the chip has entered the debug mode. All other interface with the OnCE module must occur through the JTAG port.

SECTION 2

SPECIFICATIONS

INTRODUCTION

The DSP56311 is fabricated in high density CMOS with Transistor-Transistor Logic (TTL) compatible inputs and outputs. The DSP56311 specifications are preliminary and are from design simulations, and may not be fully tested or guaranteed. Finalized specifications will be published after full characterization and device qualifications are complete.

MAXIMUM RATINGS

CAUTION

This device contains circuitry protecting against damage due to high static voltage or electrical fields; however, normal precautions should be taken to avoid exceeding maximum voltage ratings. Reliability is enhanced if unused inputs are tied to an appropriate logic voltage level (for example, either GND or V_{CC}).

Note: In the calculation of timing requirements, adding a maximum value of one specification to a minimum value of another specification does not yield a reasonable sum. A maximum specification is calculated using a worst case variation of process parameter values in one direction. The minimum specification is calculated using the worst case for the same parameters in the opposite direction. Therefore, a “maximum” value for a specification never occurs in the same device that has a “minimum” value for another specification; adding a maximum to a minimum represents a condition that can never exist.

Table 2-1. Maximum Ratings

Note: 1. Rating[1]	Symbol	Value[1, 2]	Unit
Supply Voltage	V_{CC}	-0.1 to 2.0	V
Input/Output Supply Voltage	V_{CCH}	-0.3 to 4	V
All input voltages	V_{IN}	GND - 0.3 to V_{CCH} + 0.3	V
Current drain per pin excluding V_{CC} and GND	I	10	mA
Operating temperature range	T_J	0 to +100	°C
Storage temperature	T_{STG}	-55 to +150	°C

Notes:
1. GND = 0 V, V_{CC} = 1.8 V ± 0.1 V, V_{CCH} = 3.3 V ± 0.3 V, T_J = 0×C to +100×C, CL = 50 pF
2. Absolute maximum ratings are stress ratings only, and functional operation at the maximum is not guaranteed. Stress beyond the maximum rating may affect device reliability or cause permanent damage to the device.
3. Power up sequence: During power up, and throughout the DSP56311 operation, V_{CCH} voltage must always be higher or equal to V_{CC} voltage.

THERMAL CHARACTERISTICS

Table 2-2. Thermal Characteristics

Characteristic	Symbol	PBGA[3] Value	PBGA[4] Value	Unit
Junction-to-ambient thermal resistance	$R_{\theta JA}$ or θ_{JA}	49	TBD	°C/W
Junction-to-case thermal resistance	$R_{\theta JC}$ or θ_{JC}	10	—	°C/W
Thermal characterization parameter	Ψ_{JT}	8	TBD	°C/W

Notes:
1. Junction-to-ambient thermal resistance is based on measurements on a horizontal single-sided printed circuit board per SEMI G38-87 in natural convection.(SEMI is Semiconductor Equipment and Materials International, 805 East Middlefield Rd., Mountain View, CA 94043, (415) 964-5111)
2. Junction-to-case thermal resistance is based on measurements using a cold plate per SEMI G30-88, with the exception that the cold plate temperature is used for the case temperature.
3. These are simulated values; testing is not complete. See note 1 for test board conditions.
4. These are simulated values; testing is not complete. The test board has two, 2-ounce signal layers and two 1-ounce solid ground planes internal to the test board.

DC ELECTRICAL CHARACTERISTICS

Table 2-3. DC Electrical Characteristics[6]

Characteristics	Symbol	Min	Typ	Max	Unit
Supply voltage	V_{CC}	1.7	1.8	1.9	V
Input/Output Supply Voltage	V_{CCH}	3.0	3.3	3.6	V
Input high voltage • All inputs except EXTAL • All inputs except EXTAL • EXTAL[8]	 V_{IH} V_{IHP} V_{IHX}	 2.0 2.0 $0.8 \times V_{CCH}$	 — — —	 $V_{CCH} + 0.4$ $V_{CCH} + 0.4$ V_{CCH}	 V V V
Input low voltage • D(0:23), $\overline{\text{BG}}$, $\overline{\text{BB}}$, $\overline{\text{TA}}$, MOD[1]/$\overline{\text{IRQ}}$[1], $\overline{\text{RESET}}$, PINIT • All JTAG/ESSI/SCI/Timer/HI08 pins • EXTAL[8]	 V_{IL} V_{ILP} V_{ILX}	 –0.3 –0.3 –0.3	 — — —	 0.8 0.8 $0.2 \times V_{CCH}$	 V V V
Input leakage current	I_{IN}	–10	—	10	µA
High impedance (off-state) input current (@ 2.4 V / 0.4 V)	I_{TSI}	–10	—	10	µA
Output high voltage • TTL (I_{OH} = –0.4 mA)[5,7] • CMOS (I_{OH} = –10 µA)[5]	V_{OH}	 2.4 $V_{CC} - 0.01$	 — —	 — —	 V V
Output low voltage • TTL (I_{OL} = 3.0 mA, open-drain pins I_{OL} = 6.7 mA)[5,7] • CMOS (I_{OL} = 10 µA)[5]	V_{OL}	 — —	 — —	 0.4 0.01	 V V
Internal supply current[2]: • In Normal mode • In Wait mode[3] • In Stop mode[4]	 I_{CCI} I_{CCW} I_{CCS}	 — — —	 150 MHz: 110 150 MHz: 7. 5 100	 — — —	 mA mA µA
PLL supply current		—	1	2.5	mA
Input capacitance[5]	C_{IN}	—	—	10	pF

Table 2-3. DC Electrical Characteristics[6] (Continued)

Characteristics	Symbol	Min	Typ	Max	Unit
Notes: 1. Refers to MODA/$\overline{\text{IRQA}}$, MODB/$\overline{\text{IRQB}}$, MODC/$\overline{\text{IRQC}}$, and MODD/$\overline{\text{IRQD}}$ pins 2. **Power Consumption Considerations** on page 4-3 provides a formula to compute the estimated current requirements in Normal mode. In order to obtain these results, all inputs must be terminated (that is, not allowed to float). Measurements are based on synthetic intensive DSP benchmarks (see **Appendix A**). The power consumption numbers in this specification are 90 percent of the measured results of this benchmark. This reflects typical DSP applications. Typical internal supply current is measured with V_{CCH} = 3.3 V, V_{CC} = 1.8 V at T_J = 100°C. 3. In order to obtain these results, all inputs must be terminated (that is, not allowed to float). PLL and XTAL signals are disabled during Stop state. 4. DC current in stop model is based on preliminary estimation, and will be evaluated based on measurements. In order to obtain these results, all inputs, which are not disconnected at Stop mode, must be terminated (that is, not allowed to float). 5. Periodically sampled and not 100 percent tested 6. V_{CCH} = 3.3 V ± 0.3 V, V_{CC} = 1.8 V ± 0.1 V; T_J = 0°C to +100 °C, C_L = 50 pF 7. This characteristic does not apply to XTAL and PCAP. 8. Driving EXTAL to the low V_{IHX} or the high V_{ILX} value may cause additional power consumption (DC current). To minimize power consumption, the minimum V_{IHX} should be no lower than $0.9 \times V_{CCH}$ and the maximum V_{ILX} should be no higher than $0.1 \times V_{CCH}$.					

AC ELECTRICAL CHARACTERISTICS

The timing waveforms shown in the AC electrical characteristics section are tested with a V_{IL} maximum of 0.3 V and a V_{IH} minimum of 2.4 V for all pins except EXTAL, which is tested using the input levels shown in **Note 6** of the previous table. AC timing specifications, which are referenced to a device input signal, are measured in production with respect to the 50 percent point of the respective input signal's transition. DSP56311 output levels are measured with the production test machine V_{OL} and V_{OH} reference levels set at 0.4 V and 2.4 V, respectively.

Note: Although the minimum value for the frequency of EXTAL is 0 MHz, the device AC test conditions are 15 MHz and rated speed.

INTERNAL CLOCKS

Table 2-4. Internal Clocks, ~~CLKOUT~~

Characteristics	Symbol	Expression[1, 2]		
		Min	**Typ**	**Max**
Internal operation frequency ~~and CLKOUT~~ with PLL enabled	f	—	(Ef × MF)/ (PDF × DF)	—
Internal operation frequency ~~and CLKOUT~~ with PLL disabled	f	—	Ef/2	—
Internal clock ~~and CLKOUT~~ high period • With PLL disabled • With PLL enabled and MF ≤ 4 • With PLL enabled and MF > 4	T_H	— 0.49 × ET_C × PDF × DF/MF 0.47 × ET_C × PDF × DF/MF	ET_C — —	— 0.51 × ET_C × PDF × DF/MF 0.53 × ET_C × PDF × DF/MF
Internal clock ~~and CLKOUT~~ low period • With PLL disabled • With PLL enabled and MF ≤ 4 • With PLL enabled and MF > 4	T_L	— 0.49 × ET_C × PDF × DF/MF 0.47 × ET_C × PDF × DF/MF	ET_C — —	— 0.51 × ET_C × PDF × DF/MF 0.53 × ET_C × PDF × DF/MF
Internal clock ~~and CLKOUT~~ cycle time with PLL enabled	T_C	—	ET_C × PDF × DF/MF	—
Internal clock ~~and CLKOUT~~ cycle time with PLL disabled	T_C	—	2 × ET_C	—
Instruction cycle time	I_{CYC}	—	T_C	—

Notes:
1. DF = Division Factor
   Ef = External frequency
   ET_C = External clock cycle
   MF = Multiplication Factor
   PDF = Predivision Factor
   T_C = internal clock cycle
2. See the **PLL and Clock Generation** section in the *DSP56300 Family Manual* for a detailed discussion of the PLL.

EXTERNAL CLOCK OPERATION

The DSP56311 system clock can be derived from the on–chip crystal oscillator, as shown in **Figure 1** on the cover page, or it can be externally supplied. An externally supplied square wave voltage source should connect to EXTAL (see **Figure 2-1**), leaving XTAL physically not connected to the board or socket.

Fundamental Frequency
Fork Crystal Oscillator

Suggested Component Values:

f_{OSC} = 32.768 kHz
R1 = 3.9 MΩ ± 10%
C = 22 pF ± 20%
R2 = 200 kΩ ± 10%

Calculations were done for a 32.768 kHz crystal with the following parameters:
a load capacitance (C_L) of 12.5 pF,
a shunt capacitance (C_0) of 1.8 pF,
a series resistance of 40 kΩ, and
a drive level of 1 µW.

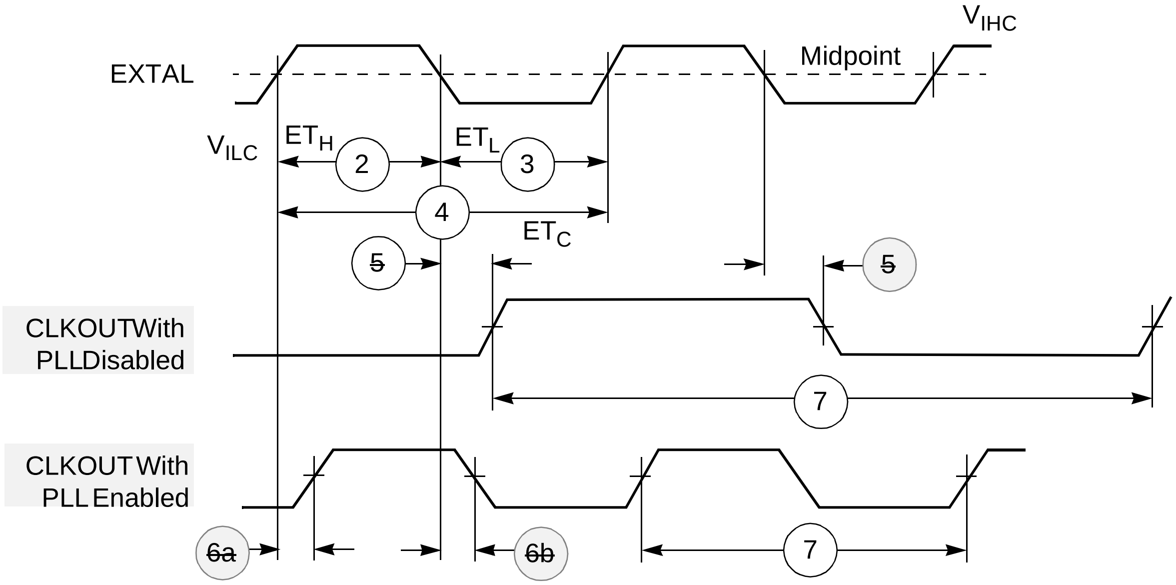
Fundamental Frequency
Crystal Oscillator

Suggested Component Values:

f_{OSC} = 4 MHz	f_{OSC} = 20 MHz
R = 680 kΩ ± 10%	R = 680 kΩ ± 10%
C = 56 pF ± 20%	C = 22 pF ± 20%

Calculations were done for a 4/20 MHz crystal with the following parameters:
a $C_L$of 30/20 pF,
a C_0 of 7/6 pF,
a series resistance of 100/20 Ω, and
a drive level of 2 mW.

Figure 2-1. Crystal Oscillator Circuits



Note: The midpoint is 0.5 (V_{IHC} + V_{ILC}).

Figure 2-2. External Clock Timing

Table 2-5. Clock Operation

No.	Characteristics	Symbol	150 MHz	
			Min	Max
1	Frequency of EXTAL (EXTAL Pin Frequency) The rise and fall time of this external clock should be 3 ns maximum.	Ef	0	150.0
2	EXTAL input high[1, 2] • With PLL disabled (46.7%–53.3% duty cycle[6]) • With PLL enabled (42.5%–57.5% duty cycle[6])	ET_H	3.11 ns 2.83 ns	∞ 78.5 µs
3	EXTAL input low[1, 2] • With PLL disabled (46.7%–53.3% duty cycle[6]) • With PLL enabled (42.5%–57.5% duty cycle[6])	ET_L	3.11 ns 2.83 ns	∞ 78.5 µs
4	EXTAL cycle time[2] • With PLL disabled • With PLL enabled	ET_C	6.67 ns 6.67 ns	∞ 136.5 µs
~~5~~	~~CLKOUT change from EXTAL fall with PLL disabled~~		~~4.3 ns~~	~~11.0 ns~~

Table 2-5. Clock Operation (Continued)

No.	Characteristics	Symbol	150 MHz	
			Min	Max
~~6~~	~~CLKOUT rising edge from EXTAL rising edge with PLL enabled (MF = 1, PDF = 1, Ef > 15 MHz)~~[3,5]		~~0.0 ns~~	~~1.8 ns~~
	~~CLKOUT falling edge from EXTAL rising edge with PLL enabled (MF = 2 or 4, PDF = 1, Ef > 15 MHz)~~[3,5]		~~0.0 ns~~	~~1.8 ns~~
	~~CLKOUT falling edge from EXTAL falling edge with PLL enabled (MF $\leq$ 4, PDF $\neq$ 1, Ef / PDF > 15 MHz)~~[3,5]		~~0.0 ns~~	~~1.8 ns~~
7	Instruction cycle time = $I_{CYC} = T_C$[4] (see **Figure 2-4**) (46.7%–53.3% duty cycle)	I_{CYC}		
	• With PLL disabled		13.33 ns	∞
	• With PLL enabled		6.7 ns	4266.7 ns

Notes:
1. Measured at 50 percent of the input transition
2. The maximum value for PLL enabled is given for minimum V_{CO} and maximum MF.
3. Periodically sampled and not 100 percent tested
4. The maximum value for PLL enabled is given for minimum V_{CO} and maximum DF.
5. ~~The skew is not guaranteed for any other MF value.~~
6. The indicated duty cycle is for the specified maximum frequency for which a part is rated. The minimum clock high or low time required for correction operation, however, remains the same at lower operating frequencies; therefore, at a lower clock frequency, the signal symmetry can vary from the specified duty cycle as long as the minimum high time and low time requirements are met.

PHASE LOCK LOOP (PLL) CHARACTERISTICS

Table 2-6. PLL Characteristics

Characteristics	150 MHz		Unit
	Min	Max	
V_{CO} frequency when PLL enabled ($MF \times E_f \times 2/PDF$)	30	300	MHz
PLL external capacitor (PCAP pin to V_{CCP}) (C_{PCAP}[1)])			
• @ MF $\leq$ 4	$(580 \times MF) - 100$	$(780 \times MF) - 140$	pF
• @ MF > 4	$830 \times MF$	$1470 \times MF$	pF

Note: C_{PCAP} is the value of the PLL capacitor (connected between the PCAP pin and V_{CCP}). The recommended value in pF for C_{PCAP} can be computed from one of the following equations:

$(680 \times MF) - 120$, for MF $\leq$ 4, or

$1100 \times MF$, for MF > 4.

RESET, STOP, MODE SELECT, AND INTERRUPT TIMING

Table 2-7. Reset, Stop, Mode Select, and Interrupt Timing[6]

No.	Characteristics	Expression	150 MHz		Unit
			Min	Max	
8	Delay from $\overline{\text{RESET}}$ assertion to all pins at reset value[3]	—	—	26.0	ns
9	Required $\overline{\text{RESET}}$ duration[4] • Power on, external clock generator, PLL disabled • Power on, external clock generator, PLL enabled • Power on, internal oscillator • During STOP, XTAL disabled (PCTL Bit 16 = 0) • During STOP, XTAL enabled (PCTL Bit 16 = 1) • During normal operation	 $50 \times ET_C$ $1000 \times ET_C$ $75000 \times ET_C$ $75000 \times ET_C$ $2.5 \times T_C$ $2.5 \times T_C$	 334.0 6.67 0.50 0.50 16.7 16.7	 — — — — — —	 ns µs ms ms ns ns
10	Delay from asynchronous $\overline{\text{RESET}}$ deassertion to first external address output (internal reset deassertion)[5] • Minimum • Maximum	 $3.25 \times T_C + 2.0$ $20.25\ T_C + 7.50$	 23.7 —	 — 142.6	 ns ns
~~11~~	~~Synchronous reset setup time from $\overline{\text{RESET}}$ deassertion to CLKOUT Transition 1~~ ~~• Minimum~~ ~~• Maximum~~	 ~~T_C~~	 ~~5.9~~ ~~—~~	 ~~—~~ ~~10.0~~	 ~~ns~~ ~~ns~~
~~12~~	~~Synchronous reset deasserted, delay time from the CLKOUT Transition 1 to the first external address output~~ ~~• Minimum~~ ~~• Maximum~~	 ~~$3.25 \times T_C + 1.0$~~ ~~$20.25\ T_C + 5.0$~~	 ~~33.5~~ ~~—~~	 ~~—~~ ~~207.5~~	 ~~ns~~ ~~ns~~
13	Mode select setup time		30.0	—	ns
14	Mode select hold time		0.0	—	ns
15	Minimum edge-triggered interrupt request assertion width		6.6	—	ns
16	Minimum edge-triggered interrupt request deassertion width		6.6	—	ns
17	Delay from $\overline{\text{IRQA}}$, $\overline{\text{IRQB}}$, $\overline{\text{IRQC}}$, $\overline{\text{IRQD}}$, $\overline{\text{NMI}}$ assertion to external memory access address out valid • Caused by first interrupt instruction fetch • Caused by first interrupt instruction execution	 $4.25 \times T_C + 2.0$ $7.25 \times T_C + 2.0$	 30.4 51.0	 — —	 ns ns
18	Delay from $\overline{\text{IRQA}}$, $\overline{\text{IRQB}}$, $\overline{\text{IRQC}}$, $\overline{\text{IRQD}}$, $\overline{\text{NMI}}$ assertion to general-purpose transfer output valid caused by first interrupt instruction execution	$10 \times T_C + 5.0$	72.0	—	ns
19	Delay from address output valid caused by first interrupt instruction execute to interrupt request deassertion for level sensitive fast interrupts[1, 7, 8]	$(WS + 3.75) \times T_C - 11.0$	—	—	ns
20	Delay from $\overline{\text{RD}}$ assertion to interrupt request deassertion for level sensitive fast interrupts[1, 7, 8]	$(WS + 3.25) \times T_C - 11.0$	—	—	ns

Table 2-7. Reset, Stop, Mode Select, and Interrupt Timing[6] (Continued)

No.	Characteristics	Expression	150 MHz		Unit
			Min	Max	
21	Delay from $\overline{\mathrm{WR}}$ assertion to interrupt request deassertion for level sensitive fast interrupts[1, 7, 8]				
	• DRAM for all WS	(WS + 3.5) × T_C – 11.0	—	—	ns
	• SRAM WS = 1	(WS + 3.5) × T_C – 11.0	—	—	ns
	• SRAM WS = 2, 3	(WS + 3) × T_C – 11.0	—	—	ns
	• SRAM WS ≥ 4	(WS + 2.5) × T_C – 11.0	—	—	ns
~~22~~	~~Synchronous interrupt setup time from $\overline{\mathrm{IRQA}}$, $\overline{\mathrm{IRQB}}$, $\overline{\mathrm{IRQC}}$, $\overline{\mathrm{IRQD}}$, $\overline{\mathrm{NMI}}$ assertion to the CLKOUT Transition 2~~		~~5.9~~	~~T_C~~	~~ns~~
~~23~~	~~Synchronous interrupt delay time from the CLKOUT Transition 2 to the first external address output valid caused by the first instruction fetch after coming out of Wait Processing state~~				
	~~• Minimum~~	~~9.25 × T_C + 1.0~~	~~93.5~~	~~—~~	~~ns~~
	~~• Maximum~~	~~24.75 × T_C + 5.0~~	~~—~~	~~252.5~~	~~ns~~
24	Duration for $\overline{\mathrm{IRQA}}$ assertion to recover from Stop state		5.9	—	ns
25	Delay from $\overline{\mathrm{IRQA}}$ assertion to fetch of first instruction (when exiting Stop)[2, 3]				
	• PLL is not active during Stop (PCTL Bit 17 = 0) and Stop delay is enabled (OMR Bit 6 = 0)	PLC × ET_C × PDF + (128 K – PLC/2) × T_C	1.3	9.1	ms
	• PLL is not active during Stop (PCTL Bit 17 = 0) and Stop delay is not enabled (OMR Bit 6 = 1)	PLC × ET_C × PDF + (23.75 ± 0.5) × T_C	232.5 ns	12.3 ms	
	• PLL is active during Stop (PCTL Bit 17 = 1) (Implies No Stop Delay)	(8.25 ± 0.5) × T_C	51.7	58.3	ns
26	Duration of level sensitive $\overline{\mathrm{IRQA}}$ assertion to ensure interrupt service (when exiting Stop)[2, 3]				
	• PLL is not active during Stop (PCTL Bit 17 = 0) and Stop delay is enabled (OMR Bit 6 = 0)	PLC × ET_C × PDF + (128K – PLC/2) × T_C	13.6	—	ms
	• PLL is not active during Stop (PCTL Bit 17 = 0) and Stop delay is not enabled (OMR Bit 6 = 1)	PLC × ET_C × PDF + (20.5 ± 0.5) × T_C	12.3	—	ms
	• PLL is active during Stop (PCTL Bit 17 = 1) (implies no Stop delay)	5.5 × T_C	55.0	—	ns
27	Interrupt Requests Rate				
	• HI08, ESSI, SCI, Timer	$12T_C$	—	80.0	ns
	• DMA	$8T_C$	—	53.3	ns
	• $\overline{\mathrm{IRQ}}$, $\overline{\mathrm{NMI}}$ (edge trigger)	$8T_C$	—	53.3	ns
	• $\overline{\mathrm{IRQ}}$, $\overline{\mathrm{NMI}}$ (level trigger)	$12T_C$	—	80.0	ns
28	DMA Requests Rate				
	• Data read from HI08, ESSI, SCI	$6T_C$	—	40.0	ns
	• Data write to HI08, ESSI, SCI	$7T_C$	—	46.7	ns
	• Timer	$2T_C$	—	13.3	ns
	• $\overline{\mathrm{IRQ}}$, $\overline{\mathrm{NMI}}$ (edge trigger)	$3T_C$	—	20.0	ns

Table 2-7. Reset, Stop, Mode Select, and Interrupt Timing[6] (Continued)

No.	Characteristics	Expression	150 MHz		Unit
			Min	Max	
29	Delay from $\overline{\mathrm{IRQA}}$, $\overline{\mathrm{IRQB}}$, $\overline{\mathrm{IRQC}}$, $\overline{\mathrm{IRQD}}$, $\overline{\mathrm{NMI}}$ assertion to external memory (DMA source) access address out valid	$4.25 \times T_C + 2.0$	30.3	—	ns

Notes:
1. When fast interrupts are used and $\overline{\mathrm{IRQA}}$, $\overline{\mathrm{IRQB}}$, $\overline{\mathrm{IRQC}}$, and $\overline{\mathrm{IRQD}}$ are defined as level-sensitive, timings 19 through 21 apply to prevent multiple interrupt service. To avoid these timing restrictions, the deasserted Edge-triggered mode is recommended when fast interrupts are used. Long interrupts are recommended for Level-sensitive mode.
2. This timing depends on several settings:

   For PLL disable, using internal oscillator (PLL Control Register (PCTL) Bit 16 = 0) and oscillator disabled during Stop (PCTL Bit 17 = 0), a stabilization delay is required to assure that the oscillator is stable before programs are executed. In that case, resetting the Stop delay (OMR Bit 6 = 0) provides the proper delay. While OMR Bit 6 = 1 can be set, it is not recommended, and these specifications do not guarantee timings for that case.

   For PLL disable, using internal oscillator (PCTL Bit 16 = 0) and oscillator enabled during Stop (PCTL Bit 17=1), no stabilization delay is required and recovery is minimal (OMR Bit 6 setting is ignored).

   For PLL disable, using external clock (PCTL Bit 16 = 1), no stabilization delay is required and recovery time is defined by the PCTL Bit 17 and OMR Bit 6 settings.

   For PLL enable, if PCTL Bit 17 is 0, the PLL is shutdown during Stop. Recovering from Stop requires the PLL to get locked. The PLL lock procedure duration, PLL Lock Cycles (PLC), may be in the range of 0 to 1000 cycles. This procedure occurs in parallel with the stop delay counter, and stop recovery ends when the last of these two events occurs. The stop delay counter completes count or PLL lock procedure completion.

   PLC value for PLL disable is 0.

   The maximum value for ET_C is 4096 (maximum MF) divided by the desired internal frequency (that is, for 66 MHz it is 4096/66 MHz = 62 μs). During the stabilization period, T_C, $T_{H,}$ and T_L is not constant, and their width may vary, so timing may vary as well.
3. Periodically sampled and not 100 percent tested.
4. For an external clock generator, $\overline{\mathrm{RESET}}$ duration is measured while $\overline{\mathrm{RESET}}$ is asserted, V_{CC} is valid, and the EXTAL input is active and valid.

   For internal oscillator, $\overline{\mathrm{RESET}}$ duration is measured while $\overline{\mathrm{RESET}}$ is asserted and V_{CC} is valid. The specified timing reflects the crystal oscillator stabilization time after power-up. This number is affected both by the specifications of the crystal and other components connected to the oscillator and reflects worst case conditions.

   When the V_{CC} is valid, but the other “required $\overline{\mathrm{RESET}}$ duration” conditions (as specified above) have not been yet met, the device circuitry is in an uninitialized state that can result in significant power consumption and heat-up. Designs should minimize this state to the shortest possible duration.
5. If PLL does not lose lock.
6. $V_{CCH} = 3.3\ \mathrm{V} \pm 0.3\ \mathrm{V}$, $V_{CC} = 1.8\ \mathrm{V} \pm 0.1\ \mathrm{V}$; $T_J = 0°\mathrm{C}$ to $+100°\mathrm{C}$, $C_L = 50\ \mathrm{pF}$
7. WS = number of wait states (measured in clock cycles, number of T_C).
8. Use expression to compute maximum value.

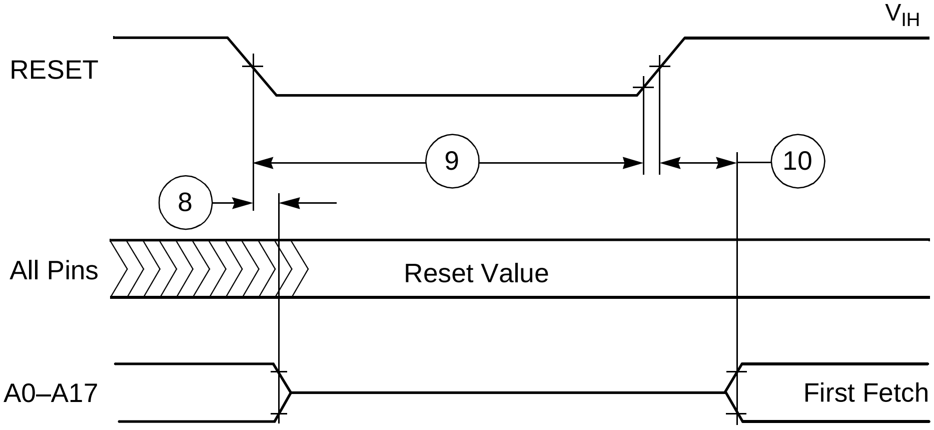


Figure 2-3. Reset Timing

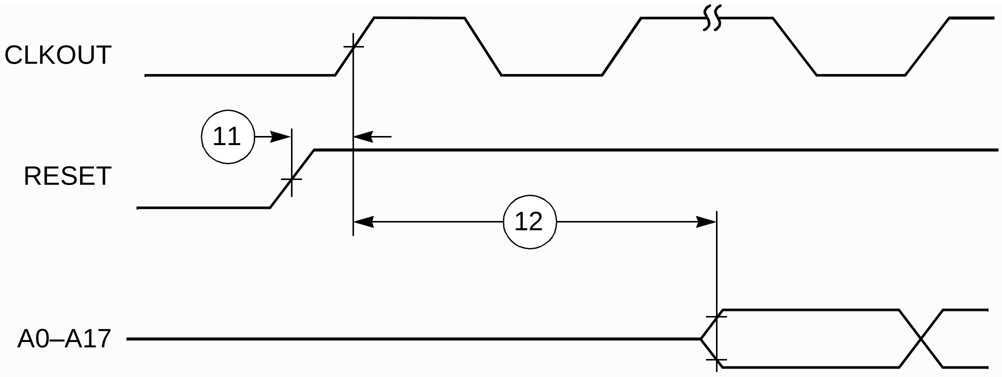
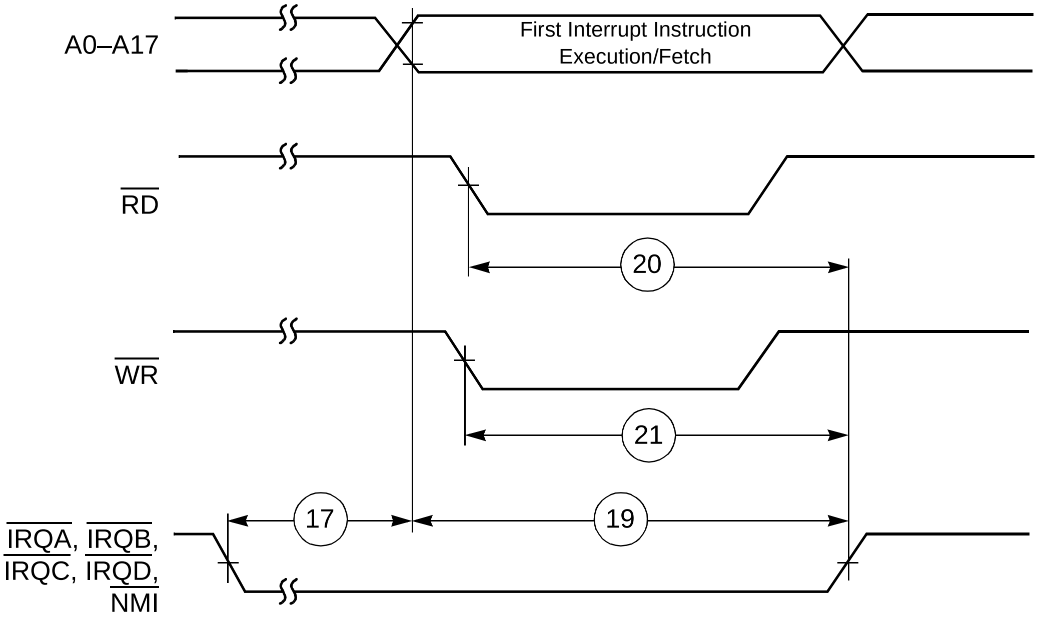
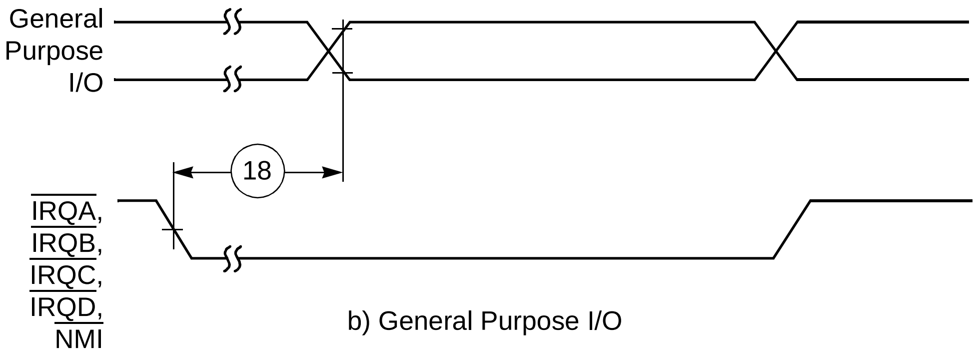


Figure 2-4. ~~Synchronous Reset Timing~~



a) First Interrupt Instruction Execution



b) General Purpose I/O

Figure 2-5. External Fast Interrupt Timing

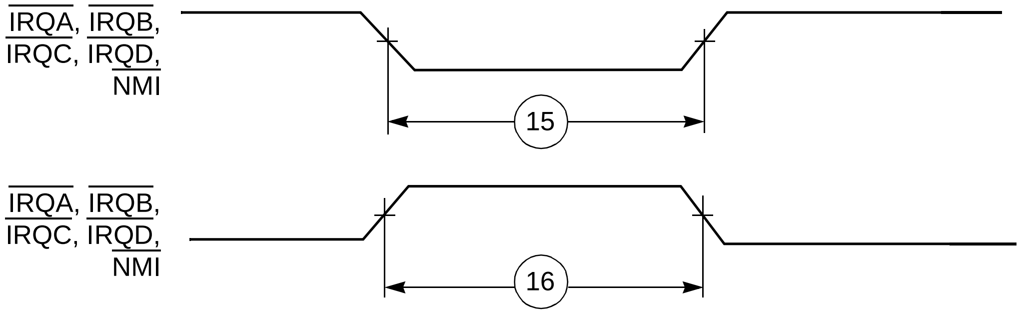


Figure 2-6. External Interrupt Timing (Negative Edge-Triggered)

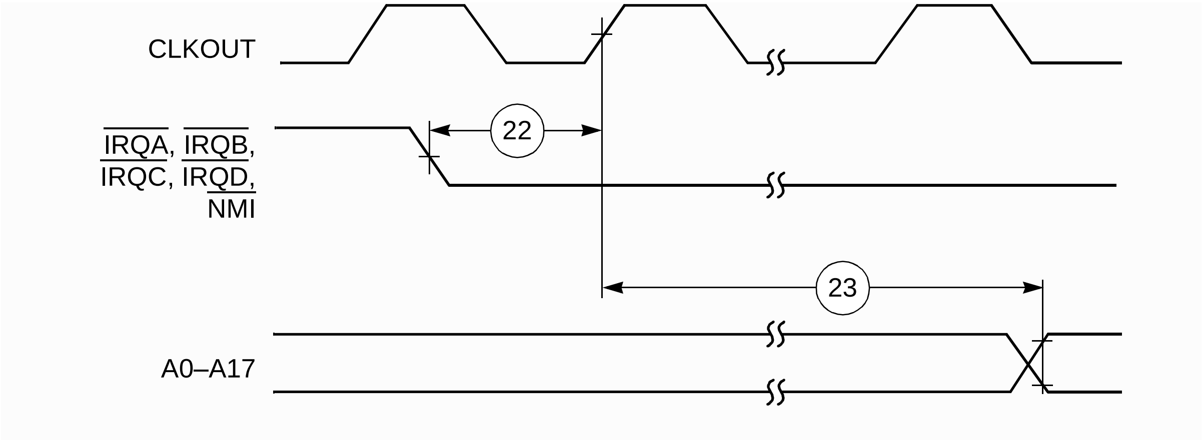


Figure 2-7. Synchronous Interrupt from Wait State Timing

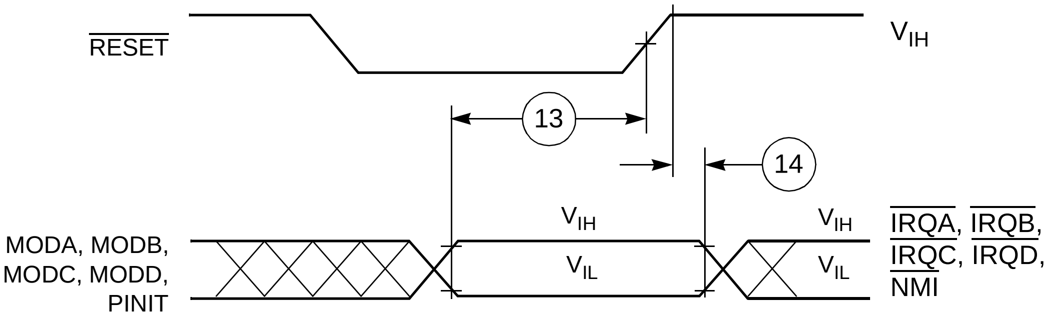


Figure 2-8. Operating Mode Select Timing

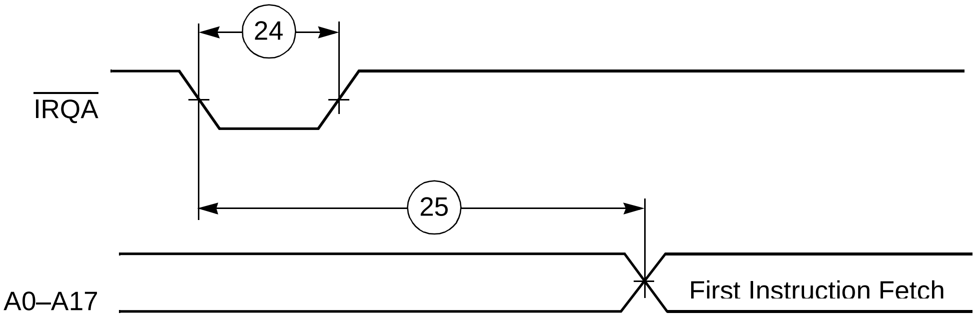


Figure 2-9. Recovery from Stop State Using $\overline{\text{IRQA}}$

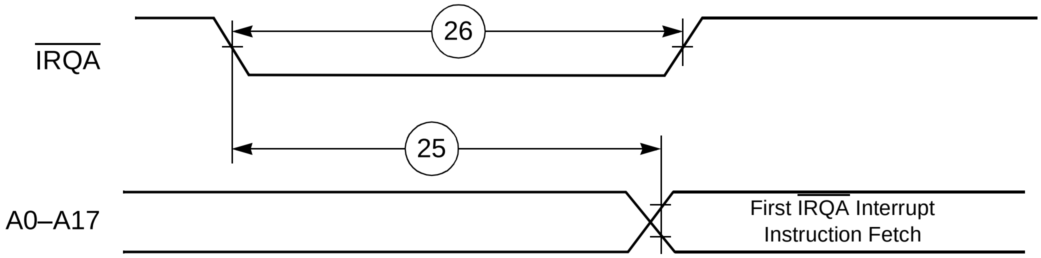


Figure 2-10. Recovery from Stop State Using IRQA Interrupt Service

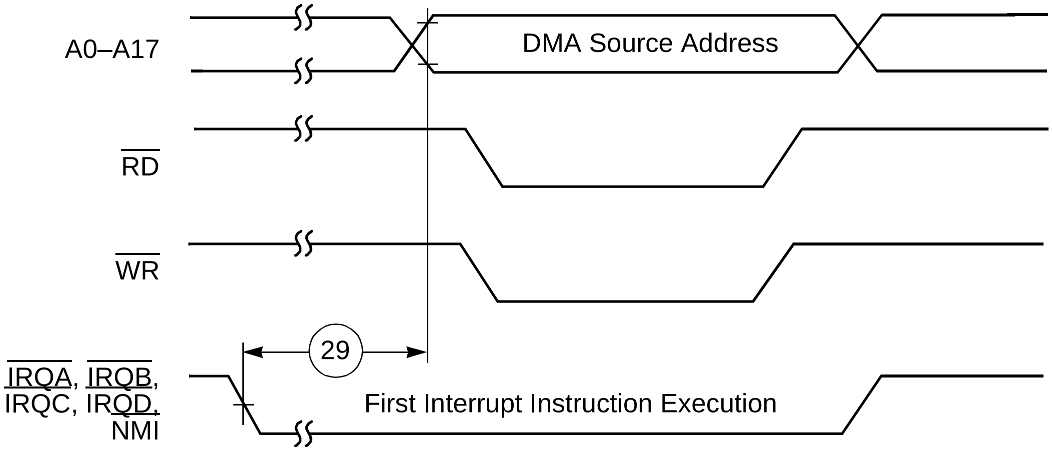


Figure 2-11. External Memory Access (DMA Source) Timing

EXTERNAL MEMORY EXPANSION PORT (PORT A)

SRAM Timing

Table 2-8. SRAM Read and Write Accesses[3]

No.	Characteristics	Symbol	Expression[1]	150 MHz		Unit
				Min	Max	
100	Address valid and AA assertion pulse width[2]	t_{RC}, t_{WC}	(WS + 2) × T_C – 4.0 [2 ≤ WS ≤ 7] (WS + 3) × T_C – 4.0 [WS ≥ 8]	22.7 69.3	 —	ns ns
101	Address and AA valid to $\overline{WR}$ assertion	t_{AS}	0.75 × T_C – 3.0 [2 ≤ WS ≤ 3] 1.25 × T_C – 3.0 [WS ≥ 4]	2.0 5.3	— —	ns ns
102	$\overline{WR}$ assertion pulse width	t_{WP}	WS × T_C – 4.0 [2 ≤ WS ≤ 3] (WS – 0.5) × T_C – 4.0 [WS ≥ 4]	9.3 19.3	— —	ns ns
103	$\overline{WR}$ deassertion to address not valid	t_{WR}	1.25 × T_C – 4.0 [2 ≤ WS ≤ 7] 2.25 × T_C – 4.0 [WS ≥ 8]	4.3 11.0	— —	ns ns
104	Address and AA valid to input data valid	t_{AA}, t_{AC}	(WS + 0.75) × T_C – 6.5 [WS ≥ 2]	—	11.8	ns
105	$\overline{RD}$ assertion to input data valid	t_{OE}	(WS + 0.25) × T_C – 6.5 [WS ≥ 2]	—	8.5	ns
106	$\overline{RD}$ deassertion to data not valid (data hold time)	t_{OHZ}		0.0	—	ns
107	Address valid to $\overline{WR}$ deassertion[2]	t_{AW}	(WS + 0.75) × T_C – 4.0 [WS ≥ 2]	14.3	—	ns
108	Data valid to $\overline{WR}$ deassertion (data setup time)	t_{DS} (t_{DW})	(WS – 0.25) × T_C – 5.4 [WS ≥ 2]	6.3	—	ns
109	Data hold time from $\overline{WR}$ deassertion	t_{DH}	1.25 × T_C – 4.0 [2 ≤ WS ≤ 7] 2.25 × T_C – 4.0 [WS ≥ 8]	4.3 11.0	— —	ns ns
~~110~~	~~$\overline{WR}$ assertion to data active~~	— — —	~~0.75 × T_C – 4.0~~ ~~[WS = 1]~~ ~~0.25 × T_C – 4.0~~ ~~[2 ≤ WS ≤ 3]~~ ~~–0.25 × T_C – 4.0~~ ~~[WS ≥ 4]~~	~~1~~ ~~–2.3~~ ~~–5.7~~	— — —	~~ns~~ ~~ns~~ ~~ns~~

Table 2-8. SRAM Read and Write Accesses[3] (Continued)

No.	Characteristics	Symbol	Expression[1]	150 MHz		Unit
				Min	Max	
~~111~~	~~$\overline{\text{WR}}$ deassertion to data high impedance~~	—		—	—	~~ns~~
~~112~~	~~Previous $\overline{\text{RD}}$ deassertion to data active (write)~~	—		—	—	~~ns~~
113	$\overline{\text{RD}}$ deassertion time	— —	$1.75 \times T_C - 4.0$ $[2 \leq WS \leq 7]$ $2.75 \times T_C - 4.0$ $[WS \geq 8]$	7.6 14.3	— —	ns ns
114	$\overline{\text{WR}}$ deassertion time	— — —	$2 \times T_C - 4.0$ $[2 \leq WS \leq 3]$ $2.5 \times T_C - 4.0$ $[4 \leq WS \leq 7]$ $3.5 \times T_C - 4.0$ $[WS \geq 8]$	9.3 12.6 19.3	— — —	ns ns ns
~~115~~	~~Address valid to $\overline{\text{RD}}$ assertion~~	—	~~$0.5 \times T_C - 2.0$~~	~~1.3~~	—	~~ns~~
116	$\overline{\text{RD}}$ assertion pulse width	—	$(WS + 0.25) \times T_C - 4.0$	11.0	—	ns
117	$\overline{\text{RD}}$ deassertion to address not valid	— —	$1.25 \times T_C - 4.0$ $[2 \leq WS \leq 7]$ $2.25 \times T_C - 4.0$ $[WS \geq 8]$	4.3 11.0	— —	ns ns
118	$\overline{\text{TA}}$ setup before $\overline{\text{RD}}$ or $\overline{\text{WR}}$ deassertion[4]	—	$0.25 \times T_C + 1.5$	3.2	—	ns
119	$\overline{\text{TA}}$ hold after $\overline{\text{RD}}$ or $\overline{\text{WR}}$ deassertion	—		0	—	ns

Notes:
1. WS is the number of wait states specified in the BCR. The value is given for the minimum for a given category. (for example, for a category of $[2 \leq WS \leq 7]$ timing is specified for 2 wait states.) Two wait states is the minimum otherwise.
2. Timings 100 and107 are guaranteed by design, not tested.
3. All timings for 150 MHz are measured from $0.5 \cdot V_{CCH}$ to $0.5 \cdot V_{CCH}$.
4. For $\overline{\text{TA}}$ negation: timing 118 is relative to the deassertion edge of $\overline{\text{RD}}$ or $\overline{\text{WR}}$ were $\overline{\text{TA}}$ to remain active.

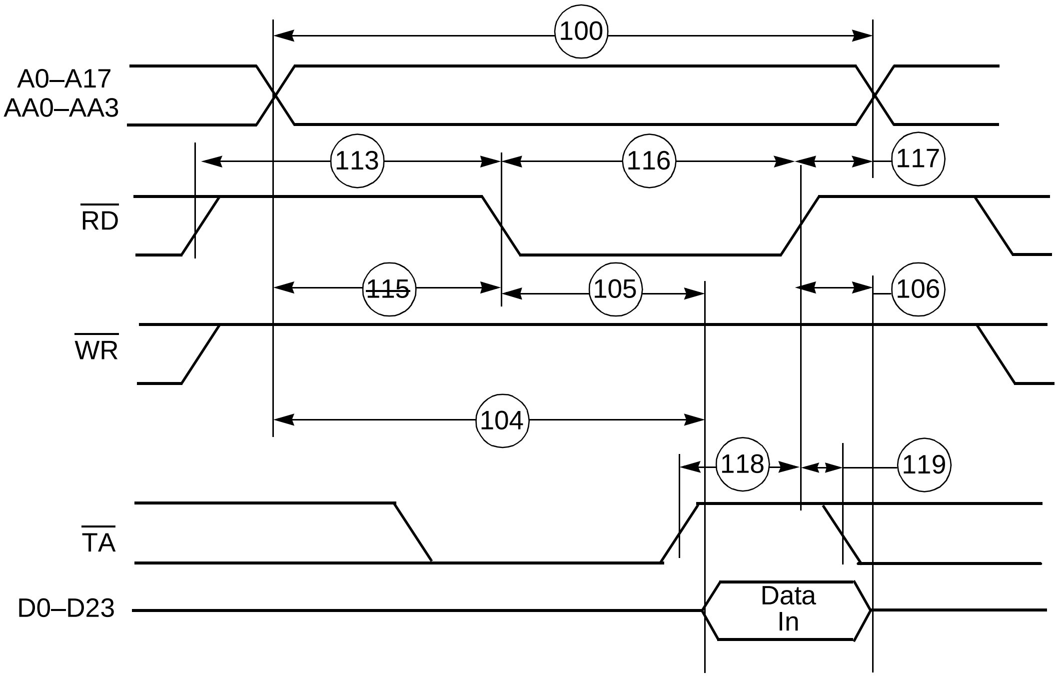


Figure 2-12. SRAM Read Access

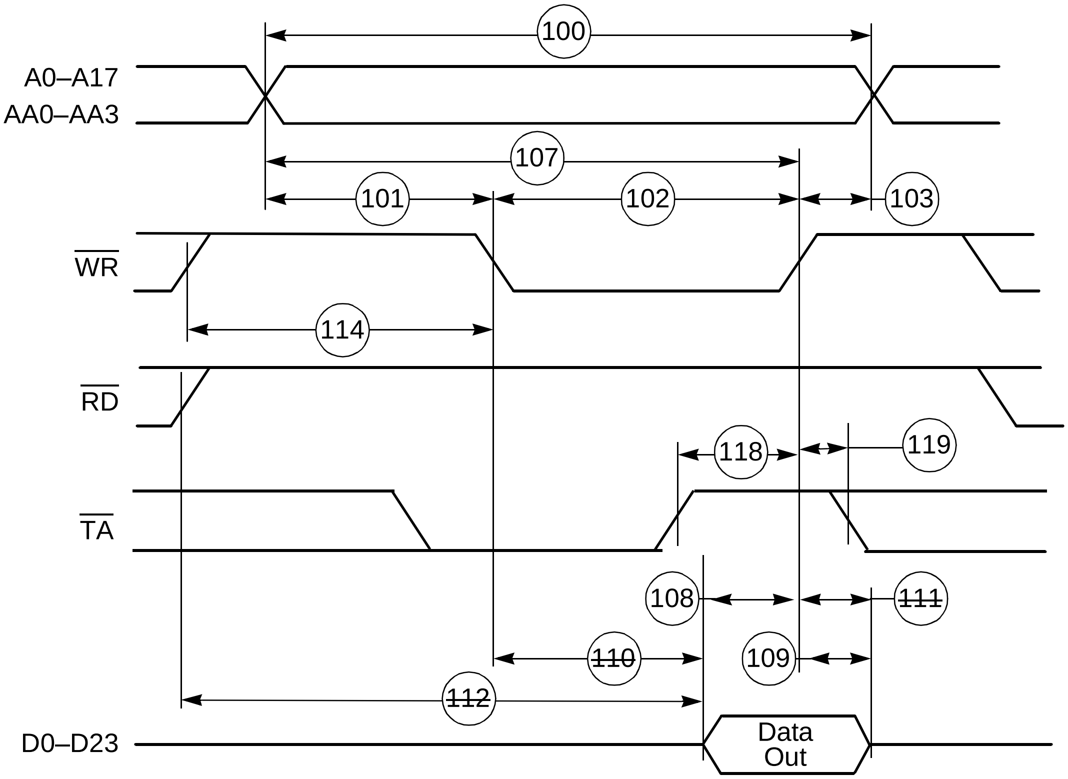


Figure 2-13. SRAM Write Access

Arbitration Timings

Table 2-9. Asynchronous Bus Arbitration timing

No.	Characteristics	Expression	150 MHz		Unit
			Min	Max	
250	$\overline{\text{BB}}$ assertion window from $\overline{\text{BG}}$ input negation.	2.5 × Tc + 5	—	22	ns
251	Delay from $\overline{\text{BB}}$ assertion to $\overline{\text{BG}}$ assertion	2 × Tc + 5	18.3	—	ns
Comments: 1. Bit 13 in the OMR register must be set to enable Asynchronous Arbitration mode. 2. At 150 MHz, Asynchronous Arbitration mode is recommended. 3. To guarantee timings 250 and 251, it is recommended to assert $\overline{\text{BG}}$ inputs to different DSP56300 devices (on the same bus) in a non-overlap manner, as shown in **Figure 2-14**, where BG1 is the $\overline{\text{BG}}$ signal for one DSP56300 device while BG2 is the BG signal for a second DSP56300 device.					

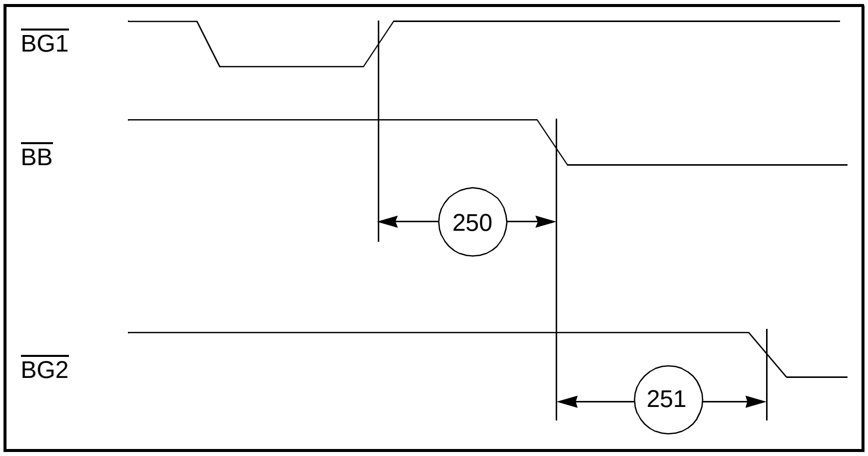


Figure 2-14. Asynchronous Bus Arbitration Timing

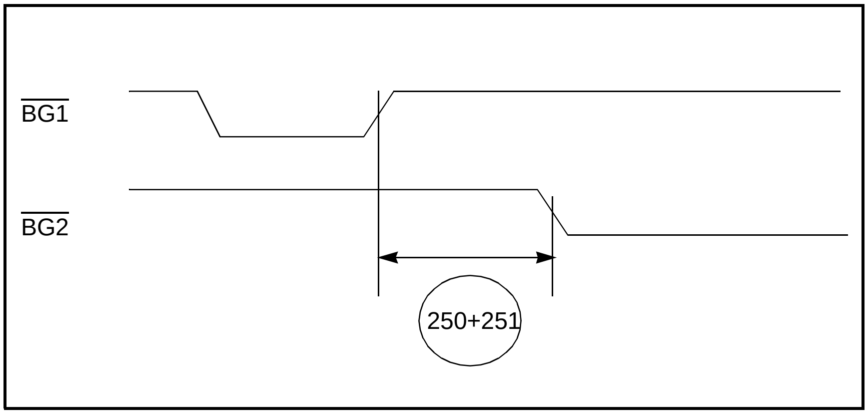


Figure 2-15. Asynchronous Bus Arbitration Timing

The asynchronous bus arbitration is enabled by internal synchronization circuits on $\overline{\mathrm{BG}}$, and $\overline{\mathrm{BB}}$ inputs. These synchronization circuits add delay from the external signal until it is exposed to internal logic. As a result of this delay, a DSP56300 part may assume mastership and assert $\overline{\mathrm{BB}}$, for some time after $\overline{\mathrm{BG}}$ is negated. This is the reason for timing 250.

Once $\overline{\mathrm{BB}}$ is asserted, there is a synchronization delay from $\overline{\mathrm{BB}}$ assertion to the time this assertion is exposed to other DSP56300 components that are potential masters on the same bus. If $\overline{\mathrm{BG}}$ input is asserted before that time, and $\overline{\mathrm{BG}}$ is asserted and $\overline{\mathrm{BB}}$ is negated, another DSP56300 component may assume mastership at the same time. Therefore, some non-overlap period between one $\overline{\mathrm{BG}}$ input active to another $\overline{\mathrm{BG}}$ input active is required. Timing 251 ensures that overlaps are avoided.

HOST INTERFACE TIMING

Table 2-10. Host Interface Timing[1, 2, 12]

No.	Characteristic[10]	Expression	150 MHz		Unit
			Min	Max	
317	Read data strobe assertion width[5] $\overline{\mathrm{HACK}}$ assertion width	$T_C + 9.0$	15.7	—	ns
318	Read data strobe deassertion width[5] $\overline{\mathrm{HACK}}$ deassertion width		9.9	—	ns
319	Read data strobe deassertion width[5] after “Last Data Register” reads[8,11], or between two consecutive CVR, ICR, or ISR reads[3] $\overline{\mathrm{HACK}}$ deassertion width after “Last Data Register” reads[8,11]	$2.5 \times T_C + 6.6$	23.3	—	ns
320	Write data strobe assertion width[6]		13.2	—	ns

Table 2-10. Host Interface Timing[1, 2, 12] (Continued)

No.	Characteristic[10]	Expression	150 MHz		Unit
			Min	Max	
321	Write data strobe deassertion width[8] $\overline{\text{HACK}}$ write deassertion width • after HCTR, HCVR and “Last Data Register” writes • after TXH:TXM writes (with HBE=0) after TXL:TXM writes (with HBE=1)	2.5 × T_C + 6.6	23.3 16.5	— —	ns ns
322	$\overline{\text{HAS}}$ assertion width		9.9	—	ns
323	$\overline{\text{HAS}}$ deassertion to data strobe assertion[4]		0.0	—	ns
324	Host data input setup time before write data strobe deassertion[6]		9.9	—	ns
325	Host data input hold time after write data strobe deassertion[6]		3.3	—	ns
326	Read data strobe assertion to output data active from high impedance[5] $\overline{\text{HACK}}$ assertion to output data active from high impedance		3.3	—	ns
327	Read data strobe assertion to output data valid[5] $\overline{\text{HACK}}$ assertion to output data valid		—	23.5	ns
328	Read data strobe deassertion to output data high impedance[5] $\overline{\text{HACK}}$ deassertion to output data high impedance		—	9.9	ns
329	Output data hold time after read data strobe deassertion[5] Output data hold time after $\overline{\text{HACK}}$ deassertion		4.1	—	ns
330	$\overline{\text{HCS}}$ assertion to read data strobe deassertion[5]	T_C + 9.9	12.7	—	ns
331	$\overline{\text{HCS}}$ assertion to write data strobe deassertion[6]		9.9	—	ns
332	$\overline{\text{HCS}}$ assertion to output data valid		—	16.5	ns
333	$\overline{\text{HCS}}$ hold time after data strobe deassertion[4]		0.0	—	ns
334	Address (AD7–AD0) setup time before $\overline{\text{HAS}}$ deassertion (HMUX=1)		4.7	—	ns
335	Address (AD7–AD0) hold time after $\overline{\text{HAS}}$ deassertion (HMUX=1)		3.3	—	ns
336	A10–A8 (HMUX=1), A2–A0 (HMUX=0), HR/$\overline{\text{W}}$ setup time before data strobe assertion[4] • Read • Write		0 4.7	— —	ns ns
337	A10–A8 (HMUX=1), A2–A0 (HMUX=0), HR/$\overline{\text{W}}$ hold time after data strobe deassertion[4]		3.3	—	ns
338	Delay from read data strobe deassertion to host request assertion for “Last Data Register” read[5, 7, 8]	2 × T_C + 20.6	34.0	—	ns

Table 2-10. Host Interface Timing[1, 2, 12] (Continued)

No.	Characteristic[10]	Expression	150 MHz		Unit
			Min	Max	
339	Delay from write data strobe deassertion to host request assertion for “Last Data Register” write[6, 7, 8]	$1.5 \times T_C + 16.5$	26.5	—	ns
340	Delay from data strobe assertion to host request deassertion for “Last Data Register” read or write (HROD=0)[4, 7, 8]		—	20.24	ns
341	Delay from data strobe assertion to host request deassertion for “Last Data Register” read or write (HROD=1, open drain host request)[4, 7, 8, 9]		—	300.0	ns

Notes:
1. See the Programmer’s Model section in the chapter on the HI08 in the *DSP56311 User’s Manual*.
2. In the timing diagrams below, the controls pins are drawn as active low. The pin polarity is programmable.
3. This timing is applicable only if two consecutive reads from one of these registers are executed.
4. The data strobe is Host Read (HRD) or Host Write (HWR) in the Dual Data Strobe mode and Host Data Strobe (HDS) in the Single Data Strobe mode.
5. The read data strobe is HRD in the Dual Data Strobe mode and HDS in the Single Data Strobe mode.
6. The write data strobe is HWR in the Dual Data Strobe mode and HDS in the Single Data Strobe mode.
7. The host request is HREQ in the Single Host Request mode and HRRQ and HTRQ in the Double Host Request mode.
8. The “Last Data Register” is the register at address $7, which is the last location to be read or written in data transfers. This is RXL/TXL in the Little Endian mode (HBE = 0), or RXH/TXH in the Big Endian mode (HBE = 1).
9. In this calculation, the host request signal is pulled up by a 4.7 kΩ resistor in the Open-drain mode.
10. $V_{CCH} = 3.3\ V \pm 0.3\ V$, $V_{CC} = 1.8\ V \pm 0.1\ V$; $T_J = 0°C$ to $+100\ °C$, $C_L = 50\ pF$
11. This timing is applicable only if a read from the “Last Data Register” is followed by a read from the RXL, RXM, or RXH registers without first polling RXDF or HREQ bits, or waiting for the assertion of the HREQ signal.
12. After the external host writes a new value to the ICR, the HI08 will be ready for operation after three DSP clock cycles ($3 \times Tc$).

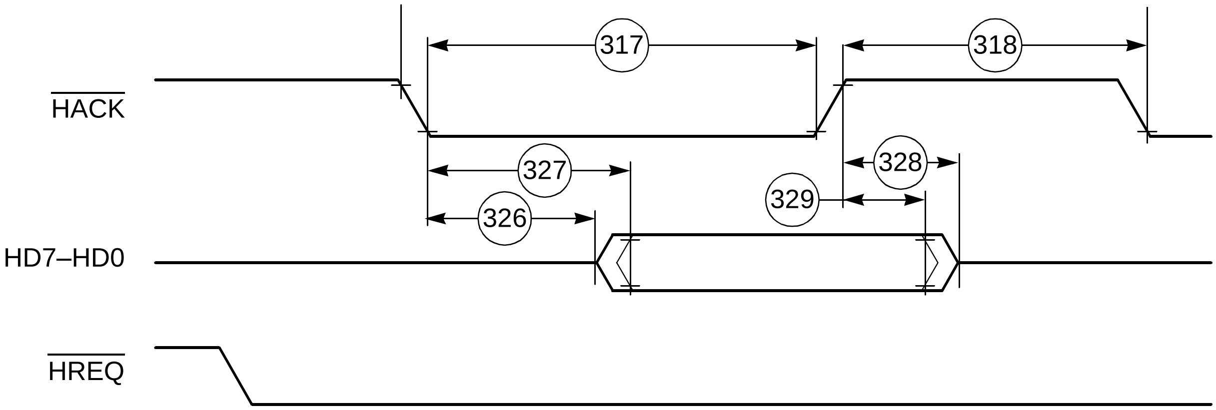


Figure 2-16. Host Interrupt Vector Register (IVR) Read Timing Diagram

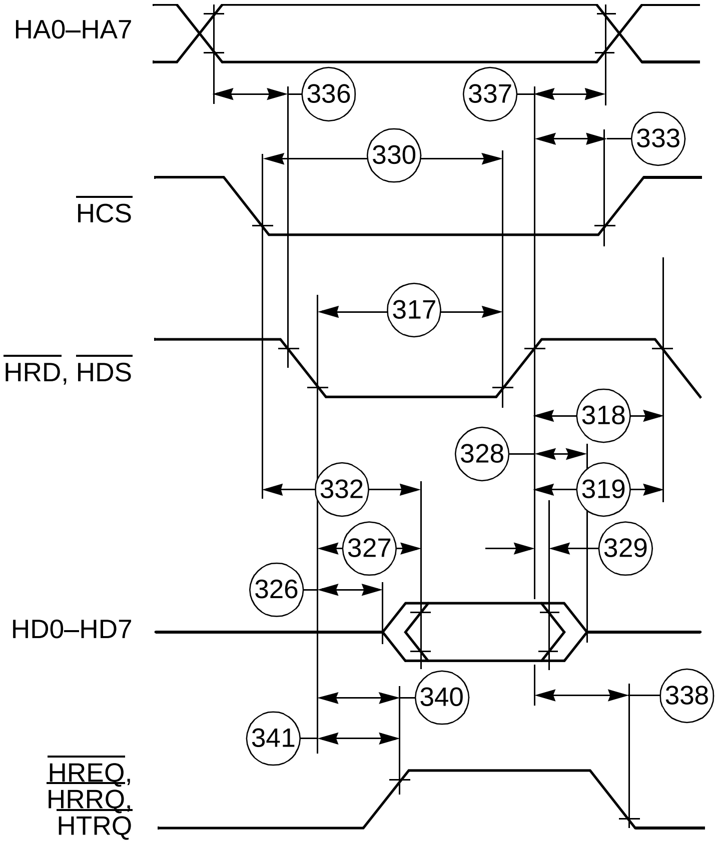


Figure 2-17. Read Timing Diagram, Non-Multiplexed Bus

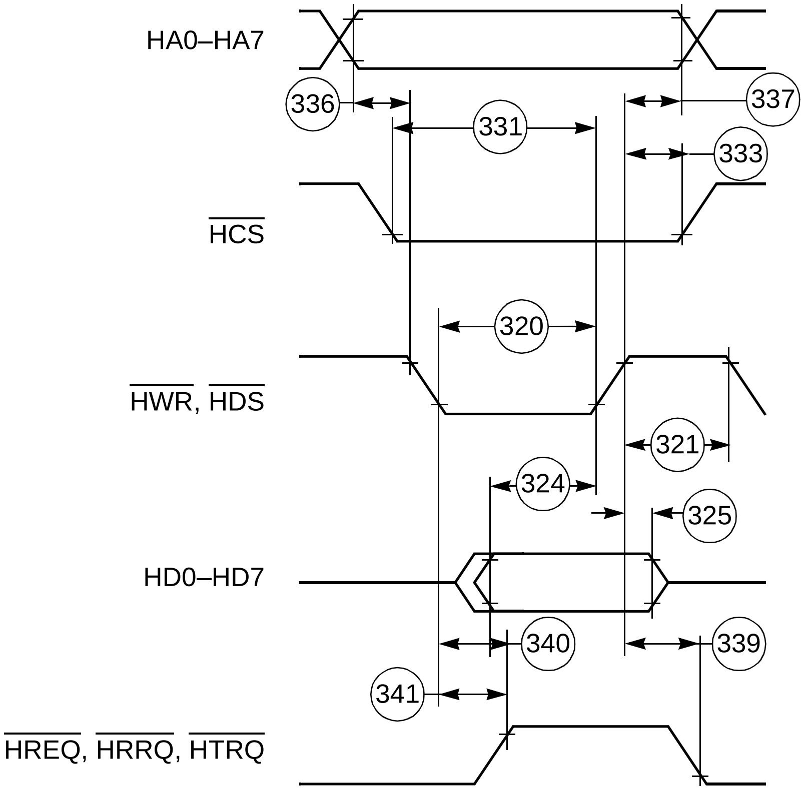


Figure 2-18. Write Timing Diagram, Non-Multiplexed Bus

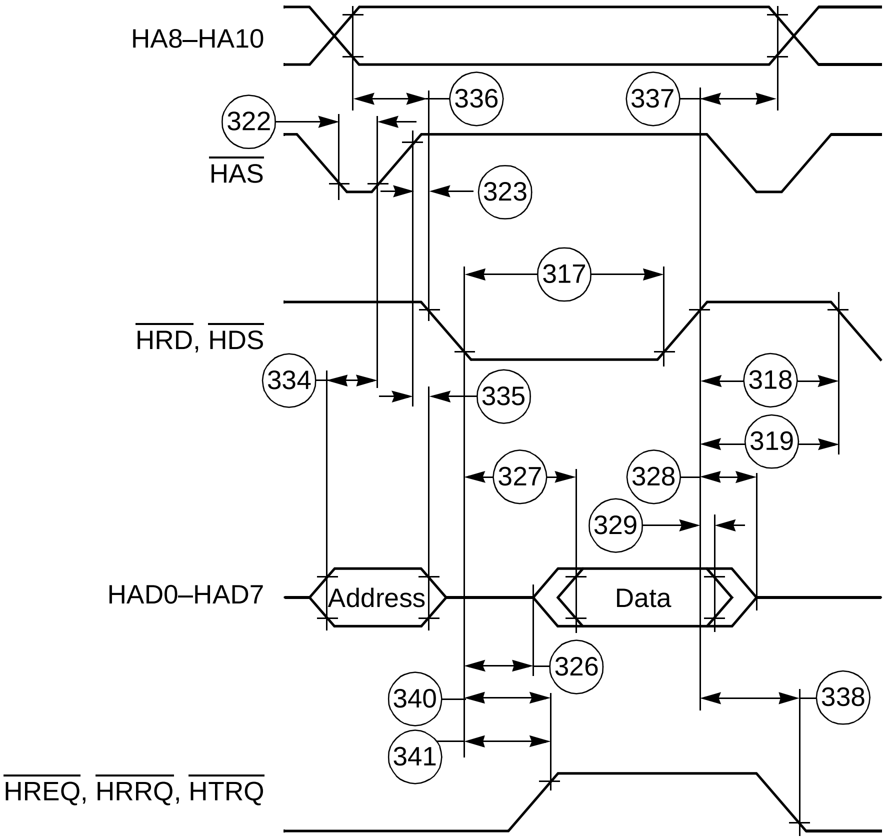


Figure 2-19. Read Timing Diagram, Multiplexed Bus

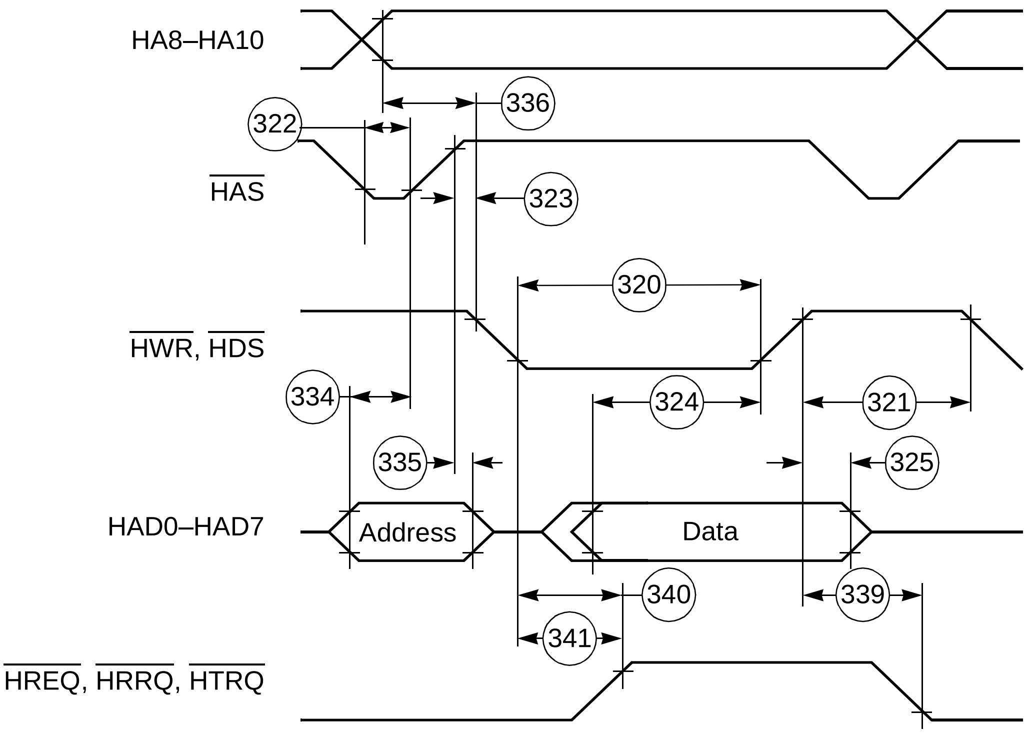


Figure 2-20. Write Timing Diagram, Multiplexed Bus

SCI TIMING

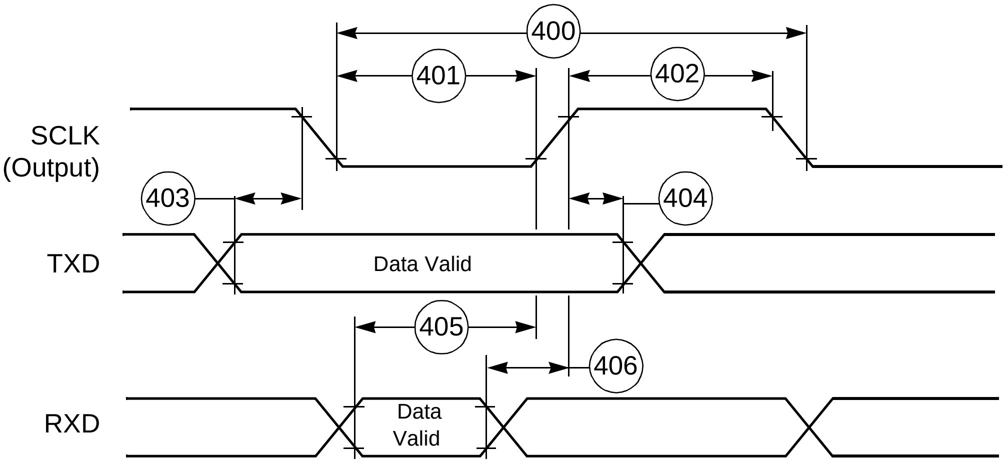
Table 2-11. SCI Timing

No.	Characteristics[1]	Symbol	Expression	150 MHz		Unit
				Min	Max	
400	Synchronous clock cycle	t_{SCC}[2]	$8 \times T_C$	53.3	—	ns
401	Clock low period		$t_{SCC}/2 - 10.0$	30.0	—	ns
402	Clock high period		$t_{SCC}/2 - 10.0$	30.0	—	ns
403	Output data setup to clock falling edge (internal clock)		$t_{SCC}/4 + 0.5 \times T_C - 17.0$	8.0	—	ns
404	Output data hold after clock rising edge (internal clock)		$t_{SCC}/4 - 0.5 \times T_C$	15.0	—	ns
405	Input data setup time before clock rising edge (internal clock)		$t_{SCC}/4 + 0.5 \times T_C + 25.0$	50.0	—	ns

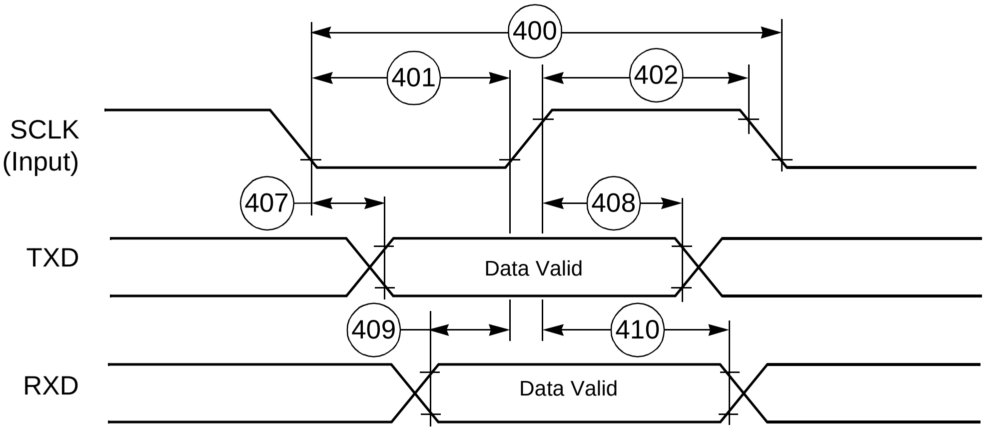
Table 2-11. SCI Timing (Continued)

No.	Characteristics[1]	Symbol	Expression	150 MHz		Unit
				Min	Max	
406	Input data not valid before clock rising edge (internal clock)		$t_{SCC}/4 + 0.5 \times T_C - 5.5$	—	19.5	ns
407	Clock falling edge to output data valid (external clock)			—	32.0	ns
408	Output data hold after clock rising edge (external clock)		$T_C + 8.0$	14.7	—	ns
409	Input data setup time before clock rising edge (external clock)			0.0	—	ns
410	Input data hold time after clock rising edge (external clock)			9.0	—	ns
411	Asynchronous clock cycle	t_{ACC}[3]	$64 \times T_C$	427.0	—	ns
412	Clock low period		$t_{ACC}/2 - 10.0$	310.0	—	ns
413	Clock high period		$t_{ACC}/2 - 10.0$	310.0	—	ns
414	Output data setup to clock rising edge (internal clock)		$t_{ACC}/2 - 30.0$	290.0	—	ns
415	Output data hold after clock rising edge (internal clock)		$t_{ACC}/2 - 30.0$	290.0	—	ns

Notes:
1. $V_{CCH} = 3.3$ V ± 0.3 V, $V_{CC} = 1.8$ V ± 0.1 V; $T_J = 0°C$ to +100 °C, $C_L = 50$ pF
2. t_{SCC} = synchronous clock cycle time (For internal clock, t_{SCC} is determined by the SCI clock control register and T_C.)
3. t_{ACC} = asynchronous clock cycle time; value given for 1X Clock mode (For internal clock, t_{ACC} is determined by the SCI clock control register and T_C.)



a) Internal Clock



b) External Clock

Figure 2-21. SCI Synchronous Mode Timing

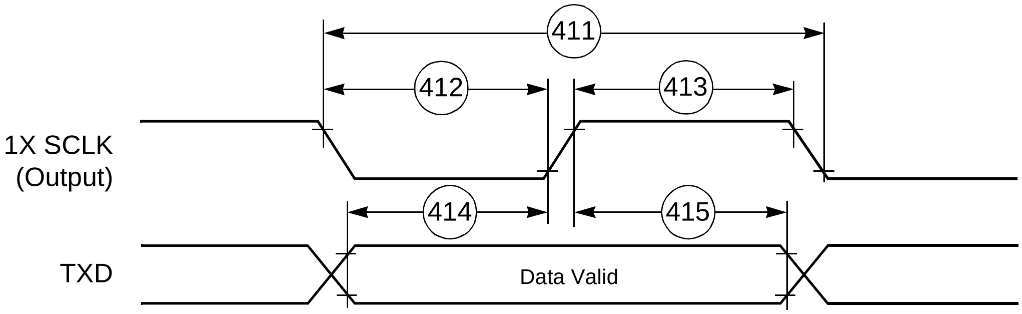


Figure 2-22. SCI Asynchronous Mode Timing

ESSI0/ESSI1 TIMING

Table 2-12. ESSI Timings

No.	Characteristics[4, 6, 7]	Symbol	Expression	150 MHz		Cond-ition[5]	Unit
				Min	Max		
430	Clock cycle[1]	t_{SSICC}	$8 \times T_C$ $6 \times T_C$	53.4 40.0	— —	i ck x ck	ns
431	Clock high period • For internal clock • For external clock		$4 \times T_C - 10.0$ $3 \times T_C$	16.7 20.0	— —		ns ns
432	Clock low period • For internal clock • For external clock		$4 \times T_C - 10.0$ $3 \times T_C$	16.7 20.0	— —		ns ns
433	RXC rising edge to FSR out (bl) high			— —	37.0 22.0	x ck i ck a	ns
434	RXC rising edge to FSR out (bl) low			— —	37.0 22.0	x ck i ck a	ns
435	RXC rising edge to FSR out (wr) high[2]			— —	39.0 24.0	x ck i ck a	ns
436	RXC rising edge to FSR out (wr) low[2]			— —	39.0 24.0	x ck i ck a	ns
437	RXC rising edge to FSR out (wl) high			— —	36.0 21.0	x ck i ck a	ns
438	RXC rising edge to FSR out (wl) low			— —	37.0 22.0	x ck i ck a	ns
439	Data in setup time before RXC (SCK in Synchronous mode) falling edge			0.0 19.0	— —	x ck i ck	ns
440	Data in hold time after RXC falling edge			5.0 3.0	— —	x ck i ck	ns
441	FSR input (bl, wr) high before RXC falling edge[2]			1.0 6.0	— —	x ck i ck a	ns
442	FSR input (wl) high before RXC falling edge			1.0 6.0	— —	x ck i ck a	ns
443	FSR input hold time after RXC falling edge			3.0 0.0	— —	x ck i ck a	ns
444	Flags input setup before RXC falling edge			0.0 19.0	— —	x ck i ck s	ns
445	Flags input hold time after RXC falling edge			6.0 0.0	— —	x ck i ck s	ns
446	TXC rising edge to FST out (bl) high			— —	29.0 15.0	x ck i ck	ns

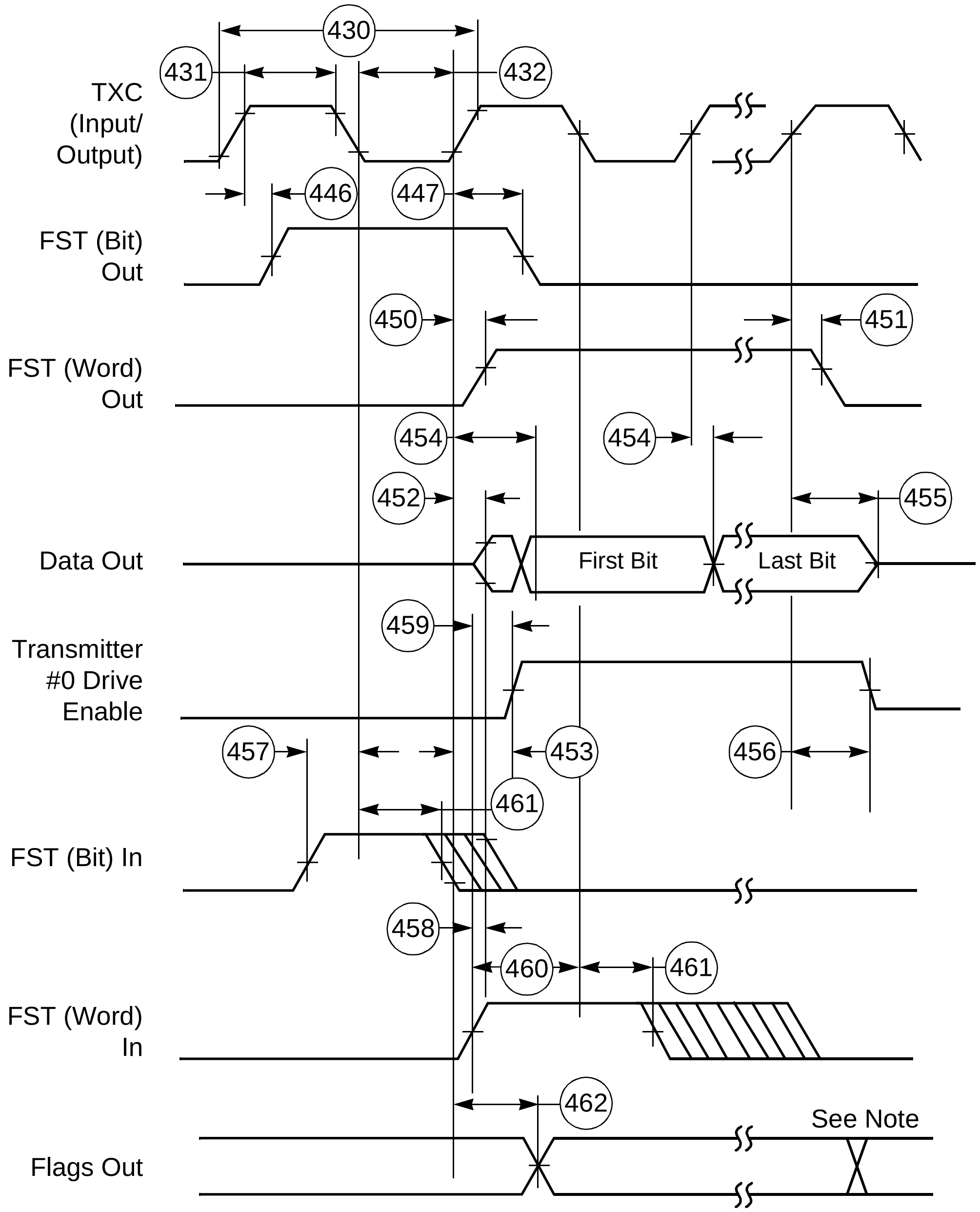
Table 2-12. ESSI Timings (Continued)

No.	Characteristics[4, 6, 7]	Symbol	Expression	150 MHz		Cond-ition[5]	Unit
				Min	Max		
447	TXC rising edge to FST out (bl) low			— —	31.0 17.0	x ck i ck	ns
448	TXC rising edge to FST out (wr) high[2]			— —	31.0 17.0	x ck i ck	ns
449	TXC rising edge to FST out (wr) low[2]			— —	33.0 19.0	x ck i ck	ns
450	TXC rising edge to FST out (wl) high			— —	30.0 16.0	x ck i ck	ns
451	TXC rising edge to FST out (wl) low			— —	31.0 17.0	x ck i ck	ns
452	TXC rising edge to data out enable from high impedance			— —	31.0 17.0	x ck i ck	ns
453	TXC rising edge to Transmitter #0 drive enable assertion			— —	34.0 20.0	x ck i ck	ns
454	TXC rising edge to data out valid		$35 + 0.5 \times T_C$	— —	38.4 21.0	x ck i ck	ns
455	TXC rising edge to data out high impedance[3]			— —	31.0 16.0	x ck i ck	ns
456	TXC rising edge to Transmitter #0 drive enable deassertion[3]			— —	34.0 20.0	x ck i ck	ns
457	FST input (bl, wr) setup time before TXC falling edge[2]			2.0 21.0	— —	x ck i ck	ns
458	FST input (wl) to data out enable from high impedance			—	27.0	—	ns
459	FST input (wl) to Transmitter #0 drive enable assertion			—	31.0	—	ns
460	FST input (wl) setup time before TXC falling edge			2.0 21.0	— —	x ck i ck	ns
461	FST input hold time after TXC falling edge			4.0 0.0	— —	x ck i ck	ns
462	Flag output valid after TXC rising edge			— —	32.0 18.0	x ck i ck	ns

Table 2-12. ESSI Timings (Continued)

No.	Characteristics[4, 6, 7]	Symbol	Expression	150 MHz		Cond-ition[5]	Unit
				Min	Max		

Notes:
1. For the internal clock, the external clock cycle is defined by Icyc and the ESSI control register.
2. The word-relative frame sync signal waveform relative to the clock operates in the same manner as the bit-length frame sync signal waveform, but spreads from one serial clock before first bit clock (same as Bit Length Frame Sync signal), until the one before last bit clock of the first word in frame.
3. Periodically sampled and not 100 percent tested
4. V_{CCH} = 3.3 V ± 0.3 V, V_{CC} = 1.8 V ± 0.1 V; T_J = 0°C to +100 °C, C_L = 50 pF
5. TXC (SCK Pin) = Transmit Clock
   RXC (SC0 or SCK Pin) = Receive Clock
   FST (SC2 Pin) = Transmit Frame Sync
   FSR (SC1 or SC2 Pin) Receive Frame Sync
6. i ck = Internal Clock
   x ck = External Clock
   i ck a = Internal Clock, Asynchronous Mode
   (Asynchronous implies that TXC and RXC are two different clocks)
   i ck s = Internal Clock, Synchronous Mode
   (Synchronous implies that TXC and RXC are the same clock)
7. bl = bit length
   wl = word length
   wr = word length relative



Note: In Network mode, output flag transitions can occur at the start of each time slot within the frame. In Normal mode, the output flag state is asserted for the entire frame period.

Figure 2-23. ESSI Transmitter Timing

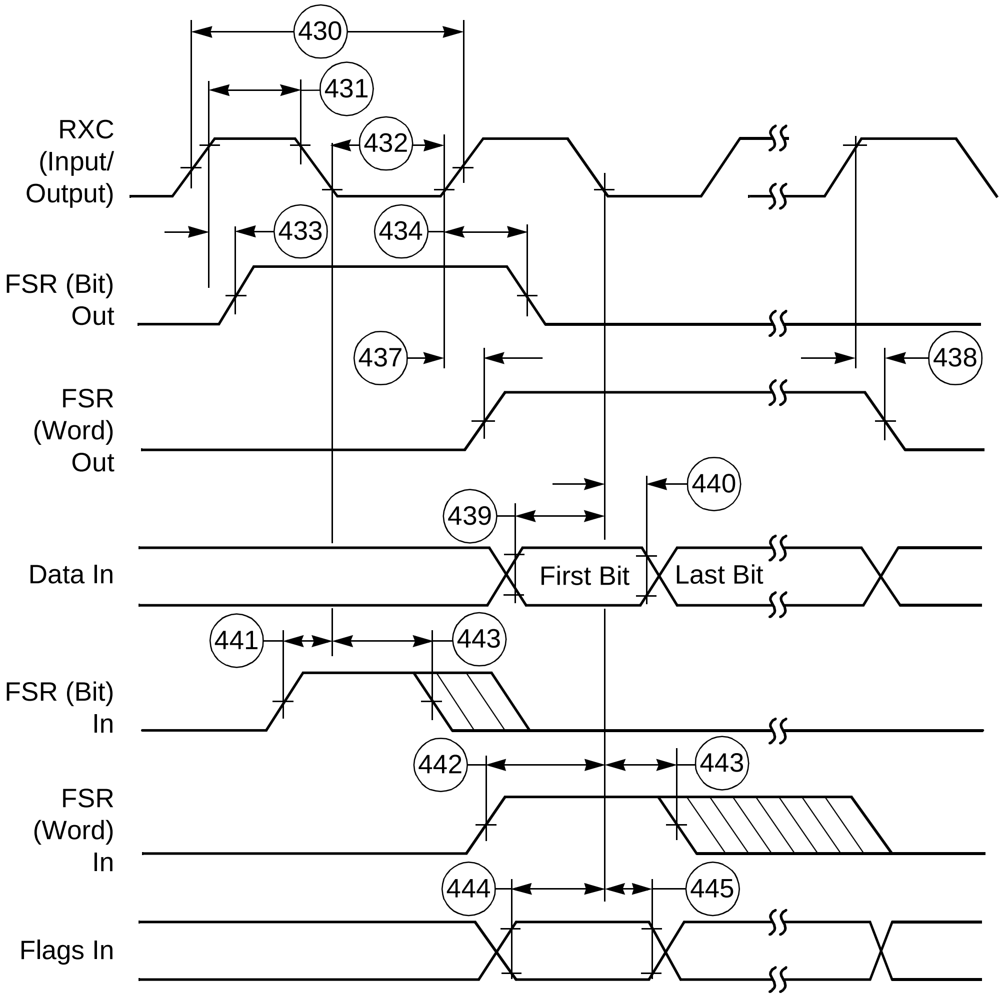


Figure 2-24. ESSI Receiver Timing

TIMER TIMING

Table 2-13. Timer Timing

No.	Characteristics	Expression	150 MHz		Unit
			Min	Max	
480	TIO Low	$2 \times T_C + 2.0$	15.4	—	ns
481	TIO High	$2 \times T_C + 2.0$	15.4	—	ns
~~482~~	~~Timer setup time from TIO (Input) assertion to CLKOUT rising edge~~		~~9.0~~	~~10.0~~	~~ns~~

Table 2-13. Timer Timing (Continued)

No.	Characteristics	Expression	150 MHz		Unit
			Min	Max	
~~483~~	~~Synchronous timer delay time from CLKOUT rising edge to the external memory access address out valid caused by first interrupt instruction execution~~	~~10.25 × T_C + 1.0~~	~~70.0~~	—	~~ns~~
~~484~~	~~CLKOUT rising edge to TIO (Output) assertion~~ - ~~Minimum~~ - ~~Maximum~~	 ~~0.5 × T_C + 3.5~~ ~~0.5 × T_C + 19.8~~	 ~~6.8~~ —	 — ~~23.2~~	 ~~ns~~ ~~ns~~
~~485~~	~~CLKOUT rising edge to TIO (Output) deassertion~~ - ~~Minimum~~ - ~~Maximum~~	 ~~60.5 × T_C + 3.5~~ ~~0.5 × T_C + 19.0~~	 ~~8.5~~ —	 — ~~24.8~~	 ~~ns~~ ~~ns~~
Note: V_{CCH} = 3.3 V ± 0.3 V, V_{CC} = 1.8 V ± 0.1 V; T_J = 0°C to +100 °C, C_L = 50 pF					

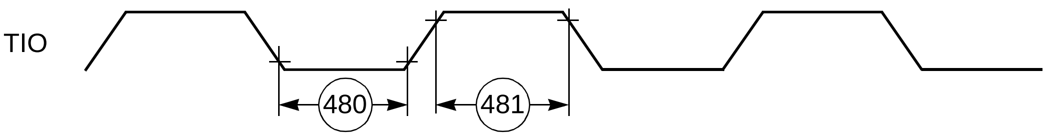


Figure 2-25. TIO Timer Event Input Restrictions

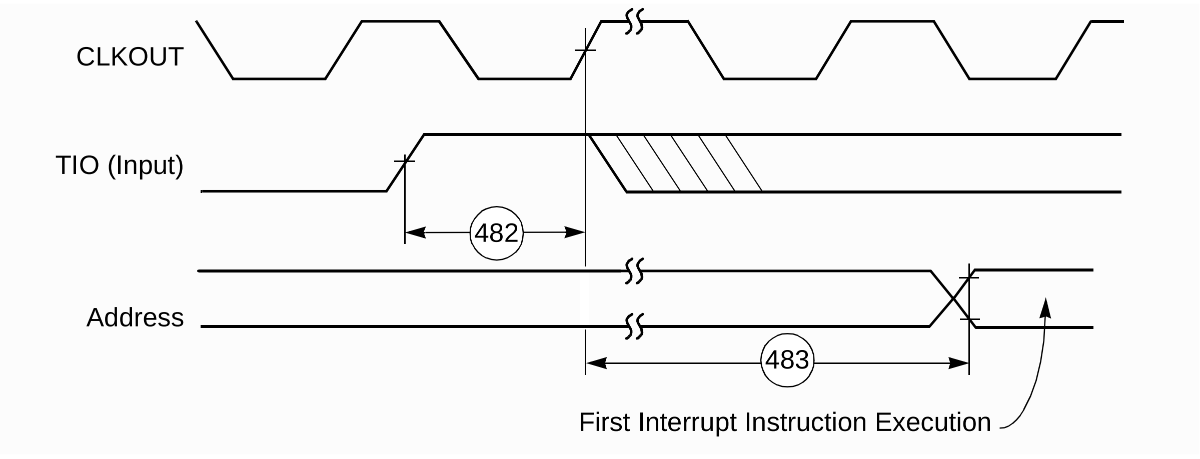


Figure 2-26. ~~Timer Interrupt Generation~~

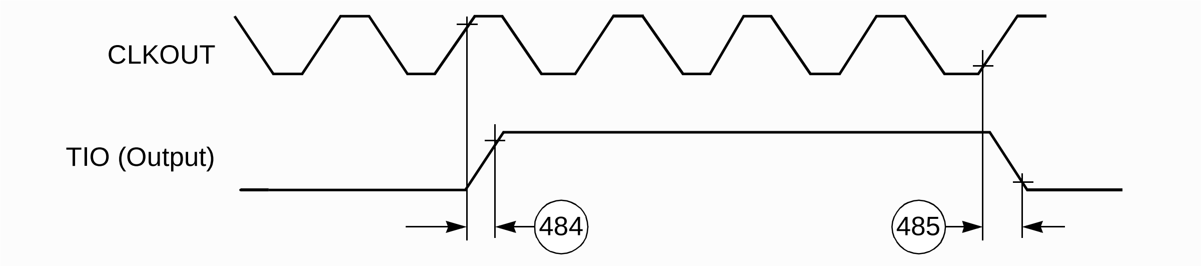


Figure 2-27. External Pulse Generation

CONSIDERATIONS FOR GPIO USE

The following considerations can be helpful when GPIO is used for output or input.

GPIO as Output

When you use a GPIO pin as an output, note that:

- The time from fetch of the instruction that changes the GPIO pin to the actual change is 7 core clock cycles. This is true, assuming that the instruction is a on- cycle instruction and that there are no pipeline stalls or any other pipeline delays.
- The maximum rise or fall time of a GPIO pin is 13ns (TTL levels, assuming that the maximum of 50 pF load limit is met).

GPIO as Input

GPIO inputs are not synchronized with the core clock. When only one GPIO bit is polled, this lack of synchronization presents no problem, since the read value can be either the previous value or the new value of the corresponding GPIO pin. However, there is the risk of reading an intermediate state if:

- Two or more GPIO bits are treated as a coupled group (for example, four possible status states encoded in two bits) and
- The read operation occurs during a simultaneous change of GPIO pins (for example, the change of 00 to 11 may happen through an intermediate state of 01 or 10).

Therefore, when GPIO bits are read, the recommended practice is to poll continuously until two consecutive read operations have identical results.

JTAG TIMING

Table 2-14. JTAG Timing

No.	Characteristics	All frequencies		Unit
		Min	Max	
500	TCK frequency of operation	0.0	22.0	MHz
501	TCK cycle time in Crystal mode	45.0	—	ns
502	TCK clock pulse width measured at 1.5 V	20.0	—	ns
503	TCK rise and fall times	0.0	3.0	ns
504	Boundary scan input data setup time	5.0	—	ns
505	Boundary scan input data hold time	24.0	—	ns
506	TCK low to output data valid	0.0	40.0	ns
507	TCK low to output high impedance	0.0	40.0	ns
508	TMS, TDI data setup time	5.0	—	ns
509	TMS, TDI data hold time	25.0	—	ns
510	TCK low to TDO data valid	0.0	44.0	ns
511	TCK low to TDO high impedance	0.0	44.0	ns
512	$\overline{\text{TRST}}$ assert time	100.0	—	ns
513	$\overline{\text{TRST}}$ setup time to TCK low	40.0	—	ns

Notes:
1. V_{CCH} = 3.3 V ± 0.3 V, V_{CC} = 1.8 V ± 0.1 V; T_J = 0°C to +100 °C, C_L = 50 pF
2. All timings apply to OnCE module data transfers because it uses the JTAG port as an interface.

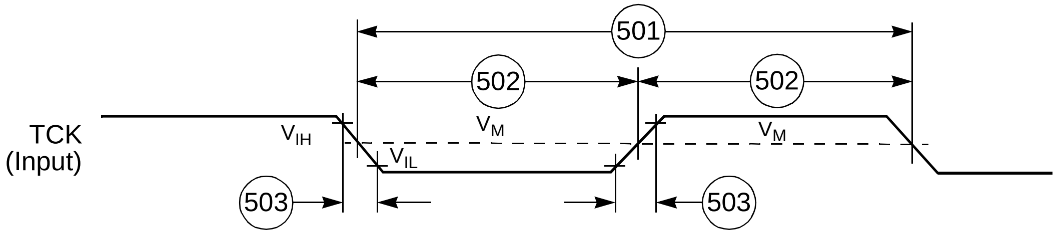


Figure 2-28. Test Clock Input Timing Diagram

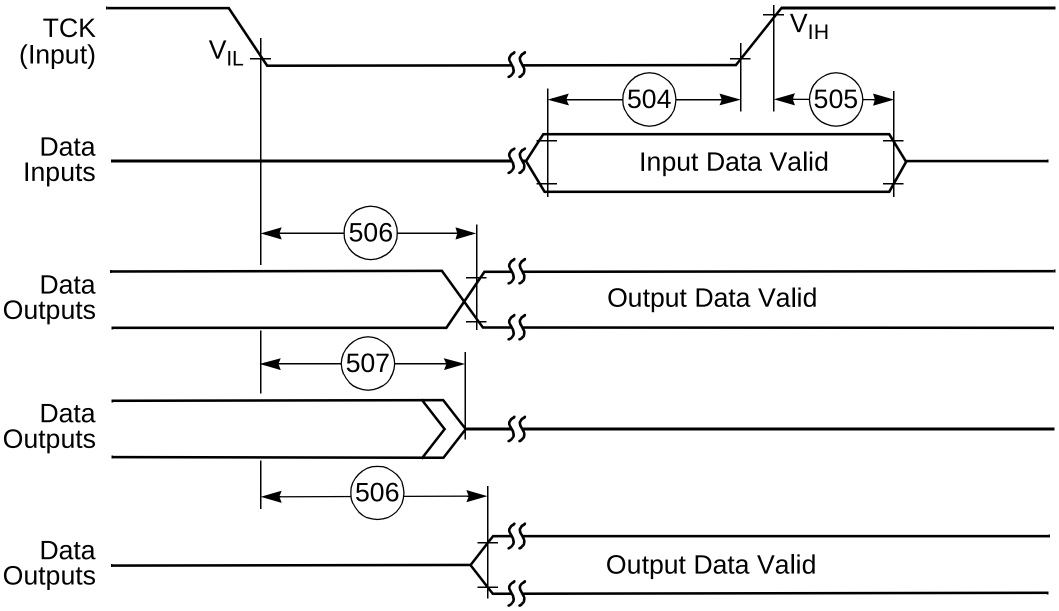


Figure 2-29. Boundary Scan (JTAG) Timing Diagram

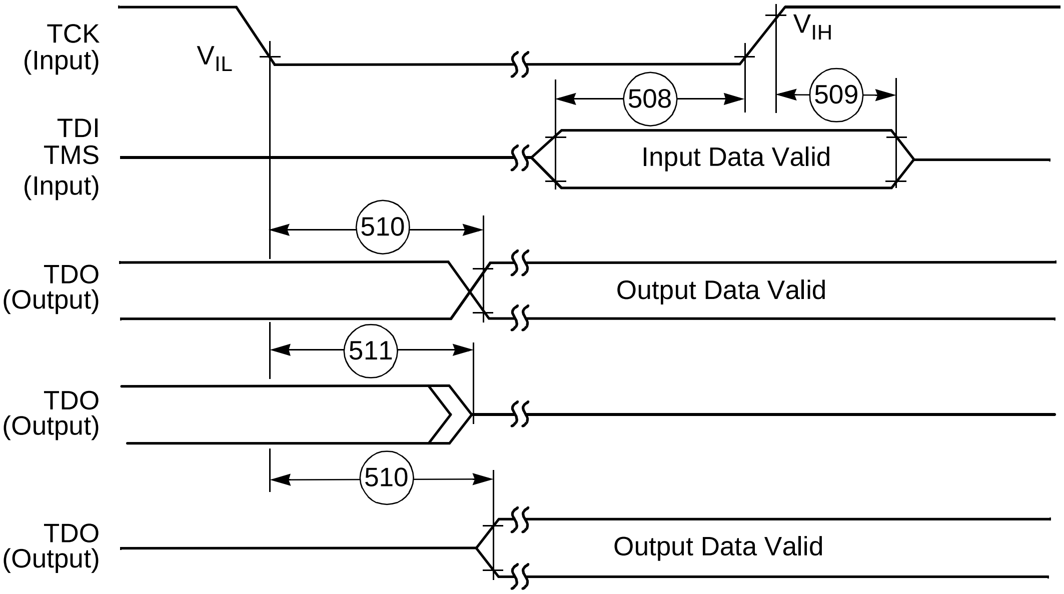


Figure 2-30. Test Access Port Timing Diagram

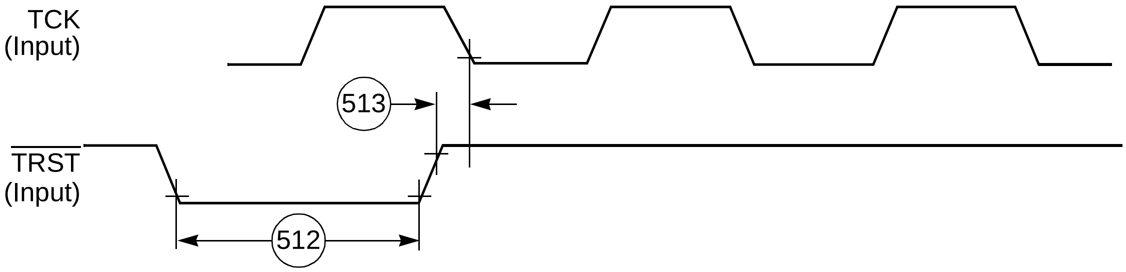


Figure 2-31. $\overline{\text{TRST}}$ Timing Diagram

OnCE MODULE TIMING

Table 2-15. OnCE Module Timing

No.	Characteristics	Expression	150 MHz		Unit
			Min	**Max**	
500	TCK frequency of operation	Max 22.0 MHz	0.0	22.0	MHz
514	$\overline{\text{DE}}$ assertion time in order to enter Debug mode	$1.5 \times T_C + 10.0$	20.0	—	ns
515	Response time when DSP56311 is executing NOP instructions from internal memory	$5.5 \times T_C + 30.0$	—	67.0	ns
516	Debug acknowledge assertion time	$3 \times T_C + 10.0$	30.0	—	ns
Note: V_{CCH} = 3.3 V ± 0.3 V, V_{CC} = 1.8 V ± 0.1 V; T_J = 0°C to +100 °C, C_L = 50 pF					

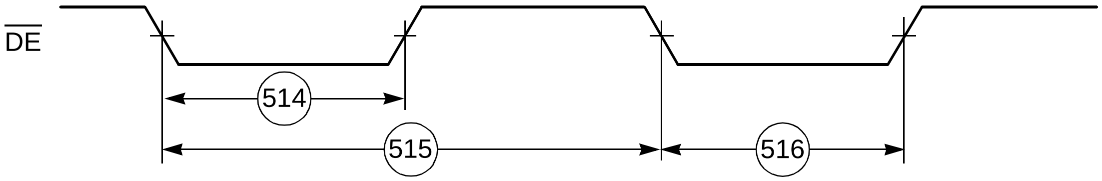


Figure 2-32. OnCE—Debug Request

SECTION 3

PACKAGING

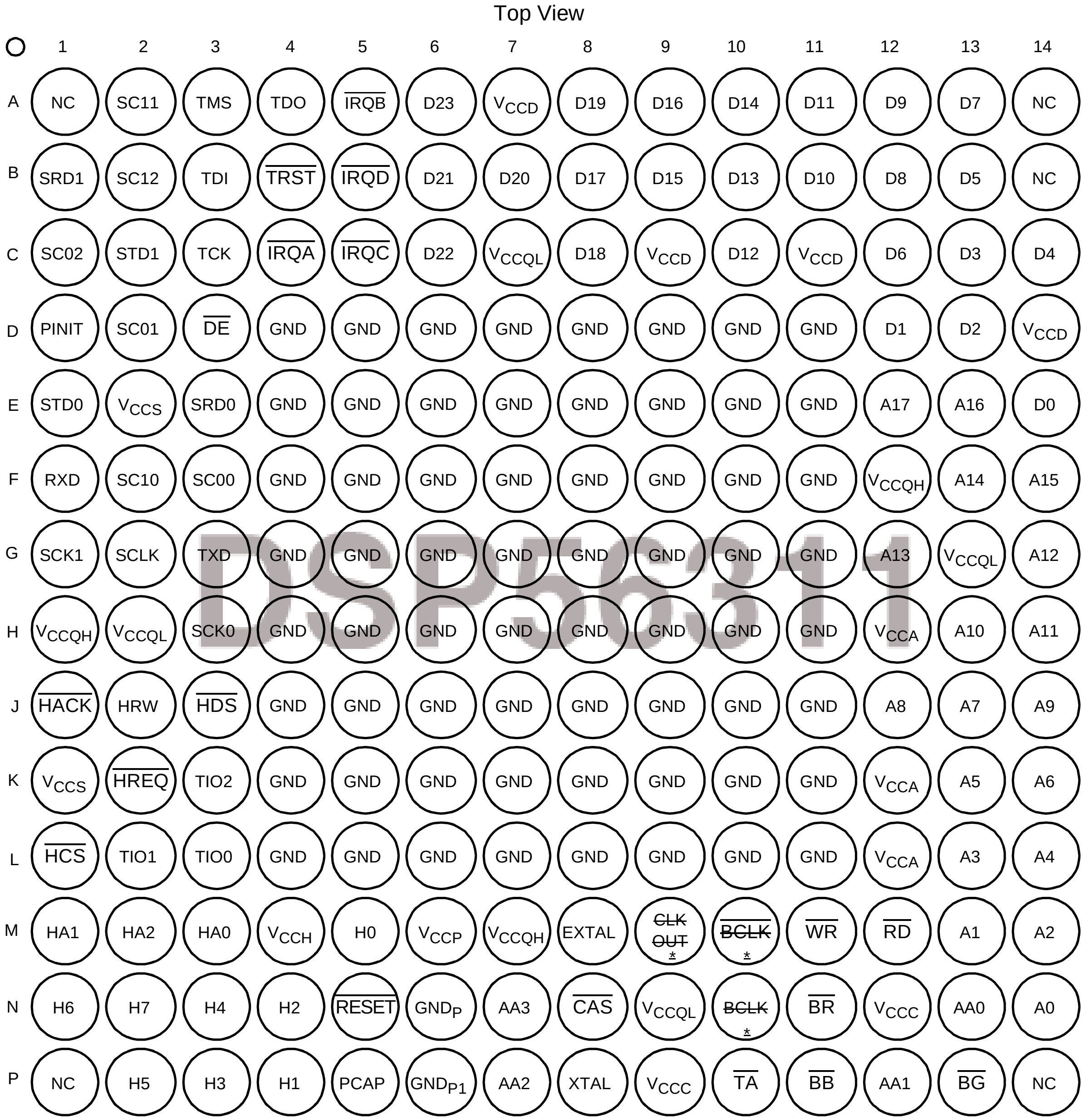
PIN-OUT AND PACKAGE INFORMATION

This section provides information on the package for the DSP56311, including diagrams of the package pinouts and tables describing how the signals described in **Section 1** are allocated for the package.

The DSP56311 is available in a 196-pin Plastic Ball Grid Array (PBGA) package.

PBGA Package Description

Top and bottom views of the PBGA package are shown in **Figure 3-1.** and **Figure 3-2.** with their pin-outs.



Top View

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
A	NC	SC11	TMS	TDO	$\overline{\text{IRQB}}$	D23	V_{CCD}	D19	D16	D14	D11	D9	D7	NC
B	SRD1	SC12	TDI	$\overline{\text{TRST}}$	$\overline{\text{IRQD}}$	D21	D20	D17	D15	D13	D10	D8	D5	NC
C	SC02	STD1	TCK	$\overline{\text{IRQA}}$	$\overline{\text{IRQC}}$	D22	V_{CCQL}	D18	V_{CCD}	D12	V_{CCD}	D6	D3	D4
D	PINIT	SC01	$\overline{\text{DE}}$	GND	GND	GND	GND	GND	GND	GND	GND	D1	D2	V_{CCD}
E	STD0	V_{CCS}	SRD0	GND	GND	GND	GND	GND	GND	GND	GND	A17	A16	D0
F	RXD	SC10	SC00	GND	GND	GND	GND	GND	GND	GND	GND	V_{CCQH}	A14	A15
G	SCK1	SCLK	TXD	GND	GND	GND	GND	GND	GND	GND	GND	A13	V_{CCQL}	A12
H	V_{CCQH}	V_{CCQL}	SCK0	GND	GND	GND	GND	GND	GND	GND	GND	V_{CCA}	A10	A11
J	$\overline{\text{HACK}}$	HRW	$\overline{\text{HDS}}$	GND	GND	GND	GND	GND	GND	GND	GND	A8	A7	A9
K	V_{CCS}	$\overline{\text{HREQ}}$	TIO2	GND	GND	GND	GND	GND	GND	GND	GND	V_{CCA}	A5	A6
L	$\overline{\text{HCS}}$	TIO1	TIO0	GND	GND	GND	GND	GND	GND	GND	GND	V_{CCA}	A3	A4
M	HA1	HA2	HA0	V_{CCH}	H0	V_{CCP}	V_{CCQH}	EXTAL	~~CLK OUT~~ *	~~$\overline{\text{BCLK}}$~~ *	$\overline{\text{WR}}$	$\overline{\text{RD}}$	A1	A2
N	H6	H7	H4	H2	$\overline{\text{RESET}}$	GND_P	AA3	$\overline{\text{CAS}}$	V_{CCQL}	~~BCLK~~ *	$\overline{\text{BR}}$	V_{CCC}	AA0	A0
P	NC	H5	H3	H1	PCAP	GND_{P1}	AA2	XTAL	V_{CCC}	$\overline{\text{TA}}$	$\overline{\text{BB}}$	AA1	$\overline{\text{BG}}$	NC

Figure 3-1. DSP56311 Plastic Ball Grid Array (PBGA), Top View

* Leave the CKLOUT, $\overline{\text{BCLK}}$ and BCLK pins unconnected. Do not tie them high or low.

Bottom View

14	13	12	11	10	9	8	7	6	5	4	3	2	1	○
NC	D7	D9	D11	D14	D16	D19	V_{CCD}	D23	$\overline{\text{IRQB}}$	TDO	TMS	SC11	NC	A
NC	D5	D8	D10	D13	D15	D17	D20	D21	$\overline{\text{IRQD}}$	$\overline{\text{TRST}}$	TDI	SC12	SRD1	B
D4	D3	D6	V_{CCD}	D12	V_{CCD}	D18	V_{CCQL}	D22	$\overline{\text{IRQC}}$	$\overline{\text{IRQA}}$	TCK	STD1	SC02	C
V_{CCD}	D2	D1	GND	GND	GND	GND	GND	GND	GND	GND	$\overline{\text{DE}}$	SC01	PINIT	D
D0	A16	A17	GND	GND	GND	GND	GND	GND	GND	GND	SRD0	V_{CCS}	STD0	E
A15	A14	V_{CCQH}	GND	GND	GND	GND	GND	GND	GND	GND	SC00	SC10	RXD	F
A12	V_{CCQL}	A13	GND	GND	GND	GND	GND	GND	GND	GND	TXD	SCLK	SCK1	G
A11	A10	V_{CCA}	GND	GND	GND	GND	GND	GND	GND	GND	SCK0	V_{CCQL}	V_{CCQH}	H
A9	A7	A8	GND	GND	GND	GND	GND	GND	GND	GND	$\overline{\text{HDS}}$	HRW	$\overline{\text{HACK}}$	J
A6	A5	V_{CCA}	GND	GND	GND	GND	GND	GND	GND	GND	TIO2	$\overline{\text{HREQ}}$	V_{CCS}	K
A4	A3	V_{CCA}	GND	GND	GND	GND	GND	GND	GND	GND	TIO0	TIO1	$\overline{\text{HCS}}$	L
A2	A1	$\overline{\text{RD}}$	$\overline{\text{WR}}$	~~$\overline{\text{BCLK}}$~~ *	~~CLK OUT~~ *	EXTAL	V_{CCQH}	V_{CCP}	H0	V_{CCH}	HA0	HA2	HA1	M
A0	AA0	V_{CCC}	$\overline{\text{BR}}$	~~BCLK~~ *	V_{CCQL}	$\overline{\text{CAS}}$	AA3	GND_P	$\overline{\text{RESET}}$	H2	H4	H7	H6	N
NC	$\overline{\text{BG}}$	AA1	$\overline{\text{BB}}$	$\overline{\text{TA}}$	V_{CCC}	XTAL	AA2	GND_{P1}	PCAP	H1	H3	H5	NC	P

Figure 3-2. DSP56311 Plastic Ball Grid Array (PBGA), Bottom View

* Leave the CKLOUT, $\overline{\text{BCLK}}$ and BCLK pins unconnected. Do not tie them high or low.

Table 3-1. DSP56311 PBGA Signal Identification by Pin Number

Pin No.	Signal Name	Pin No.	Signal Name	Pin No.	Signal Name
A1	Not Connected (NC), reserved	B12	D8	D9	GND
A2	SC11 or PD1	B13	D5	D10	GND
A3	TMS	B14	NC	D11	GND
A4	TDO	C1	SC02 or PC2	D12	D1
A5	MODB/$\overline{\text{IRQB}}$	C2	STD1 or PD5	D13	D2
A6	D23	C3	TCK	D14	V_{CCD}
A7	V_{CCD}	C4	MODA/$\overline{\text{IRQA}}$	E1	STD0 or PC5
A8	D19	C5	MODC/$\overline{\text{IRQC}}$	E2	V_{CCS}
A9	D16	C6	D22	E3	SRD0 or PC4
A10	D14	C7	V_{CCQL}	E4	GND
A11	D11	C8	D18	E5	GND
A12	D9	C9	V_{CCD}	E6	GND
A13	D7	C10	D12	E7	GND
A14	NC	C11	V_{CCD}	E8	GND
B1	SRD1 or PD4	C12	D6	E9	GND
B2	SC12 or PD2	C13	D3	E10	GND
B3	TDI	C14	D4	E11	$\overline{\text{GND}}$
B4	$\overline{\text{TRST}}$	D1	PINIT/$\overline{\text{NMI}}$	E12	A17
B5	MODD/$\overline{\text{IRQD}}$	D2	SC01 or PC1	E13	A16
B6	D21	D3	$\overline{\text{DE}}$	E14	D0
B7	D20	D4	GND	F1	RXD or PE0
B8	D17	D5	GND	F2	SC10 or PD0
B9	D15	D6	GND	F3	SC00 or PC0
B10	D13	D7	GND	F4	GND
B11	D10	D8	GND	F5	GND

Table 3-1. DSP56311 PBGA Signal Identification by Pin Number (Continued)

Pin No.	Signal Name	Pin No.	Signal Name	Pin No.	Signal Name
F6	GND	H3	SCK0 or PC3	J14	A9
F7	GND	H4	GND	K1	V_{CCS}
F8	GND	H5	GND	K2	$\overline{\text{HREQ}}$/HREQ, $\overline{\text{HTRQ}}$/HTRQ, or PB14
F9	GND	H6	GND	K3	TIO2
F10	GND	H7	GND	K4	GND
F11	GND	H8	GND	K5	GND
F12	V_{CCQH}	H9	GND	K6	GND
F13	A14	H10	GND	K7	GND
F14	A15	H11	GND	K8	GND
G1	SCK1 or PD3	H12	V_{CCA}	K9	GND
G2	SCLK or PE2	H13	A10	K10	GND
G3	TXD or PE1	H14	A11	K11	GND
G4	GND	J1	$\overline{\text{HACK}}$/HACK, $\overline{\text{HRRQ}}$/HRRQ, or PB15	K12	V_{CCA}
G5	GND	J2	HRW, $\overline{\text{HRD}}$/HRD, or PB11	K13	A5
G6	GND	J3	$\overline{\text{HDS}}$/HDS, $\overline{\text{HWR}}$/HWR, or PB12	K14	A6
G7	GND	J4	GND	L1	$\overline{\text{HCS}}$/HCS, HA10, or PB13
G8	GND	J5	GND	L2	TIO1
G9	GND	J6	GND	L3	TIO0
G10	GND	J7	GND	L4	GND
G11	GND	J8	GND	L5	GND
G12	A13	J9	GND	L6	GND
G13	V_{CCQL}	J10	GND	L7	GND
G14	A12	J11	GND	L8	GND
H1	V_{CCQH}	J12	A8	L9	GND

Table 3-1. DSP56311 PBGA Signal Identification by Pin Number (Continued)

Pin No.	Signal Name	Pin No.	Signal Name	Pin No.	Signal Name
H2	V_{CCQL}	J13	A7	L10	GND
L11	GND	M13	A1	P1	NC
L12	V_{CCA}	M14	A2	P2	H5, HAD5, or PB5
L13	A3	N1	H6, HAD6, or PB6	P3	H3, HAD3, or PB3
L14	A4	N2	H7, HAD7, or PB7	P4	H1, HAD1, or PB1
M1	HA1, HA8, or PB9	N3	H4, HAD4, or PB4	P5	PCAP
M2	HA2, HA9, or PB10	N4	H2, HAD2, or PB2	P6	GND_{P1}
M3	HA0, $\overline{\text{HAS}}$/HAS, or PB8	N5	$\overline{\text{RESET}}$	P7	AA2/$\overline{\text{RAS2}}$
M4	V_{CCH}	N6	GND_{P}	P8	XTAL
M5	H0, HAD0, or PB0	N7	AA3/$\overline{\text{RAS3}}$	P9	V_{CCC}
M6	V_{CCP}	N8	$\overline{\text{CAS}}$	P10	$\overline{\text{TA}}$
M7	V_{CCQH}	N9	V_{CCQL}	P11	$\overline{\text{BB}}$
M8	EXTAL	~~N10~~	~~BCLK~~	P12	AA1/$\overline{\text{RAS1}}$
~~M9~~	~~CLKOUT~~	N11	$\overline{\text{BR}}$	P13	$\overline{\text{BG}}$
M10	~~$\overline{\text{BCLK}}$~~	N12	V_{CCC}	P14	NC
M11	$\overline{\text{WR}}$	N13	AA0/$\overline{\text{RAS0}}$		
M12	$\overline{\text{RD}}$	N14	A0		

Note: Signal names are based on configured functionality. Most connections supply a single signal. Some connections provide a signal with dual functionality, such as the MODx/$\overline{\text{IRQx}}$ pins that select an operating mode after $\overline{\text{RESET}}$ is deasserted but act as interrupt lines during operation. Some signals have configurable polarity; these names are shown with and without overbars, such as $\overline{\text{HAS}}$/HAS. Some connections have two or more configurable functions; names assigned to these connections indicate the function for a specific configuration. For example, connection N2 is data line H7 in non-multiplexed bus mode, data/address line HAD7 in multiplexed bus mode, or GPIO line PB7 when the GPIO function is enabled for this pin. Unlike the TQFP package, most of the GND pins are connected internally in the center of the connection array and act as heat sink for the chip. Therefore, except for GND_{P} and GND_{P1} that support the PLL, other GND signals do not support individual subsystems in the chip.

Table 3-2. DSP56311 PBGA Signal Identification by Name

Signal Name	Pin No.	Signal Name	Pin No.	Signal Name	Pin No.
A0	N14	$\overline{\text{BG}}$	P13	D7	A13
A1	M13	$\overline{\text{BR}}$	N11	D8	B12
A10	H13	$\overline{\text{CAS}}$	N8	D9	A12
A11	H14	~~CLKOUT~~	M9	$\overline{\text{DE}}$	D3
A12	G14	D0	E14	EXTAL	M8
A13	G12	D1	D12	GND	D4
A14	F13	D10	B11	GND	D5
A15	F14	D11	A11	GND	D6
A16	E13	D12	C10	GND	D7
A17	E12	D13	B10	GND	D8
A2	M14	D14	A10	GND	D9
A3	L13	D15	B9	GND	D10
A4	L14	D16	A9	GND	D11
A5	K13	D17	B8	GND	E4
A6	K14	D18	C8	GND	E5
A7	J13	D19	A8	GND	E6
A8	J12	D2	D13	GND	E7
A9	J14	D20	B7	GND	E8
AA0	N13	D21	B6	GND	E9
AA1	P12	D22	C6	GND	E10
AA2	P7	D23	A6	GND	E11
AA3	N7	D3	C13	GND	F4
$\overline{\text{BB}}$	P11	D4	C14	GND	F5
~~$\overline{\text{BCLK}}$~~	M10	D5	B13	GND	F6
~~BCLK~~	N10	D6	C12	GND	F7

Table 3-2. DSP56311 PBGA Signal Identification by Name (Continued)

Signal Name	Pin No.	Signal Name	Pin No.	Signal Name	Pin No.
GND	F8	GND	J9	H4	N3
GND	F9	GND	J10	H5	P2
GND	F10	GND	J11	H6	N1
GND	F11	GND	K4	H7	N2
GND	G4	GND	K5	HA0	M3
GND	G5	GND	K6	HA1	M1
GND	G6	GND	K7	HA10	L1
GND	G7	GND	K8	HA2	M2
GND	G8	GND	K9	HA8	M1
GND	G9	GND	K10	HA9	M2
GND	G10	GND	K11	$\overline{\text{HACK}}$/HACK	J1
GND	G11	GND	L4	HAD0	M5
GND	H4	GND	L5	HAD1	P4
GND	H5	GND	L6	HAD2	N4
GND	H6	GND	L7	HAD3	P3
GND	H7	GND	L8	HAD4	N3
GND	H8	GND	L9	HAD5	P2
GND	H9	GND	L10	HAD6	N1
GND	H10	GND	L11	HAD7	N2
GND	H11	GND$_P$	N6	$\overline{\text{HAS}}$/HAS	M3
GND	J4	GND$_{P1}$	P6	$\overline{\text{HCS}}$/HCS	L1
GND	J5	H0	M5	$\overline{\text{HDS}}$/HDS	J3
GND	J6	H1	P4	$\overline{\text{HRD}}$/HRD	J2
GND	J7	H2	N4	$\overline{\text{HREQ}}$/HREQ	K2
GND	J8	H3	P3	$\overline{\text{HRRQ}}$/HRRQ	J1
HRW	J2	PB2	N4	$\overline{\text{RAS0}}$	N13

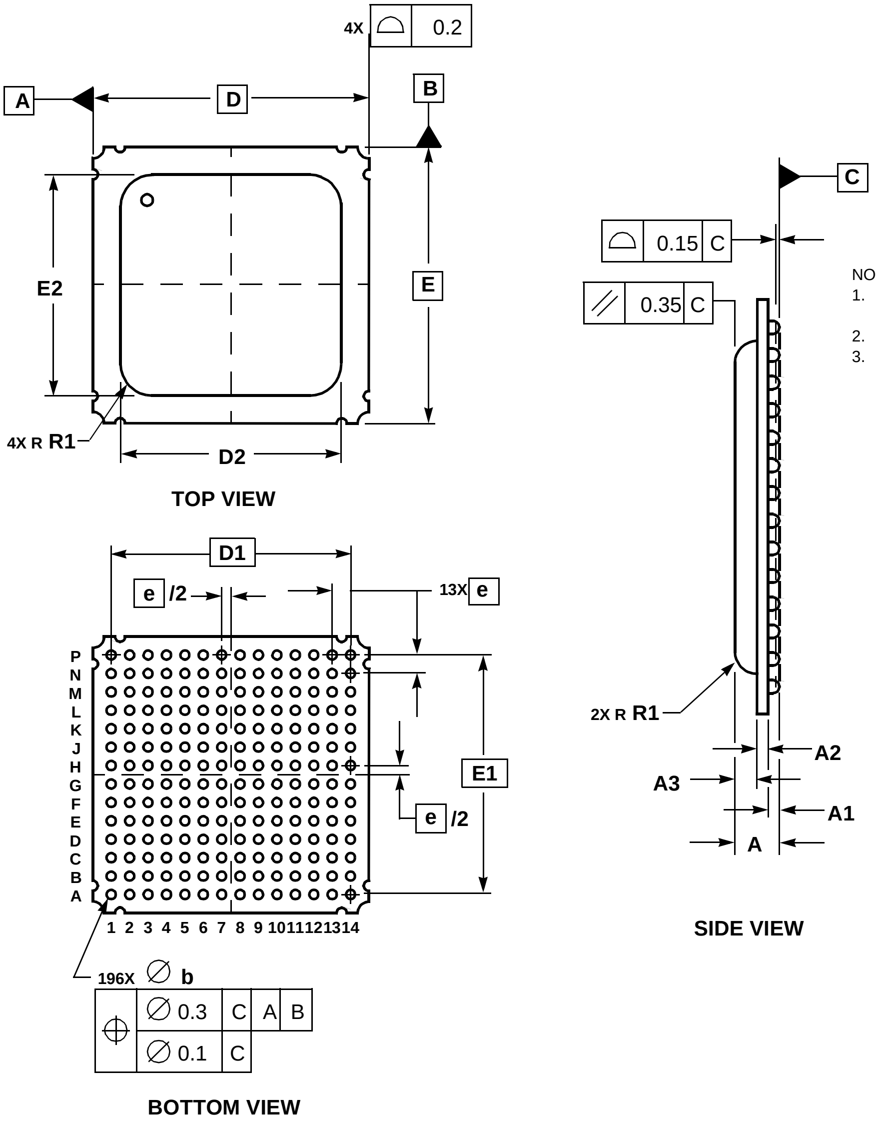
Table 3-2. DSP56311 PBGA Signal Identification by Name (Continued)

Signal Name	Pin No.	Signal Name	Pin No.	Signal Name	Pin No.
$\overline{\text{HTRQ}}$/HTRQ	K2	PB3	P3	$\overline{\text{RAS1}}$	P12
$\overline{\text{HWR}}$/HWR	J3	PB4	N3	$\overline{\text{RAS2}}$	P7
$\overline{\text{IRQA}}$	C4	PB5	P2	$\overline{\text{RAS3}}$	N7
$\overline{\text{IRQB}}$	A5	PB6	N1	$\overline{\text{RD}}$	M12
$\overline{\text{IRQC}}$	C5	PB7	N2	$\overline{\text{RESET}}$	N5
$\overline{\text{IRQD}}$	B5	PB8	M3	RXD	F1
MODA	C4	PB9	M1	SC00	F3
MODB	A5	PC0	F3	SC01	D2
MODC	C5	PC1	D2	SC02	C1
MODD	B5	PC2	C1	SC10	F2
NC	A1	PC3	H3	SC11	A2
NC	A14	PC4	E3	SC12	B2
NC	B14	PC5	E1	SCK0	H3
NC	P1	PCAP	P5	SCK1	G1
NC	P14	PD0	F2	SCLK	G2
$\overline{\text{NMI}}$	D1	PD1	A2	SRD0	E3
PB0	M5	PD2	B2	SRD1	B1
PB1	P4	PD3	G1	STD0	E1
PB10	M2	PD4	B1	STD1	C2
PB11	J2	PD5	C2	$\overline{\text{TA}}$	P10
PB12	J3	PE0	F1	TCK	C3
PB13	L1	PE1	G3	TDI	B3
PB14	K2	PE2	G2	TDO	A4
PB15	J1	PINIT	D1	TIO0	L3
TIO1	L2	V_{CCC}	P9	V_{CCQH}	M7
TIO2	K3	V_{CCD}	A7	V_{CCQL}	C7

Table 3-2. DSP56311 PBGA Signal Identification by Name (Continued)

Signal Name	Pin No.	Signal Name	Pin No.	Signal Name	Pin No.
TMS	A3	V_{CCD}	C9	V_{CCQL}	G13
$\overline{TRST}$	B4	V_{CCD}	C11	V_{CCQL}	H2
TXD	G3	V_{CCD}	D14	V_{CCQL}	N9
V_{CCA}	H12	V_{CCH}	M4	V_{CCS}	E2
V_{CCA}	K12	V_{CCP}	M6	V_{CCS}	K1
V_{CCA}	L12	V_{CCQH}	F12	$\overline{WR}$	M11
V_{CCC}	N12	V_{CCQH}	H1	XTAL	P8

PBGA Package Mechanical Drawing



NOTES:

1. Dimensioning and tolerancing per ASME Y14.5M, 1994.
2. Dimensions in millimeters.
3. Dimension b is the solder bal diameter measured parallel T datum C.

	MILLIMETERS	
DIM	MIN	MAX
A	1.25	1.91
A1	0.27	0.47
A2	0.28	0.44
A3	0.70	1.00
b	0.35	0.65
D	15.00 BSC	
D1	13.00 BSC	
D2	12.00	15.00
E	15.00 BSC	
E1	13.00 BSC	
E2	12.00	15.00
e	1.00 BSC	
R1	—	2.50

CASE 1128-01
ISSUE B
DATE 11/22/96

Figure 3-3. DSP56311 Mechanical Information, 196-pin PBGA Package

ORDERING DRAWINGS

Complete mechanical information on DSP56311 packaging is available by facsimile through Motorola's Mfax system. Call the following number to obtain information by facsimile:

(602) 244-6609

The Mfax automated system requests the following information:

- The receiving facsimile telephone number including area code or country code
- The caller's personal identification number (PIN)

Note: For first time callers, the system provides instructions for setting up a PIN, which requires entry of a name and telephone number.

- The type of information requested:
  - Instructions for using the system
  - A literature order form
  - Specific part technical information or data sheets
  - Other information described by the system messages

A total of three documents can be ordered per call.

The DSP56311 196-pin PBGA package mechanical drawing is referenced as 1128-01.

SECTION 4

DESIGN CONSIDERATIONS

THERMAL DESIGN CONSIDERATIONS

An estimate of the chip junction temperature, T_J, in °C can be obtained from this equation:

Equation 1: $T_J = T_A + (P_D \times R_{\theta JA})$

Where:

T_A = ambient temperature °C
$R_{\theta JA}$ = package junction-to-ambient thermal resistance °C/W
P_D = power dissipation in package

Historically, thermal resistance has been expressed as the sum of a junction-to-case thermal resistance and a case-to-ambient thermal resistance, as in this equation:

Equation 2: $R_{\theta JA} = R_{\theta JC} + R_{\theta CA}$

Where:

$R_{\theta JA}$ = package junction-to-ambient thermal resistance °C/W
$R_{\theta JC}$ = package junction-to-case thermal resistance °C/W
$R_{\theta CA}$ = package case-to-ambient thermal resistance °C/W

$R_{\theta JC}$ is device-related and cannot be influenced by the user. The user controls the thermal environment to change the case-to-ambient thermal resistance, $R_{\theta CA}$. For example, the user can change the air flow around the device, add a heat sink, change the mounting arrangement on the printed circuit board (PCB) or otherwise change the thermal dissipation capability of the area surrounding the device on a PCB. This model is most useful for ceramic packages with heat sinks; some 90 percent of the heat flow is dissipated through the case to the heat sink and out to the ambient environment. For ceramic packages, in situations where the heat flow is split between a path to the case and an alternate path through the PCB, analysis of the device thermal performance may need the additional modeling capability of a system-level thermal simulation tool.

The thermal performance of plastic packages is more dependent on the temperature of the PCB to which the package is mounted. Again, if the estimates obtained from $R_{\theta JA}$ do not satisfactorily answer whether the thermal performance is adequate, a system-level model may be appropriate.

A complicating factor is the existence of three common ways to determine the junction-to-case thermal resistance in plastic packages.

- To minimize temperature variation across the surface, the thermal resistance is measured from the junction to the outside surface of the package (case) closest to the chip mounting area when that surface has a proper heat sink.
- To define a value approximately equal to a junction-to-board thermal resistance, the thermal resistance is measured from the junction to the point at which the leads attach to the case.
- If the temperature of the package case (T_T) is determined by a thermocouple, thermal resistance is computed from the value obtained by the equation $(T_J - T_T)/P_D$.

As noted earlier, the junction-to-case thermal resistances quoted in this data sheet are determined using the first definition. From a practical standpoint, that value is also suitable to determine the junction temperature from a case thermocouple reading in forced convection environments. In natural convection, the use of the junction-to-case thermal resistance to estimate junction temperature from a thermocouple reading on the case of the package will yield an estimate of a junction temperature slightly higher than actual temperature. Hence, the new thermal metric, thermal characterization parameter or Ψ_{JT}, has been defined to be $(T_J - T_T)/P_D$. This value gives a better estimate of the junction temperature in natural convection when the surface temperature of the package is used. Remember that surface temperature readings of packages are subject to significant errors caused by inadequate attachment of the sensor to the surface and to errors caused by heat loss to the sensor. The recommended technique is to attach a 40-gauge thermocouple wire and bead to the top center of the package with thermally conductive epoxy.

ELECTRICAL DESIGN CONSIDERATIONS

CAUTION

This device contains protective circuitry to guard against damage due to high static voltage or electrical fields. However, normal precautions are advised to avoid application of any voltages higher than maximum rated voltages to this high-impedance circuit. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (for example, either GND or V_{CC}).

Use the following list of recommendations to insure correct DSP operation.

- Provide a low-impedance path from the board power supply to each V_{CC} pin on the DSP and from the board ground to each GND pin.
- Use at least six 0.01–0.1 μF bypass capacitors positioned as close as possible to the four sides of the package to connect the V_{CC} power source to GND.
- Ensure that capacitor leads and associated printed circuit traces that connect to the chip V_{CC} and GND pins are less than 0.5 inch per capacitor lead.
- Use at least a four-layer PCB with two inner layers for V_{CC} and GND.
- Because the DSP output signals have fast rise and fall times, PCB trace lengths should be minimal. This recommendation particularly applies to the address and data buses as well as the $\overline{\mathrm{IRQA}}$, $\overline{\mathrm{IRQB}}$, $\overline{\mathrm{IRQC}}$, $\overline{\mathrm{IRQD}}$, $\overline{\mathrm{TA}}$, and $\overline{\mathrm{BG}}$ pins. Maximum PCB trace lengths on the order of 6 inches are recommended.
- Consider all device loads as well as parasitic capacitance due to PCB traces when you calculate capacitance. This is especially critical in systems with higher capacitive loads that could create higher transient currents in the V_{CC} and GND circuits.
- All inputs must be terminated (that is, not allowed to float) by CMOS levels except for the three pins with internal pull-up resistors ($\overline{\mathrm{TRST}}$, TMS, $\overline{\mathrm{DE}}$).
- Take special care to minimize noise levels on the V_{CCP}, GND_P, and GND_{P1} pins.
- The following pins must be asserted after power-up: $\overline{\mathrm{RESET}}$ and $\overline{\mathrm{TRST}}$.
- If multiple DSP devices are on the same board, check for cross-talk or excessive spikes on the supplies due to synchronous operation of the devices.
- RESET must be asserted when the chip is powered up. A stable EXTAL signal should be supplied before deassertion of RESET.

POWER CONSUMPTION CONSIDERATIONS

Power dissipation is a key issue in portable DSP applications. Some of the factors affecting current consumption are described in this section. Most of the current consumed by CMOS devices is alternating current (ac), which is charging and discharging the capacitances of the pins and internal nodes.

Current consumption is described by this formula:

Equation 3: $I = C \times V \times f$

Where:

C = node/pin capacitance
V = voltage swing
f = frequency of node/pin toggle

Example 4-1. Current Consumption

For a Port A address pin loaded with 50 pF capacitance, operating at 3.3 V, with a 66 MHz clock, toggling at its maximum possible rate (33 MHz), the current consumption is expressed in this equation:

Equation 4: $$I = 50 \times 10^{-12} \times 3.3 \times 33 \times 10^{6} = 5.48 \text{ mA}$$

The maximum internal current (I_{CCI}max) value reflects the typical possible switching of the internal buses on best-case operation conditions—not necessarily a real application case. The typical internal current (I_{CCItyp}) value reflects the average switching of the internal buses on typical operating conditions.

Perform the following steps for applications that require very low current consumption:

1. Set the EBD bit when you are not accessing external memory.
2. Minimize external memory accesses, and use internal memory accesses.
3. Minimize the number of pins that are switching.
4. Minimize the capacitive load on the pins.
5. Connect the unused inputs to pull-up or pull-down resistors.
6. Disable unused peripherals.
7. Disable unused pin activity (for example, CLKOUT, XTAL).

One way to evaluate power consumption is to use a current-per-MIPS measurement methodology to minimize specific board effects (that is, to compensate for measured board current not caused by the DSP). A benchmark power consumption test algorithm is listed in **Appendix A**. Use the test algorithm, specific test current measurements, and the following equation to derive the current-per-MIPS value.

Equation 5: $$I / MIPS = I / MHz = (I_{typF2} - I_{typF1}) / (F2 - F1)$$

Where:

I_{typF2} = current at F2
I_{typF1} = current at F1
F2 = high frequency (any specified operating frequency)
F1 = low frequency (any specified operating frequency lower than F2)

Note: F1 should be significantly less than F2. For example, F2 could be 66 MHz and F1 could be 33 MHz. The degree of difference between F1 and F2 determines the amount of precision with which the current rating can be determined for an application.

PLL PERFORMANCE ISSUES

The following explanations should be considered as general observations on expected PLL behavior. There is no test that replicates these exact numbers. These observations were measured on a limited number of parts and were not verified over the entire temperature and voltage ranges.

~~Phase Skew Performance~~

~~The phase skew of the PLL is defined as the time difference between the falling edges of EXTAL and CLKOUT for a given capacitive load on CLKOUT over the entire process, temperature, and voltage ranges. As defined in **Figure 2-2.** on page 2-7for input frequencies greater than 15 MHz and the MF ≤ 4, this skew is greater than or equal to 0.0 ns and less than 1.8 ns; otherwise, this skew is not guaranteed. However, for MF < 10 and input frequencies greater than 10 MHz, this skew is between –1.4 ns and +3.2 ns.~~

~~Phase Jitter Performance~~

~~The phase jitter of the PLL is defined as the variations in the skew between the falling edges of EXTAL and CLKOUT for a given device in specific temperature, voltage, input frequency, MF, and capacitive load on CLKOUT. These variations are a result of the PLL locking mechanism. For input frequencies greater than 15 MHz and MF ≤ 4, this jitter is less than ±0.6 ns; otherwise, this jitter is not guaranteed. However, for MF < 10 and input frequencies greater than 10 MHz, this jitter is less than ±2 ns.~~

~~Frequency Jitter Performance~~

~~The frequency jitter of the PLL is defined as the variation of the frequency of CLKOUT. For small MF (MF < 10) this jitter is smaller than 0.5 percent. For mid-range MF (10 < MF < 500) this jitter is between 0.5 percent and approximately 2 percent. For large MF (MF > 500), the frequency jitter is 2–3 percent.~~

Input (EXTAL) Jitter Requirements

The allowed jitter on the frequency of EXTAL is 0.5 percent. If the rate of change of the frequency of EXTAL is slow (that is, it does not jump between the minimum and maximum values in one cycle) or the frequency of the jitter is fast (that is, it does not stay at an extreme value for a long time), then the allowed jitter can be 2 percent. The phase and frequency jitter performance results are valid only if the input jitter is less than the prescribed values.

APPENDIX A

POWER CONSUMPTION BENCHMARK

The following benchmark program evaluates DSP power use in a test situation. It enables the PLL, disables the external clock, and uses repeated multiply-accumulate (MAC) instructions with a set of synthetic DSP application data to emulate intensive sustained DSP operation.

```
;**************************************************************************
;**************************************************************************
;*                                                                         *
;* CHECKS   Typical Power Consumption                                      *
;*                                                                         *
;**************************************************************************

          page     200,55,0,0,0
          nolist

I_VEC EQU $000000  ; Interrupt vectors for program debug only
START EQU $8000    ; MAIN (external) program starting address
INT_PROG  EQU $100 ; INTERNAL program memory starting address
INT_XDAT  EQU $0   ; INTERNAL X-data memory starting address
INT_YDAT  EQU $0   ; INTERNAL Y-data memory starting address

          INCLUDE "ioequ.asm"
          INCLUDE "intequ.asm"

          list

          org      P:START
;
          movep #$0243FF,x:M_BCR ;; BCR: Area 3 = 2 w.s (SRAM)
; Default: 2w.s (SRAM)
;
          movep    #$0d0000,x:M_PCTL           ; XTAL disable
                                               ; PLL enable
                                               ; CLKOUT disable
;
; Load the program
;
          move     #INT_PROG,r0
          move     #PROG_START,r1
          do       #(PROG_END-PROG_START),PLOAD_LOOP
          move     p:(r1)+,x0
          move     x0,p:(r0)+
          nop
PLOAD_LOOP
;
; Load the X-data
;
```

```
         move     #INT_XDAT,r0
         move     #XDAT_START,r1
         do       #(XDAT_END-XDAT_START),XLOAD_LOOP
         move     p:(r1)+,x0
         move     x0,x:(r0)+
XLOAD_LOOP
;
; Load the Y-data
;
         move     #INT_YDAT,r0
         move     #YDAT_START,r1
         do       #(YDAT_END-YDAT_START),YLOAD_LOOP
         move     p:(r1)+,x0
         move     x0,y:(r0)+
YLOAD_LOOP
;

         jmp      INT_PROG

PROG_START
         move     #$0,r0
         move     #$0,r4
         move     #$3f,m0
         move     #$3f,m4
;
         clr      a
         clr      b
         move     #$0,x0
         move     #$0,x1
         move     #$0,y0
         move     #$0,y1
         bset     #4,omr             ; ebd
;
sbr      dor      #60,_end
         mac      x0,y0,a  x:(r0)+,x1        y:(r4)+,y1
         mac      x1,y1,a  x:(r0)+,x0        y:(r4)+,y0
         add      a,b
         mac      x0,y0,a  x:(r0)+,x1
         mac      x1,y1,a                    y:(r4)+,y0
         move     b1,x:$ff
_end
         bra      sbr
         nop
         nop
         nop
         nop
PROG_END
         nop
         nop


XDAT_START
;        org      x:0
```

```
        dc        $262EB9
        dc        $86F2FE
        dc        $E56A5F
        dc        $616CAC
        dc        $8FFD75
        dc        $9210A
        dc        $A06D7B
        dc        $CEA798
        dc        $8DFBF1
        dc        $A063D6
        dc        $6C6657
        dc        $C2A544
        dc        $A3662D
        dc        $A4E762
        dc        $84F0F3
        dc        $E6F1B0
        dc        $B3829
        dc        $8BF7AE
        dc        $63A94F
        dc        $EF78DC
        dc        $242DE5
        dc        $A3E0BA
        dc        $EBAB6B
        dc        $8726C8
        dc        $CA361
        dc        $2F6E86
        dc        $A57347
        dc        $4BE774
        dc        $8F349D
        dc        $A1ED12
        dc        $4BFCE3
        dc        $EA26E0
        dc        $CD7D99
        dc        $4BA85E
        dc        $27A43F
        dc        $A8B10C
        dc        $D3A55
        dc        $25EC6A
        dc        $2A255B
        dc        $A5F1F8
        dc        $2426D1
        dc        $AE6536
        dc        $CBBC37
        dc        $6235A4
        dc        $37F0D
        dc        $63BEC2
        dc        $A5E4D3
        dc        $8CE810
        dc        $3FF09
        dc        $60E50E
        dc        $CFFB2F
        dc        $40753C
        dc        $8262C5
```

```
          dc        $CA641A
          dc        $EB3B4B
          dc        $2DA928
          dc        $AB6641
          dc        $28A7E6
          dc        $4E2127
          dc        $482FD4
          dc        $7257D
          dc        $E53C72
          dc        $1A8C3
          dc        $E27540
XDAT_END

YDAT_START
;         org       y:0
          dc        $5B6DA
          dc        $C3F70B
          dc        $6A39E8
          dc        $81E801
          dc        $C666A6
          dc        $46F8E7
          dc        $AAEC94
          dc        $24233D
          dc        $802732
          dc        $2E3C83
          dc        $A43E00
          dc        $C2B639
          dc        $85A47E
          dc        $ABFDDF
          dc        $F3A2C
          dc        $2D7CF5
          dc        $E16A8A
          dc        $ECB8FB
          dc        $4BED18
          dc        $43F371
          dc        $83A556
          dc        $E1E9D7
          dc        $ACA2C4
          dc        $8135AD
          dc        $2CE0E2
          dc        $8F2C73
          dc        $432730
          dc        $A87FA9
          dc        $4A292E
          dc        $A63CCF
          dc        $6BA65C
          dc        $E06D65
          dc        $1AA3A
          dc        $A1B6EB
          dc        $48AC48
          dc        $EF7AE1
          dc        $6E3006
          dc        $62F6C7
```

```
         dc       $6064F4
         dc       $87E41D
         dc       $CB2692
         dc       $2C3863
         dc       $C6BC60
         dc       $43A519
         dc       $6139DE
         dc       $ADF7BF
         dc       $4B3E8C
         dc       $6079D5
         dc       $E0F5EA
         dc       $8230DB
         dc       $A3B778
         dc       $2BFE51
         dc       $E0A6B6
         dc       $68FFB7
         dc       $28F324
         dc       $8F2E8D
         dc       $667842
         dc       $83E053
         dc       $A1FD90
         dc       $6B2689
         dc       $85B68E
         dc       $622EAF
         dc       $6162BC
         dc       $E4A245
YDAT_END

;**************************************************************************
;
;     EQUATES for DSP56311 I/O registers and ports
;
;     Last update: June 11 1995
;
;**************************************************************************

         page     132,55,0,0,0
         opt      mex

ioequ   ident   1,0

;------------------------------------------------------------------------
;
;       EQUATES for I/O Port Programming
;
;------------------------------------------------------------------------

;       Register Addresses

M_HDR EQU $FFFFC9            ; Host port GPIO data Register
M_HDDR EQU $FFFFC8           ; Host port GPIO direction Register
M_PCRC EQU $FFFFBF           ; Port C Control Register
M_PRRC EQU $FFFFBE           ; Port C Direction Register
```

```
M_PDRC EQU $FFFFBD         ; Port C GPIO Data Register
M_PCRD EQU $FFFFAF         ; Port D Control register
M_PRRD EQU $FFFFAE         ; Port D Direction Data Register
M_PDRD EQU $FFFFAD         ; Port D GPIO Data Register
M_PCRE EQU $FFFF9F         ; Port E Control register
M_PRRE EQU $FFFF9E         ; Port E Direction Register
M_PDRE EQU $FFFF9D         ; Port E Data Register
M_OGDB EQU $FFFFFC         ; OnCE GDB Register


;------------------------------------------------------------------------
;
;       EQUATES for Host Interface
;
;------------------------------------------------------------------------

;       Register Addresses

M_HCR EQU $FFFFC2           ; Host Control Register
M_HSR EQU $FFFFC3           ; Host Status Register
M_HPCR EQU $FFFFC4          ; Host Polarity Control Register
M_HBAR EQU $FFFFC5          ; Host Base Address Register
M_HRX EQU $FFFFC6           ; Host Receive Register
M_HTX EQU $FFFFC7           ; Host Transmit Register

;       HCR bits definition
M_HRIE EQU $0               ; Host Receive interrupts Enable
M_HTIE EQU $1               ; Host Transmit Interrupt Enable
M_HCIE EQU $2               ; Host Command Interrupt Enable
M_HF2 EQU $3                ; Host Flag 2
M_HF3 EQU $4                ; Host Flag 3

;       HSR bits definition
M_HRDF EQU $0               ; Host Receive Data Full
M_HTDE EQU $1               ; Host Receive Data Empty
M_HCP EQU $2                ; Host Command Pending
M_HF0 EQU $3                ; Host Flag 0
M_HF1 EQU $4                ; Host Flag 1


;       HPCR bits definition
M_HGEN EQU $0               ; Host Port GPIO Enable
M_HA8EN EQU $1              ; Host Address 8 Enable
M_HA9EN EQU $2              ; Host Address 9 Enable
M_HCSEN EQU $3              ; Host Chip Select Enable
M_HREN EQU $4               ; Host Request Enable
M_HAEN EQU $5               ; Host Acknowledge Enable
M_HEN EQU $6                ; Host Enable
M_HOD EQU $8                ; Host Request Open Drain mode
M_HDSP EQU $9               ; Host Data Strobe Polarity
M_HASP EQU $A               ; Host Address Strobe Polarity
M_HMUX EQU $B               ; Host Multiplexed bus select
M_HD_HS EQU $C              ; Host Double/Single Strobe select
```

```
M_HCSP EQU $D                ; Host Chip Select Polarity
M_HRP EQU $E                 ; Host Request Polarity
M_HAP EQU $F                 ; Host Acknowledge Polarity


;------------------------------------------------------------------------
;
;       EQUATES for Serial Communications Interface (SCI)
;
;------------------------------------------------------------------------

;       Register Addresses

M_STXH EQU $FFFF97           ; SCI Transmit Data Register (high)
M_STXM EQU $FFFF96           ; SCI Transmit Data Register (middle)
M_STXL EQU $FFFF95           ; SCI Transmit Data Register (low)
M_SRXH EQU $FFFF9A           ; SCI Receive Data Register (high)
M_SRXM EQU $FFFF99           ; SCI Receive Data Register (middle)
M_SRXL EQU $FFFF98           ; SCI Receive Data Register (low)
M_STXA EQU $FFFF94           ; SCI Transmit Address Register
M_SCR EQU $FFFF9C            ; SCI Control Register
M_SSR EQU $FFFF93            ; SCI Status Register
M_SCCR EQU $FFFF9B           ; SCI Clock Control Register

;       SCI Control Register Bit Flags

M_WDS EQU $7                 ; Word Select Mask (WDS0-WDS3)
M_WDS0 EQU 0                 ; Word Select 0
M_WDS1 EQU 1                 ; Word Select 1
M_WDS2 EQU 2                 ; Word Select 2
M_SSFTD EQU 3                ; SCI Shift Direction
M_SBK EQU 4                  ; Send Break
M_WAKE EQU 5                 ; Wakeup Mode Select
M_RWU EQU 6                  ; Receiver Wakeup Enable
M_WOMS EQU 7                 ; Wired-OR Mode Select
M_SCRE EQU 8                 ; SCI Receiver Enable
M_SCTE EQU 9                 ; SCI Transmitter Enable
M_ILIE EQU 10                ; Idle Line Interrupt Enable
M_SCRIE EQU 11               ; SCI Receive Interrupt Enable
M_SCTIE EQU 12               ; SCI Transmit Interrupt Enable
M_TMIE EQU 13                ; Timer Interrupt Enable
M_TIR EQU 14                 ; Timer Interrupt Rate
M_SCKP EQU 15                ; SCI Clock Polarity
M_REIE EQU 16                ; SCI Error Interrupt Enable (REIE)

;       SCI Status Register Bit Flags

M_TRNE EQU 0                 ; Transmitter Empty
M_TDRE EQU 1                 ; Transmit Data Register Empty
M_RDRF EQU 2                 ; Receive Data Register Full
M_IDLE EQU 3                 ; Idle Line Flag
M_OR EQU 4                   ; Overrun Error Flag
M_PE EQU 5                   ; Parity Error
```

```
M_FE EQU 6                  ; Framing Error Flag
M_R8 EQU 7                  ; Received Bit 8 (R8) Address

;       SCI Clock Control Register

M_CD EQU $FFF               ; Clock Divider Mask (CD0-CD11)
M_COD EQU 12                ; Clock Out Divider
M_SCP EQU 13                ; Clock Prescaler
M_RCM EQU 14                ; Receive Clock Mode Source Bit
M_TCM EQU 15                ; Transmit Clock Source Bit

;------------------------------------------------------------------------
;
;       EQUATES for Synchronous Serial Interface (SSI)
;
;------------------------------------------------------------------------

;
;       Register Addresses Of SSI0
M_TX00 EQU $FFFFBC          ; SSI0 Transmit Data Register 0
M_TX01 EQU $FFFFBB          ; SSIO Transmit Data Register 1
M_TX02 EQU $FFFFBA          ; SSIO Transmit Data Register 2
M_TSR0 EQU $FFFFB9          ; SSI0 Time Slot Register
M_RX0 EQU $FFFFB8           ; SSI0 Receive Data Register
M_SSISR0 EQU $FFFFB7        ; SSI0 Status Register
M_CRB0 EQU $FFFFB6          ; SSI0 Control Register B
M_CRA0 EQU $FFFFB5          ; SSI0 Control Register A
M_TSMA0 EQU $FFFFB4         ; SSI0 Transmit Slot Mask Register A
M_TSMB0 EQU $FFFFB3         ; SSI0 Transmit Slot Mask Register B
M_RSMA0 EQU $FFFFB2         ; SSI0 Receive Slot Mask Register A
M_RSMB0 EQU $FFFFB1         ; SSI0 Receive Slot Mask Register B

;       Register Addresses Of SSI1
M_TX10 EQU $FFFFAC          ; SSI1 Transmit Data Register 0
M_TX11 EQU $FFFFAB          ; SSI1 Transmit Data Register 1
M_TX12 EQU $FFFFAA          ; SSI1 Transmit Data Register 2
M_TSR1 EQU $FFFFA9          ; SSI1 Time Slot Register
M_RX1 EQU $FFFFA8           ; SSI1 Receive Data Register
M_SSISR1 EQU $FFFFA7        ; SSI1 Status Register
M_CRB1 EQU $FFFFA6          ; SSI1 Control Register B
M_CRA1 EQU $FFFFA5          ; SSI1 Control Register A
M_TSMA1 EQU $FFFFA4         ; SSI1 Transmit Slot Mask Register A
M_TSMB1 EQU $FFFFA3         ; SSI1 Transmit Slot Mask Register B
M_RSMA1 EQU $FFFFA2         ; SSI1 Receive Slot Mask Register A
M_RSMB1 EQU $FFFFA1         ; SSI1 Receive Slot Mask Register B

;       SSI Control Register A Bit Flags

M_PM EQU $FF                ; Prescale Modulus Select Mask (PM0-PM7)
M_PSR EQU 11                ; Prescaler Range
M_DC EQU $1F000             ; Frame Rate Divider Control Mask (DC0-DC7)
M_ALC EQU 18                ; Alignment Control (ALC)
M_WL EQU $380000            ; Word Length Control Mask (WL0-WL7)
```

```
M_SSC1 EQU 22                ; Select SC1 as TR #0 drive enable (SSC1)

;       SSI Control Register B Bit Flags

M_OF EQU $3                  ; Serial Output Flag Mask
M_OF0 EQU 0                  ; Serial Output Flag 0
M_OF1 EQU 1                  ; Serial Output Flag 1
M_SCD EQU $1C                ; Serial Control Direction Mask
M_SCD0 EQU 2                 ; Serial Control 0 Direction
M_SCD1 EQU 3                 ; Serial Control 1 Direction
M_SCD2 EQU 4                 ; Serial Control 2 Direction
M_SCKD EQU 5                 ; Clock Source Direction
M_SHFD EQU 6                 ; Shift Direction
M_FSL EQU $180               ; Frame Sync Length Mask (FSL0-FSL1)
M_FSL0 EQU 7                 ; Frame Sync Length 0
M_FSL1 EQU 8                 ; Frame Sync Length 1
M_FSR EQU 9                  ; Frame Sync Relative Timing
M_FSP EQU 10                 ; Frame Sync Polarity
M_CKP EQU 11                 ; Clock Polarity
M_SYN EQU 12                 ; Sync/Async Control
M_MOD EQU 13                 ; SSI Mode Select
M_SSTE EQU $1C000            ; SSI Transmit enable Mask
M_SSTE2 EQU 14               ; SSI Transmit #2 Enable
M_SSTE1 EQU 15               ; SSI Transmit #1 Enable
M_SSTE0 EQU 16               ; SSI Transmit #0 Enable
M_SSRE EQU 17                ; SSI Receive Enable
M_SSTIE EQU 18               ; SSI Transmit Interrupt Enable
M_SSRIE EQU 19               ; SSI Receive Interrupt Enable
M_STLIE EQU 20               ; SSI Transmit Last Slot Interrupt Enable
M_SRLIE EQU 21               ; SSI Receive Last Slot Interrupt Enable
M_STEIE EQU 22               ; SSI Transmit Error Interrupt Enable
M_SREIE EQU 23               ; SI Receive Error Interrupt Enable

;       SSI Status Register Bit Flags

M_IF EQU $3                  ; Serial Input Flag Mask
M_IF0 EQU 0                  ; Serial Input Flag 0
M_IF1 EQU 1                  ; Serial Input Flag 1
M_TFS EQU 2                  ; Transmit Frame Sync Flag
M_RFS EQU 3                  ; Receive Frame Sync Flag
M_TUE EQU 4                  ; Transmitter Underrun Error FLag
M_ROE EQU 5                  ; Receiver Overrun Error Flag
M_TDE EQU 6                  ; Transmit Data Register Empty
M_RDF EQU 7                  ; Receive Data Register Full

;       SSI Transmit Slot Mask Register A

M_SSTSA EQU $FFFF            ; SSI Transmit Slot Bits Mask A (TS0-TS15)

;       SSI Transmit Slot Mask Register B

M_SSTSB EQU $FFFF            ; SSI Transmit Slot Bits Mask B (TS16-TS31)
```

```
;       SSI Receive Slot Mask Register A

M_SSRSA EQU $FFFF            ; SSI Receive Slot Bits Mask A (RS0-RS15)

;       SSI Receive Slot Mask Register B

M_SSRSB EQU $FFFF            ; SSI Receive Slot Bits Mask B (RS16-RS31)



;------------------------------------------------------------------------
;
;       EQUATES for Exception Processing
;
;------------------------------------------------------------------------


;       Register Addresses

M_IPRC EQU $FFFFFF           ; Interrupt Priority Register Core
M_IPRP EQU $FFFFFE           ; Interrupt Priority Register Peripheral

;       Interrupt Priority Register Core (IPRC)

M_IAL EQU $7                 ; IRQA Mode Mask
M_IAL0 EQU 0                 ; IRQA Mode Interrupt Priority Level (low)
M_IAL1 EQU 1                 ; IRQA Mode Interrupt Priority Level (high)
M_IAL2 EQU 2                 ; IRQA Mode Trigger Mode
M_IBL EQU $38                ; IRQB Mode Mask
M_IBL0 EQU 3                 ; IRQB Mode Interrupt Priority Level (low)
M_IBL1 EQU 4                 ; IRQB Mode Interrupt Priority Level (high)
M_IBL2 EQU 5                 ; IRQB Mode Trigger Mode
M_ICL EQU $1C0               ; IRQC Mode Mask
M_ICL0 EQU 6                 ; IRQC Mode Interrupt Priority Level (low)
M_ICL1 EQU 7                 ; IRQC Mode Interrupt Priority Level (high)
M_ICL2 EQU 8                 ; IRQC Mode Trigger Mode
M_IDL EQU $E00               ; IRQD Mode Mask
M_IDL0 EQU 9                 ; IRQD Mode Interrupt Priority Level (low)
M_IDL1 EQU 10                ; IRQD Mode Interrupt Priority Level (high)
M_IDL2 EQU 11                ; IRQD Mode Trigger Mode
M_D0L EQU $3000              ; DMA0 Interrupt priority Level Mask
M_D0L0 EQU 12                ; DMA0 Interrupt Priority Level (low)
M_D0L1 EQU 13                ; DMA0 Interrupt Priority Level (high)
M_D1L EQU $C000              ; DMA1 Interrupt Priority Level Mask
M_D1L0 EQU 14                ; DMA1 Interrupt Priority Level (low)
M_D1L1 EQU 15                ; DMA1 Interrupt Priority Level (high)
M_D2L EQU $30000             ; DMA2 Interrupt priority Level Mask
M_D2L0 EQU 16                ; DMA2 Interrupt Priority Level (low)
M_D2L1 EQU 17                ; DMA2 Interrupt Priority Level (high)
M_D3L EQU $C0000             ; DMA3 Interrupt Priority Level Mask
M_D3L0 EQU 18                ; DMA3 Interrupt Priority Level (low)
M_D3L1 EQU 19                ; DMA3 Interrupt Priority Level (high)
M_D4L EQU $300000            ; DMA4 Interrupt priority Level Mask
```

```
M_D4L0 EQU 20               ; DMA4 Interrupt Priority Level (low)
M_D4L1 EQU 21               ; DMA4 Interrupt Priority Level (high)
M_D5L EQU $C00000           ; DMA5 Interrupt priority Level Mask
M_D5L0 EQU 22               ; DMA5 Interrupt Priority Level (low)
M_D5L1 EQU 23               ; DMA5 Interrupt Priority Level (high)


;       Interrupt Priority Register Peripheral (IPRP)

M_HPL EQU $3                ; Host Interrupt Priority Level Mask
M_HPL0 EQU 0                ; Host Interrupt Priority Level (low)
M_HPL1 EQU 1                ; Host Interrupt Priority Level (high)
M_S0L EQU $C                ; SSI0 Interrupt Priority Level Mask
M_S0L0 EQU 2                ; SSI0 Interrupt Priority Level (low)
M_S0L1 EQU 3                ; SSI0 Interrupt Priority Level (high)
M_S1L EQU $30               ; SSI1 Interrupt Priority Level Mask
M_S1L0 EQU 4                ; SSI1 Interrupt Priority Level (low)
M_S1L1 EQU 5                ; SSI1 Interrupt Priority Level (high)
M_SCL EQU $C0               ; SCI  Interrupt Priority Level  Mask
M_SCL0 EQU 6                ; SCI  Interrupt Priority Level  (low)
M_SCL1 EQU 7                ; SCI  Interrupt Priority Level  (high)
M_T0L EQU $300              ; TIMER Interrupt Priority Level Mask
M_T0L0 EQU 8                ; TIMER Interrupt Priority Level (low)
M_T0L1 EQU 9                ; TIMER Interrupt Priority Level (high)


;------------------------------------------------------------------------
;
;       EQUATES for TIMER
;
;------------------------------------------------------------------------

;       Register Addresses Of TIMER0

M_TCSR0 EQU $FFFF8F         ; Timer 0 Control/Status Register
M_TLR0 EQU  $FFFF8E         ; TIMER0 Load Reg
M_TCPR0 EQU $FFFF8D         ; TIMER0 Compare Register
M_TCR0 EQU  $FFFF8C         ; TIMER0 Count Register

;       Register Addresses Of TIMER1

M_TCSR1 EQU $FFFF8B         ; TIMER1 Control/Status Register
M_TLR1 EQU  $FFFF8A         ; TIMER1 Load Reg
M_TCPR1 EQU $FFFF89         ; TIMER1 Compare Register
M_TCR1 EQU  $FFFF88         ; TIMER1 Count Register


;       Register Addresses Of TIMER2

M_TCSR2 EQU $FFFF87         ; TIMER2 Control/Status Register
M_TLR2 EQU  $FFFF86         ; TIMER2 Load Reg
M_TCPR2 EQU $FFFF85         ; TIMER2 Compare Register
M_TCR2 EQU  $FFFF84         ; TIMER2 Count Register
```

```
M_TPLR EQU  $FFFF83         ; TIMER Prescaler Load Register
M_TPCR EQU  $FFFF82         ; TIMER Prescalar Count Register


;      Timer Control/Status Register Bit Flags

M_TE EQU 0                  ; Timer Enable
M_TOIE EQU 1                ; Timer Overflow Interrupt Enable
M_TCIE EQU 2                ; Timer Compare Interrupt Enable
M_TC EQU $F0                ; Timer Control Mask (TC0-TC3)
M_INV EQU 8                 ; Inverter Bit
M_TRM EQU 9                 ; Timer Restart Mode
M_DIR EQU 11                ; Direction Bit
M_DI EQU 12                 ; Data Input
M_DO EQU 13                 ; Data Output
M_PCE EQU  15               ; Prescaled Clock Enable
M_TOF EQU 20                ; Timer Overflow Flag
M_TCF EQU 21                ; Timer Compare Flag

;       Timer Prescaler Register Bit Flags

M_PS EQU  $600000           ; Prescaler Source Mask
M_PS0 EQU  21
M_PS1 EQU  22

;         Timer Control Bits
M_TC0 EQU 4                 ; Timer Control 0
M_TC1 EQU 5                 ; Timer Control 1
M_TC2 EQU 6                 ; Timer Control 2
M_TC3 EQU 7                 ; Timer Control 3


;------------------------------------------------------------------------
;
;       EQUATES for Direct Memory Access (DMA)
;
;------------------------------------------------------------------------

;        Register Addresses Of DMA
M_DSTR EQU FFFFF4                    ; DMA Status Register
M_DOR0 EQU $FFFFF3 ; DMA Offset Register 0
M_DOR1 EQU $FFFFF2 ; DMA Offset Register 1
M_DOR2 EQU $FFFFF1 ; DMA Offset Register 2
M_DOR3 EQU $FFFFF0 ; DMA Offset Register 3


;        Register Addresses Of DMA0

M_DSR0 EQU $FFFFEF ; DMA0 Source Address Register
M_DDR0 EQU $FFFFEE ; DMA0 Destination Address Register
M_DCO0 EQU $FFFFED ; DMA0 Counter
M_DCR0 EQU $FFFFEC ; DMA0 Control Register
```

```
;       Register Addresses Of DMA1

M_DSR1 EQU $FFFFEB ; DMA1 Source Address Register
M_DDR1 EQU $FFFFEA ; DMA1 Destination Address Register
M_DCO1 EQU $FFFFE9 ; DMA1 Counter
M_DCR1 EQU $FFFFE8 ; DMA1 Control Register

;       Register Addresses Of DMA2

M_DSR2 EQU $FFFFE7 ; DMA2 Source Address Register
M_DDR2 EQU $FFFFE6 ; DMA2 Destination Address Register
M_DCO2 EQU $FFFFE5 ; DMA2 Counter
M_DCR2 EQU $FFFFE4 ; DMA2 Control Register

;       Register Addresses Of DMA4

M_DSR3 EQU $FFFFE3 ; DMA3 Source Address Register
M_DDR3 EQU $FFFFE2 ; DMA3 Destination Address Register
M_DCO3 EQU $FFFFE1 ; DMA3 Counter
M_DCR3 EQU $FFFFE0 ; DMA3 Control Register

;       Register Addresses Of DMA4


M_DSR4 EQU $FFFFDF ; DMA4 Source Address Register
M_DDR4 EQU $FFFFDE ; DMA4 Destination Address Register
M_DCO4 EQU $FFFFDD ; DMA4 Counter
M_DCR4 EQU $FFFFDC ; DMA4 Control Register

;       Register Addresses Of DMA5

M_DSR5 EQU $FFFFDB ; DMA5 Source Address Register
M_DDR5 EQU $FFFFDA ; DMA5 Destination Address Register
M_DCO5 EQU $FFFFD9 ; DMA5 Counter
M_DCR5 EQU $FFFFD8 ; DMA5 Control Register

;         DMA Control Register

M_DSS EQU $3       ; DMA Source Space Mask (DSS0-Dss1)
M_DSS0 EQU 0       ; DMA Source Memory space 0
M_DSS1 EQU 1       ; DMA Source Memory space 1
M_DDS EQU $C       ; DMA Destination Space Mask (DDS-DDS1)
M_DDS0 EQU 2       ; DMA Destination Memory Space 0
M_DDS1 EQU 3       ; DMA Destination Memory Space 1
M_DAM EQU $3f0     ; DMA Address Mode Mask (DAM5-DAM0)
M_DAM0  EQU 4      ; DMA Address Mode 0
M_DAM1  EQU 5      ; DMA Address Mode 1
M_DAM2  EQU 6      ; DMA Address Mode 2
M_DAM3  EQU 7      ; DMA Address Mode 3
M_DAM4  EQU 8      ; DMA Address Mode 4
M_DAM5  EQU 9      ; DMA Address Mode 5
M_D3D EQU 10       ; DMA Three Dimensional Mode
M_DRS EQU $F800    ; DMA Request Source Mask (DRS0-DRS4)
```

```
M_DCON EQU 16      ; DMA Continuous Mode
M_DPR EQU $60000   ; DMA Channel Priority
M_DPR0 EQU 17      ; DMA Channel Priority Level (low)
M_DPR1 EQU 18      ; DMA Channel Priority Level (high)
M_DTM EQU $380000  ; DMA Transfer Mode Mask (DTM2-DTM0)
M_DTM0 EQU 19      ; DMA Transfer Mode 0
M_DTM1 EQU 20      ; DMA Transfer Mode 1
M_DTM2 EQU 21      ; DMA Transfer Mode 2
M_DIE EQU 22       ; DMA Interrupt Enable bit
M_DE EQU 23        ; DMA Channel Enable bit

;       DMA Status Register

M_DTD EQU $3F      ; Channel Transfer Done Status MASK (DTD0-DTD5)
M_DTD0 EQU 0       ; DMA Channel Transfer Done Status 0
M_DTD1 EQU 1       ; DMA Channel Transfer Done Status 1
M_DTD2 EQU 2       ; DMA Channel Transfer Done Status 2
M_DTD3 EQU 3       ; DMA Channel Transfer Done Status 3
M_DTD4 EQU 4       ; DMA Channel Transfer Done Status 4
M_DTD5 EQU 5       ; DMA Channel Transfer Done Status 5
M_DACT EQU  8      ; DMA Active State
M_DCH EQU  $E00    ; DMA Active Channel Mask (DCH0-DCH2)
M_DCH0 EQU  9      ; DMA Active Channel 0
M_DCH1 EQU  10     ; DMA Active Channel 1
M_DCH2 EQU  11     ; DMA Active Channel 2

;-----------------------------------------------------------------------
;
;       EQUATES for Enhanced Filter Co-Processor (EFCOP)
;
;-----------------------------------------------------------------------

M_FDIR   EQU     $FFFFB0         ; EFCOP Data Input Register
M_FDOR   EQU     $FFFFB1         ; EFCOP Data Output Register
M_FKIR   EQU     $FFFFB2         ; EFCOP K-Constant Register
M_FCNT   EQU     $FFFFB3         ; EFCOP Filter Counter
M_FCSR   EQU     $FFFFB4         ; EFCOP Control Status Register
M_FACR   EQU     $FFFFB5         ; EFCOP ALU Control Register
M_FDBA   EQU     $FFFFB6         ; EFCOP Data Base Address
M_FCBA   EQU     $FFFFB7         ; EFCOP Coefficient Base Address
M_FDCH   EQU     $FFFFB8         ; EFCOP Decimation/Channel Register



;-----------------------------------------------------------------------
;
;       EQUATES for Phase Locked Loop (PLL)
;
;-----------------------------------------------------------------------

;       Register Addresses Of PLL

M_PCTL EQU $FFFFFD         ; PLL Control Register
```

```
;       PLL Control Register

M_MF EQU $FFF      : Multiplication Factor Bits Mask (MF0-MF11)
M_DF EQU $7000     ; Division Factor Bits Mask (DF0-DF2)
M_XTLR EQU 15      ; XTAL Range select bit
M_XTLD EQU 16      ; XTAL Disable Bit
M_PSTP EQU 17      ; STOP Processing State Bit
M_PEN EQU 18       ; PLL Enable Bit
M_PCOD EQU 19      ; PLL Clock Output Disable Bit
M_PD EQU $F00000   ; PreDivider Factor Bits Mask (PD0-PD3)



;------------------------------------------------------------------------
;
;       EQUATES for BIU
;
;------------------------------------------------------------------------

;       Register Addresses Of BIU


M_BCR EQU $FFFFFB  ; Bus Control Register
M_DCR EQU $FFFFFA  ; DRAM Control Register
M_AAR0 EQU $FFFFF9 ; Address Attribute Register 0
M_AAR1 EQU $FFFFF8 ; Address Attribute Register 1
M_AAR2 EQU $FFFFF7 ; Address Attribute Register 2
M_AAR3 EQU $FFFFF6 ; Address Attribute Register 3
M_IDR EQU  $FFFFF5 ; ID Register

;       Bus Control Register

M_BA0W EQU $1F     ; Area 0 Wait Control Mask (BA0W0-BA0W4)
M_BA1W EQU $3E0    ; Area 1 Wait Control Mask (BA1W0-BA14)
M_BA2W EQU $1C00   ; Area 2 Wait Control Mask (BA2W0-BA2W2)
M_BA3W EQU $E000   ; Area 3 Wait Control Mask (BA3W0-BA3W3)
M_BDFW EQU $1F0000 ; Default Area Wait Control Mask (BDFW0-BDFW4)
M_BBS EQU 21       ; Bus State
M_BLH EQU 22       ; Bus Lock Hold
M_BRH EQU 23       ; Bus Request Hold

;       DRAM Control Register

M_BCW EQU $3       ; In Page Wait States Bits Mask (BCW0-BCW1)
M_BRW EQU $C       ; Out Of Page Wait States Bits Mask (BRW0-BRW1)
M_BPS EQU $300     ; DRAM Page Size Bits Mask (BPS0-BPS1)
M_BPLE EQU 11      ; Page Logic Enable
M_BME EQU 12       ; Mastership Enable
M_BRE EQU 13       ; Refresh Enable
M_BSTR EQU 14      ; Software Triggered Refresh
M_BRF EQU $7F8000  ; Refresh Rate Bits Mask (BRF0-BRF7)
M_BRP EQU 23       ; Refresh prescaler
```

```
;       Address Attribute Registers

M_BAT EQU $3       ; Ext. Access Type and Pin Def. Bits Mask (BAT0-BAT1)
M_BAAP EQU 2       ; Address Attribute Pin Polarity
M_BPEN EQU 3       ; Program Space Enable
M_BXEN EQU 4       ; X Data Space Enable
M_BYEN EQU 5       ; Y Data Space Enable
M_BAM EQU 6        ; Address Muxing
M_BPAC EQU 7       ; Packing Enable
M_BNC EQU $F00     ; Number of Address Bits to Compare Mask (BNC0-BNC3)
M_BAC EQU $FFF000  ; Address to Compare Bits Mask (BAC0-BAC11)


;       control and status bits in SR

M_CP EQU $c00000   ; mask for CORE-DMA priority bits in SR
M_CA EQU 0         ; Carry
M_V EQU 1          ; Overflow
M_Z EQU 2          ; Zero
M_N EQU 3          ; Negative
M_U EQU 4          ; Unnormalized
M_E EQU 5          ; Extension
M_L EQU 6          ; Limit
M_S EQU 7          ; Scaling Bit
M_I0 EQU 8         ; Interupt Mask Bit 0
M_I1 EQU 9         ; Interupt Mask Bit 1
M_S0 EQU 10        ; Scaling Mode Bit 0
M_S1 EQU 11        ; Scaling Mode Bit 1
M_SC EQU 13        ; Sixteen_Bit Compatibility
M_DM EQU 14        ; Double Precision Multiply
M_LF EQU 15        ; DO-Loop Flag
M_FV EQU 16        ; DO-Forever Flag
M_SA EQU 17        ; Sixteen-Bit Arithmetic
M_CE EQU 19        ; Instruction Cache Enable
M_SM EQU 20        ; Arithmetic Saturation
M_RM EQU 21        ; Rounding Mode
M_CP0 EQU 22       ; bit 0 of priority bits in SR
M_CP1 EQU 23       ; bit 1 of priority bits in SR

;       control and status bits in OMR
M_CDP EQU $300     ; mask for CORE-DMA priority bits in OMR
M_MA    equ0       ; Operating Mode A
M_MB   equ1        ; Operating Mode B
M_MC    equ2       ; Operating Mode C
M_MD    equ3       ; Operating Mode D
M_EBD EQU 4        ; External Bus Disable bit in OMR
M_SD EQU 6         ; Stop Delay
M_MS EQU 7         ; Memory Switch bit in OMR
M_CDP0 EQU 8       ; bit 0 of priority bits in OMR
M_CDP1 EQU 9       ; bit 1 of priority bits in OMR
M_BEN   EQU 10     ; Burst Enable
M_TAS  EQU 11      ; TA Synchronize Select
```

```
M_BRT  EQU 12      ; Bus Release Timing
M_ATE EQU 15       ; Address Tracing Enable bit in OMR.
M_XYS EQU 16       ; Stack Extension space select bit in OMR.
M_EUN EQU 17       ; Extensed stack UNderflow flag in OMR.
M_EOV EQU 18       ; Extended stack OVerflow flag in OMR.
M_WRP EQU 19       ; Extended WRaP flag in OMR.
M_SEN EQU 20       ; Stack Extension Enable bit in OMR.





;**************************************************************************
;
;     EQUATES for DSP56311 interrupts
;
;     Last update: June 11 1995
;
;**************************************************************************

         page      132,55,0,0,0
         opt       mex

intequ  ident   1,0

         if        @DEF(I_VEC)
         ;leave user definition as is.
         else
I_VEC EQU $0
         endif

;------------------------------------------------------------------------
; Non-Maskable interrupts
;------------------------------------------------------------------------
I_RESET EQU I_VEC+$00       ; Hardware RESET
I_STACK EQU I_VEC+$02       ; Stack Error
I_ILL EQU I_VEC+$04         ; Illegal Instruction
I_DBG EQU I_VEC+$06         ; Debug Request
I_TRAP EQU I_VEC+$08        ; Trap
I_NMI EQU I_VEC+$0A         ; Non Maskable Interrupt

;------------------------------------------------------------------------
; Interrupt Request Pins
;------------------------------------------------------------------------
I_IRQA EQU I_VEC+$10        ; IRQA
I_IRQB EQU I_VEC+$12        ; IRQB
I_IRQC EQU I_VEC+$14        ; IRQC
I_IRQD EQU I_VEC+$16        ; IRQD

;------------------------------------------------------------------------
; DMA Interrupts
```

```
;-----------------------------------------------------------------------
I_DMA0 EQU I_VEC+$18        ; DMA Channel 0
I_DMA1 EQU I_VEC+$1A        ; DMA Channel 1
I_DMA2 EQU I_VEC+$1C        ; DMA Channel 2
I_DMA3 EQU I_VEC+$1E        ; DMA Channel 3
I_DMA4 EQU I_VEC+$20        ; DMA Channel 4
I_DMA5 EQU I_VEC+$22        ; DMA Channel 5

;-----------------------------------------------------------------------
; Timer Interrupts
;-----------------------------------------------------------------------
I_TIM0C EQU I_VEC+$24       ; TIMER 0 compare
I_TIM0OF EQU I_VEC+$26      ; TIMER 0 overflow
I_TIM1C EQU I_VEC+$28       ; TIMER 1 compare
I_TIM1OF EQU I_VEC+$2A      ; TIMER 1 overflow
I_TIM2C EQU I_VEC+$2C       ; TIMER 2 compare
I_TIM2OF EQU I_VEC+$2E      ; TIMER 2 overflow

;-----------------------------------------------------------------------
; ESSI Interrupts
;-----------------------------------------------------------------------
I_SI0RD EQU I_VEC+$30       ; ESSI0 Receive Data
I_SI0RDE EQU I_VEC+$32      ; ESSI0 Receive Data w/ exception Status
I_SI0RLS EQU I_VEC+$34      ; ESSI0 Receive last slot
I_SI0TD EQU I_VEC+$36       ; ESSI0 Transmit data
I_SI0TDE EQU I_VEC+$38      ; ESSI0 Transmit Data w/ exception Status
I_SI0TLS EQU I_VEC+$3A      ; ESSI0 Transmit last slot
I_SI1RD EQU I_VEC+$40       ; ESSI1 Receive Data
I_SI1RDE EQU I_VEC+$42      ; ESSI1 Receive Data w/ exception Status
I_SI1RLS EQU I_VEC+$44      ; ESSI1 Receive last slot
I_SI1TD EQU I_VEC+$46       ; ESSI1 Transmit data
I_SI1TDE EQU I_VEC+$48      ; ESSI1 Transmit Data w/ exception Status
I_SI1TLS EQU I_VEC+$4A      ; ESSI1 Transmit last slot

;-----------------------------------------------------------------------
; SCI Interrupts
;-----------------------------------------------------------------------
I_SCIRD EQU I_VEC+$50       ; SCI Receive Data
I_SCIRDE EQU I_VEC+$52      ; SCI Receive Data With Exception Status
I_SCITD EQU I_VEC+$54       ; SCI Transmit Data
I_SCIIL EQU I_VEC+$56       ; SCI Idle Line
I_SCITM EQU I_VEC+$58       ; SCI Timer

;-----------------------------------------------------------------------
; HOST Interrupts
;-----------------------------------------------------------------------
I_HRDF EQU I_VEC+$60        ; Host Receive Data Full
I_HTDE EQU I_VEC+$62        ; Host Transmit Data Empty
I_HC EQU I_VEC+$64          ; Default Host Command
;----------------------------------------------------------------------
; EFCOP Filter Interrupts
;----------------------------------------------------------------------
```

```
I_FDIIE  EQU     I_VEC+$68 ; EFilter input buffer empty
I_FDOIE  EQU     I_VEC+$6A ; EFilter output buffer full

;------------------------------------------------------------------------
; INTERRUPT ENDING ADDRESS
;------------------------------------------------------------------------
I_INTEND EQU I_VEC+$FF      ; last address of interrupt vector space
```

NOTES:

NOTES:

NOTES:

INDEX

Order Number: DSP56311/D
Rev. 0, 10/1999

ORDERING INFORMATION

Consult a Motorola Semiconductor sales office or authorized distributor to determine product availability and to place an order.

Part	Supply Voltage	Package Type	Pin Count	Frequency (MHz)	Order Number
DSP56311	1.8 V core 3.3 V I/O	Plastic Ball Grid Array (PBGA)	196	150	XC56311GC150

Motorola and Ⓜ are registered trademarks of Motorola, Inc.
OnCE and Mfax are trademarks of Motorola, Inc.

Motorola reserves the right to make changes without further notice to any products herein. Motorola makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Motorola assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters which may be provided in Motorola data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including “Typicals” must be validated for each customer application by customer’s technical experts. Motorola does not convey any license under its patent rights nor the rights of others. Motorola products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, or other applications intended to support life, or for any other application in which the failure of the Motorola product could create a situation where personal injury or death may occur. Should Buyer purchase or use Motorola products for any such unintended or unauthorized application, Buyer shall indemnify and hold Motorola and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that Motorola was negligent regarding the design or manufacture of the part. Motorola and Ⓜ are registered trademarks of Motorola, Inc. Motorola, Inc. is an Equal Opportunity/Affirmative Action Employer.

How to reach us:

USA/Europe/Locations Not Listed:
Motorola Literature Distribution
P.O. Box 5405
Denver, Colorado 80217
1 (800) 441-2447
1 (303) 675-2140

Motorola Fax Back System (Mfax™):
RMFAX0@email.sps.mot.com
TOUCHTONE (602) 244-6609
USA and Canada ONLY:
1-800-774-1848
http://sps.motorola.com/mfax/

Asia/Pacific:
Motorola Semiconductors H.K. Ltd.
8B Tai Ping Industrial Park
51 Ting Kok Road
Tai Po, N.T., Hong Kong
852-26629298

Technical Resource Center:
1 (800) 521-6274

DSP Helpline
dsphelp@dsp.sps.mot.com

Japan:
Nippon Motorola Ltd.
SPD, Strategic Planning Office141
4-32-1, Nishi-Gotanda
3-14-2 Tatsumi Koto-Ku
Shinagawa-ku, Tokyo, Japan
81-3-5487-8488

Customer Focus Center:
1-800-521-6274

Internet:
http://www.mot.com/SPS/DSP/