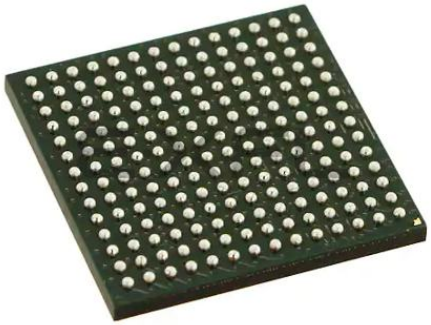


Welcome to E-XFL.COM

Understanding Embedded - DSP (Digital Signal Processors)

Embedded - DSP (Digital Signal Processors) are specialized microprocessors designed to perform complex mathematical computations on digital signals in real-time. Unlike general-purpose processors, DSPs are optimized for high-speed numeric processing tasks, making them ideal for applications that require efficient and precise manipulation of digital data. These processors are fundamental in converting and processing signals in various forms, including audio, video, and communication signals, ensuring that data is accurately interpreted and utilized in embedded systems.

Applications of Embedded - DSP (Digital Signal Processors)

Details	
Product Status	Obsolete
Type	Fixed Point
Interface	Host Interface, SSI, SCI
Clock Rate	150MHz
Non-Volatile Memory	ROM (576B)
On-Chip RAM	384kB
Voltage - I/O	3.30V
Voltage - Core	1.80V
Operating Temperature	-40°C ~ 100°C (TJ)
Mounting Type	Surface Mount
Package / Case	196-LBGA
Supplier Device Package	196-LBGA (15x15)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/dsp56311vl150

Signals/Connections 1

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)		66
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.
5. There are 5 signal connections that are not used. These are designated as no connect (NC) in the package description (see **Chapter 3**).

Note: The Clock Output (CLKOUT), BCLK, $\overline{\text{BCLK}}$, $\overline{\text{CAS}}$, and $\overline{\text{RAS[0–3]}}$ signals used by other DSP56300 family members are supported by the DSP56311 at operating frequencies up to 100 MHz. Therefore, above 100 MHz, you must enable bus arbitration by setting the Asynchronous Bus Arbitration Enable Bit (ABE) in the operating mode register. When set, the ABE bit eliminates the required set-up and hold times for $\overline{\text{BB}}$ and $\overline{\text{BG}}$ with respect to $\overline{\text{CLKOUT}}$. In addition, DRAM access is not supported above 100 MHz.

1.4 PLL

Table 1-5. Phase-Locked Loop Signals

Signal Name	Type	State During Reset	Signal Description
CLKOUT	Output	Chip-driven	**Clock Output—**Provides an output clock synchronized to the internal core clock phase. If the PLL is enabled and both the multiplication and division factors equal one, then CLKOUT is also synchronized to EXTAL. If the PLL is disabled, the CLKOUT frequency is half the frequency of EXTAL. **Note:** At operating frequencies above 100 MHz, this signal produces a low-amplitude waveform that is not usable externally by other devices. Above 100 MHz, you can use the asynchronous bus arbitration option that is enabled by the Asynchronous Bus Arbitration Enable (ABE) bit in the Operating Mode Register. When set, the DSP enters the Asynchronous Arbitration mode, which eliminates the $\overline{BB}$ and $\overline{BG}$ set-up and hold time requirements with respect to CLKOUT.
PCAP	Input	Input	**PLL Capacitor**—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 can be tied to V_{CC}, GND, or left floating.
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.

1.5 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: A[0–17], D[0–23], AA[0–3], $\overline{RD}$, $\overline{WR}$, $\overline{BB}$.

1.5.1 External Address Bus

Table 1-6. External Address Bus Signals

Signal Name	Type	State During Reset, Stop, or Wait	Signal Description
A[0–17]	Output	Tri-stated	**Address Bus**—When the DSP is the bus master, A[0–17] 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, A[0–17] do not change state when external memory spaces are not being accessed.

1.7 Host Interface (HI08)

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

1.7.1 Host Port Usage Considerations

Careful synchronization is required when the system reads multiple-bit registers that are written by another asynchronous system. This is a common problem when two asynchronous systems are connected (as they are in the Host port). The considerations for proper operation are discussed in **Table 1-10**.

Table 1-10. Host Port Usage Considerations

Action	Description
Asynchronous read of receive byte registers	When reading the receive byte registers, Receive register High (RXH), Receive register Middle (RXM), or Receive register Low (RXL), the host interface programmer should use interrupts or poll the Receive register Data Full (RXDF) flag that indicates data is available. This assures that the data in the receive byte registers is valid.
Asynchronous write to transmit byte registers	The host interface programmer should not write to the transmit byte registers, Transmit register High (TXH), Transmit register Middle (TXM), or Transmit register Low (TXL), unless the Transmit register Data Empty (TXDE) bit is set indicating that the transmit byte registers are empty. This guarantees that the transmit byte registers transfer valid data to the Host Receive (HRX) register.
Asynchronous write to host vector	The host interface programmer must change the Host Vector (HV) register only when the Host Command bit (HC) is clear. This practice guarantees that the DSP interrupt control logic receives a stable vector.

1.7.2 Host Port Configuration

HI08 signal functions vary according to the programmed configuration of the interface as determined by the 16 bits in the HI08 Port Control Register.

Table 1-11. Host Interface

Signal Name	Type	State During Reset[1,2]	Signal Description
H[0–7]	Input/Output	Ignored Input	**Host Data—**When the HI08 is programmed to interface with a non-multiplexed host bus and the HI function is selected, these signals are lines 0–7 of the bidirectional Data bus.
HAD[0–7]	Input/Output		**Host Address—**When the HI08 is programmed to interface with a multiplexed host bus and the HI function is selected, these signals are lines 0–7 of the bidirectional multiplexed Address/Data bus.
PB[0–7]	Input or Output		**Port B 0–7—**When the HI08 is configured as GPIO through the HI08 Port Control Register, these signals are individually programmed as inputs or outputs through the HI08 Data Direction Register.

Table 1-13. Enhanced Serial Synchronous Interface 1 (Continued)

Signal Name	Type	State During Reset[1,2]	Signal Description
SCK1	Input/Output	Ignored Input	**Serial Clock**—Provides 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 the Port D Direction Register. The signal can be configured as an ESSI signal SCK1 through the Port D Control Register.
SRD1	Input	Ignored Input	**Serial Receive Data**—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 the Port D Direction Register. The signal can be configured as an ESSI signal SRD1 through the Port D Control Register.
STD1	Output	Ignored Input	**Serial Transmit Data**—Transmits 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 the Port D Direction Register. The signal can be configured as an ESSI signal STD1 through the Port D Control Register.

Notes: 1. In the Stop state, the signal maintains the last state as follows:
- If the last state is input, the signal is an ignored input.
- If the last state is output, these lines have weak keepers that maintain the last output state even if the drivers are tri-stated.

2. The Wait processing state does not affect the signal state.

1.10 Serial Communication Interface (SCI)

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

Table 1-14. Serial Communication Interface

Signal Name	Type	State During Reset[1,2]	Signal Description
RXD	Input	Ignored Input	**Serial Receive Data**—Receives byte-oriented serial data and transfers it to the SCI Receive Shift Register.
PE0	Input or Output		**Port E 0**—The default configuration following reset is GPIO input PE0. When configured as PE0, signal direction is controlled through the Port E Direction Register. The signal can be configured as an SCI signal RXD through the Port E Control Register.
TXD	Output	Ignored Input	**Serial Transmit Data**—Transmits data from the SCI Transmit Data Register.
PE1	Input or Output		**Port E 1**—The default configuration following reset is GPIO input PE1. When configured as PE1, signal direction is controlled through the Port E Direction Register. The signal can be configured as an SCI signal TXD through the Port E Control Register.

Table 1-14. Serial Communication Interface (Continued)

Signal Name	Type	State During Reset[1,2]	Signal Description
SCLK	Input/Output	Ignored Input	**Serial Clock**—Provides the input or output clock used by the transmitter and/or the receiver.
PE2	Input or Output		**Port E 2**—The default configuration following reset is GPIO input PE2. When configured as PE2, signal direction is controlled through the Port E Direction Register. The signal can be configured as an SCI signal SCLK through the Port E Control Register.

Notes: 1. In the Stop state, the signal maintains the last state as follows:
- If the last state is input, the signal is an ignored input.
- If the last state is output, these lines have weak keepers that maintain the last output state even if the drivers are tri-stated.

2. The Wait processing state does not affect the signal state.

1.11 Timers

The DSP56311 has three identical and independent timers. 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 1-15. Triple Timer Signals

Signal Name	Type	State During Reset[1,2]	Signal Description
TIO0	Input or Output	Ignored 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. TIO0 can be changed to output or configured as a timer I/O through the Timer 0 Control/Status Register (TCSR0).
TIO1	Input or Output	Ignored 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. TIO1 can be changed to output or configured as a timer I/O through the Timer 1 Control/Status Register (TCSR1).
TIO2	Input or Output	Ignored 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. TIO2 can be changed to output or configured as a timer I/O through the Timer 2 Control/Status Register (TCSR2).

Notes: 1. In the Stop state, the signal maintains the last state as follows:
- If the last state is input, the signal is an ignored input.
- If the last state is output, these lines have weak keepers that maintain the last output state even if the drivers are tri-stated.

2. The Wait processing state does not affect the signal state.

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	 ∞ 157.0 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	 ∞ 157.0 µs
4	EXTAL cycle time[2] • With PLL disabled • With PLL enabled	ET_C	 6.67 ns 6.67 ns	 ∞ 273.1 µs
5	Internal clock change from EXTAL fall with PLL disabled		4.3 ns	11.0 ns
6	a.Internal clock rising edge from EXTAL rising edge with PLL enabled (MF = 1 or 2 or 4, PDF = 1, Ef > 15 MHz)[3,5] b. Internal clock falling edge from EXTAL falling edge with PLL enabled (MF ≤4, PDF ≠ 1, Ef / PDF > 15 MHz)[3,5]		0.0 ns 0.0 ns	1.8 ns 1.8 ns
7	Instruction cycle time = I_{CYC} = T_C[4] (see **Figure 2-4**) (46.7%–53.3% duty cycle) • With PLL disabled • With PLL enabled	I_{CYC}	 13.33 ns 6.7 ns	 ∞ 8.53 µs

Notes:
1. Measured at 50 percent of the input transition.
2. The maximum value for PLL enabled is given for minimum VCO frequency (see **Table 2-4**) and maximum MF.
3. Periodically sampled and not 100 percent tested.
4. The maximum value for PLL enabled is given for minimum VCO frequency 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, when a lower clock frequency is used, the signal symmetry may vary from the specified duty cycle as long as the minimum high time and low time requirements are met.

2.4.3 Phase Lock Loop (PLL) Characteristics

Table 2-6. PLL Characteristics

Characteristics	150 MHz		Unit
	Min	Max	
Voltage Controlled Oscillator (VCO) frequency when PLL enabled (MF × E_f × 2/PDF)	30	300	MHz
PLL external capacitor (PCAP pin to V_{CCP}) (C_{PCAP}[1]) • @ MF ≤4 • @ MF > 4	 (580 × MF) –100 830 × MF	 (780 × MF) –140 1470 × MF	 pF pF

Note: C_{PCAP} is the value of the PLL capacitor (connected between the PCAP pin and V_{CCP}) computed using the appropriate expression listed above.

2.4.4 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]	Minimum:			
	• Power on, external clock generator, PLL disabled	$50 \times ET_C$	333.3	—	ns
	• Power on, external clock generator, PLL enabled	$1000 \times ET_C$	6.67	—	µs
	• Power on, internal oscillator	$75000 \times ET_C$	0.50	—	ms
	• During STOP, XTAL disabled (PCTL Bit 16 = 0)	$75000 \times ET_C$	0.50	—	ms
	• During STOP, XTAL enabled (PCTL Bit 16 = 1)	$2.5 \times T_C$	16.7	—	ns
	• During normal operation	$2.5 \times T_C$	16.7	—	ns
10	Delay from asynchronous $\overline{\text{RESET}}$ deassertion to first external address output (internal reset deassertion)[5]				
	• Minimum	$3.25 \times T_C + 2.0$	23.7	—	ns
	• Maximum	$20.25 \times T_C + 10$	—	145.0	ns
13	Mode select set-up 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	Minimum:			
	• Caused by first interrupt instruction fetch	$4.25 \times T_C + 2.0$	30.4	—	ns
	• Caused by first interrupt instruction execution	$7.25 \times T_C + 2.0$	51.0	—	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	Minimum: $10 \times T_C + 5.0$	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]	Maximum: $(WS + 3.75) \times T_C - 10.94$	—	Note 8	ns
20	Delay from $\overline{\text{RD}}$ assertion to interrupt request deassertion for level sensitive fast interrupts[1, 7, 8]	Maximum: $(WS + 3.25) \times T_C - 10.94$	—	Note 8	ns
21	Delay from $\overline{\text{WR}}$ assertion to interrupt request deassertion for level sensitive fast interrupts[1, 7, 8]	Maximum:			
	• DRAM for all WS	$(WS + 3.5) \times T_C - 10.94$	—	Note 8	ns
	• SRAM WS = 1	$(WS + 3.5) \times T_C - 10.94$	—	Note 8	ns
	• SRAM WS = 2, 3	$(WS + 3) \times T_C - 10.94$	—	Note 8	ns
	• SRAM WS ≥ 4	$(WS + 2.5) \times T_C - 10.94$	—	Note 8	ns
24	Duration for $\overline{\text{IRQA}}$ assertion to recover from Stop state		5.9	—	ns
25	Delay from $\overline{\text{IRQA}}$ 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 (Operating Mode Register Bit 6 = 0)	$PLC \times ET_C \times PDF + (128\text{ K} - PLC/2) \times T_C$	1.3	9.1	ms
	• PLL is not active during Stop (PCTL Bit 17 = 0) and Stop delay is not enabled (Operating Mode Register Bit 6 = 1)	$PLC \times ET_C \times PDF + (23.75 \pm 0.5) \times T_C$	232.5 ns	12.3 ms	
	• PLL is active during Stop (PCTL Bit 17 = 1) (Implies No Stop Delay)	$(8.25 \pm 0.5) \times T_C$	51.7	58.3	ns

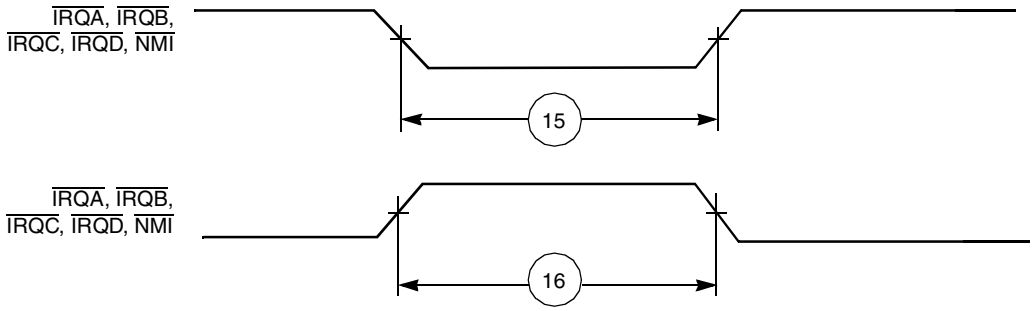


Figure 2-5. External Interrupt Timing (Negative Edge-Triggered)

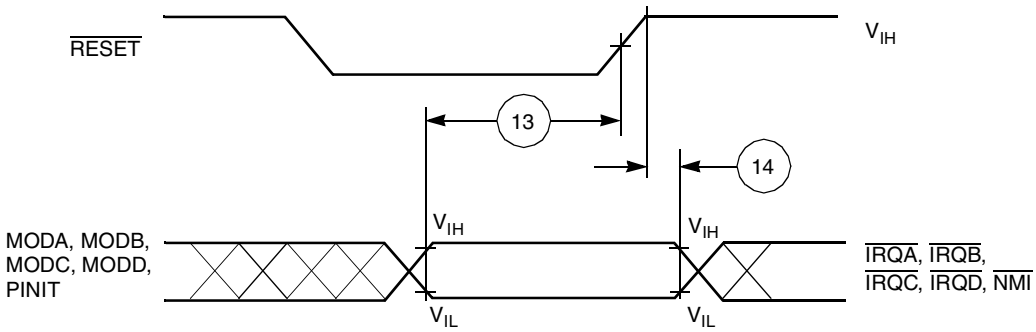


Figure 2-6. Operating Mode Select Timing

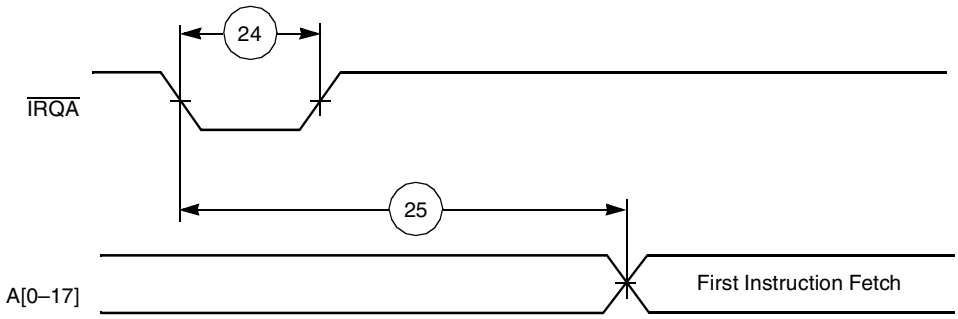


Figure 2-7. Recovery from Stop State Using IRQA

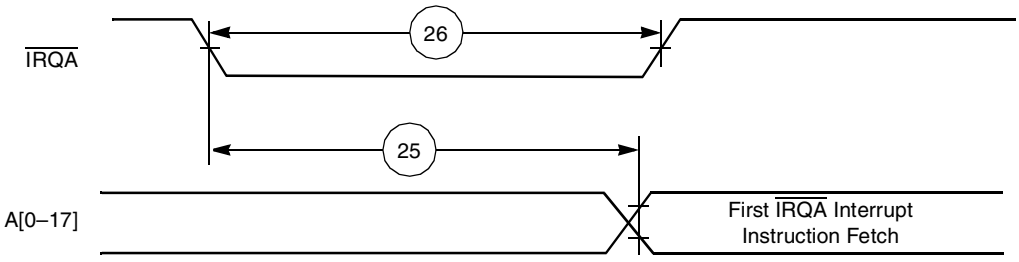
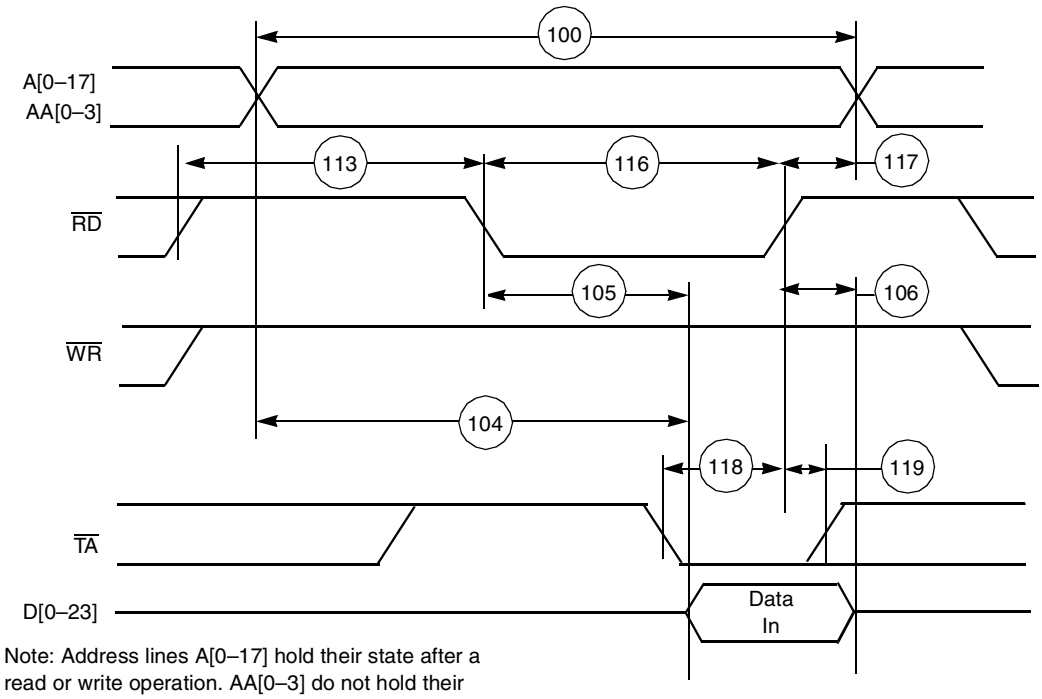
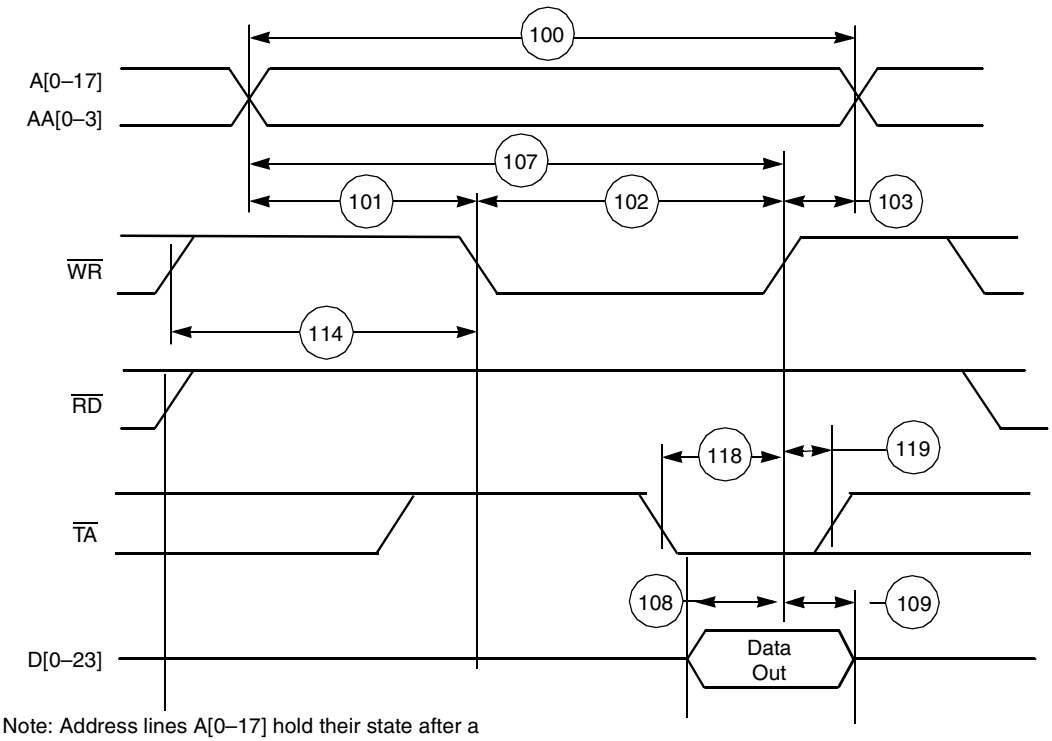


Figure 2-8. Recovery from Stop State Using IRQA Interrupt Service



Note: Address lines A[0–17] hold their state after a read or write operation. AA[0–3] do not hold their state after a read or write operation.

Figure 2-10. SRAM Read Access



Note: Address lines A[0–17] hold their state after a read or write operation. AA[0–3] do not hold their state after a read or write operation.

Figure 2-11. SRAM Write Access

Table 2-14. Host Interface Timings[1,2,12] (Continued)

No.	Characteristic[10]	Expression	150 MHz		Unit
			Min	Max	
340	Delay from data strobe assertion to host request deassertion for "Last Data Register" read or write (HROD=0)[4, 7, 8]		—	13.0	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 Big Endian mode (HLEND = 0; HLEND is the Interface Control Register bit 7—ICR[7]), or RXH/TXH in the Little Endian mode (HLEND = 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_{CCQH} = 3.3 V ± 0.3 V, V_{CC} = 1.8 V ± 0.1 V; T_J = –40°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 is ready for operation after three DSP clock cycles (3 × Tc).

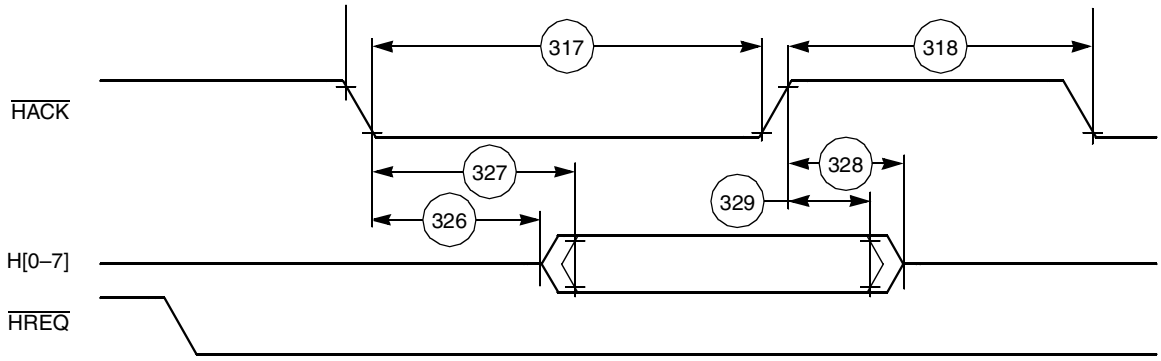


Figure 2-20. Host Interrupt Vector Register (IVR) Read Timing Diagram

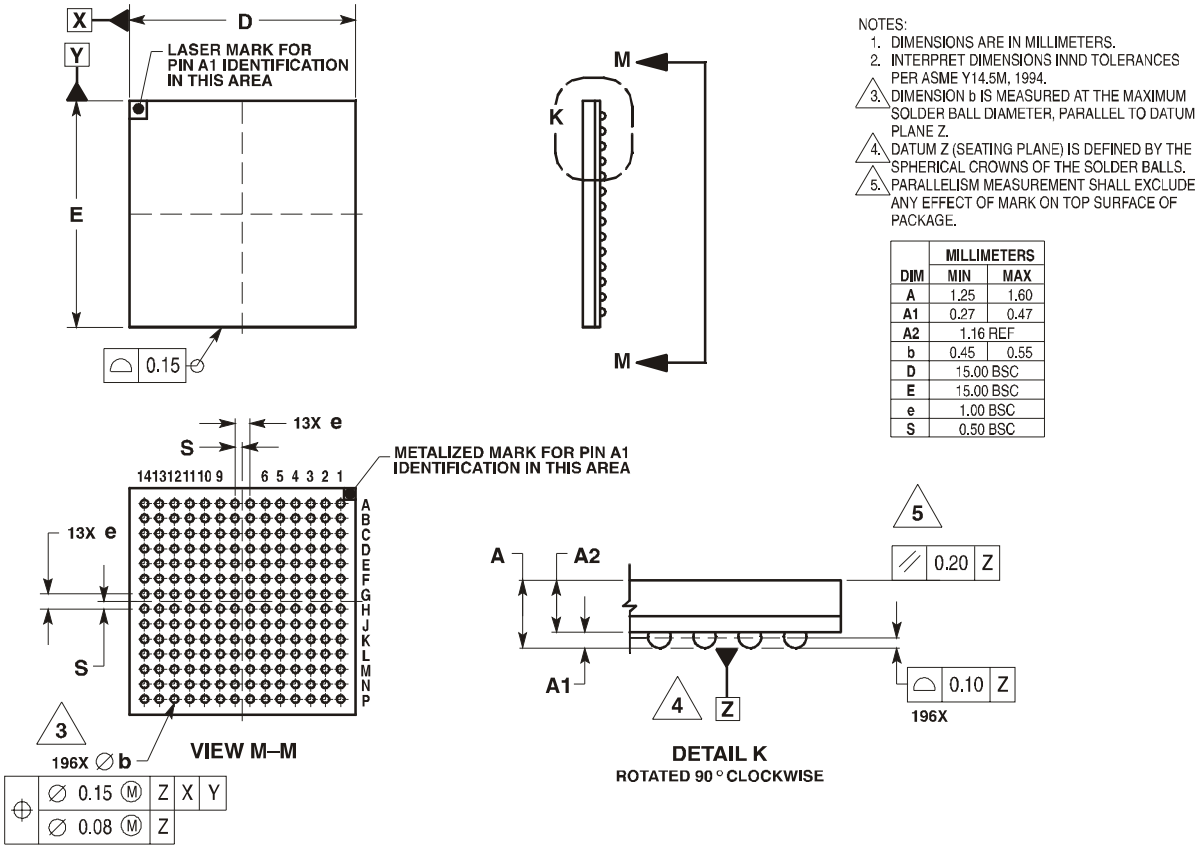
Table 3-1. Signal List by Ball Number

Ball No.	Signal Name	Ball No.	Signal Name	Ball 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	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	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 3-2. Signal List by Signal Name (Continued)

Signal Name	Ball No.	Signal Name	Ball No.	Signal Name	Ball No.
$\overline{\text{IRQA}}$	C4	PC3	H3	STD1	C2
$\overline{\text{IRQB}}$	A5	PC4	E3	$\overline{\text{TA}}$	P10
$\overline{\text{IRQC}}$	C5	PC5	E1	TCK	C3
$\overline{\text{IRQD}}$	B5	PCAP	P5	TDI	B3
MODA	C4	PD0	F2	TDO	A4
MODB	A5	PD1	A2	TIO0	L3
MODC	C5	PD2	B2	TIO1	L2
MODD	B5	PD3	G1	TIO2	K3
NC	A1	PD4	B1	TMS	A3
NC	A14	PD5	C2	$\overline{\text{TRST}}$	B4
NC	B14	PE0	F1	TXD	G3
NC	P1	PE1	G3	V_{CCA}	H12
NC	P14	PE2	G2	V_{CCA}	K12
$\overline{\text{NMI}}$	D1	PINIT	D1	V_{CCA}	L12
PB0	M5	$\overline{\text{RAS0}}$	N13	V_{CCC}	N12
PB1	P4	$\overline{\text{RAS1}}$	P12	V_{CCC}	P9
PB10	M2	$\overline{\text{RAS2}}$	P7	V_{CCD}	A7
PB11	J2	$\overline{\text{RAS3}}$	N7	V_{CCD}	C9
PB12	J3	$\overline{\text{RD}}$	M12	V_{CCD}	C11
PB13	L1	$\overline{\text{RESET}}$	N5	V_{CCD}	D14
PB14	K2	RXD	F1	V_{CCH}	M4
PB15	J1	SC00	F3	V_{CCP}	M6
PB2	N4	SC01	D2	V_{CCQH}	F12
PB3	P3	SC02	C1	V_{CCQH}	H1
PB4	N3	SC10	F2	V_{CCQH}	M7
PB5	P2	SC11	A2	V_{CCQL}	C7
PB6	N1	SC12	B2	V_{CCQL}	G13
PB7	N2	SCK0	H3	V_{CCQL}	H2
PB8	M3	SCK1	G1	V_{CCQL}	N9
PB9	M1	SCLK	G2	V_{CCS}	E2
PC0	F3	SRD0	E3	V_{CCS}	K1
PC1	D2	SRD1	B1	$\overline{\text{WR}}$	M11
PC2	C1	STD0	E1	XTAL	P8

3.2 MAP-BGA Package Mechanical Drawing



CASE 1128C–01
ISSUE O

DATE 07/28/98

Figure 3-3. DSP56311 Mechanical Information, 196-pin MAP-BGA Package

Design Considerations 4

This section describes various areas to consider when incorporating the DSP56311 device into a system design.

4.1 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.

- 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 $\overline{\text{IRQA}}$, $\overline{\text{IRQB}}$, $\overline{\text{IRQC}}$, $\overline{\text{IRQD}}$, $\overline{\text{TA}}$, and $\overline{\text{BG}}$ 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 ($\overline{\text{TRST}}$, TMS, $\overline{\text{DE}}$).
- Take special care to minimize noise levels on the V_{CCP}, GND_P, and GND_{P1} pins.
- The following pins must be asserted during power-up: $\overline{\text{RESET}}$ and $\overline{\text{TRST}}$. A stable EXTAL signal should be supplied before deassertion of $\overline{\text{RESET}}$. If the V_{CC} reaches the required level before EXTAL is stable or other "required $\overline{\text{RESET}}$ duration" conditions are met (see **Table 2-7**), the device circuitry can be in an uninitialized state that may result in significant power consumption and heat-up. Designs should minimize this condition to the shortest possible duration.
- Ensure that during power-up, and throughout the DSP56311 operation, V_{CCQH} is always higher or equal to the V_{CC} voltage level.
- 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.
- The Port A data bus (D[0–23]), HI08, ESSI0, ESSI1, SCI, and timers all use internal keepers to maintain the last output value even when the internal signal is tri-stated. Typically, no pull-up or pull-down resistors should be used with these signal lines. However, if the DSP is connected to a device that requires pull-up resistors (such as an MPC8260), the recommended resistor value is 10 KΩ or less. If more than one DSP must be connected in parallel to the other device, the pull-up resistor value requirement changes as follows:
  - 2 DSPs = 7 KΩ or less
  - 3 DSPs = 4 KΩ or less
  - 4 DSPs = 3 KΩ or less
  - 5 DSPs = 2 KΩ or less
  - 6 DSPs = 1.5 KΩ or less

4.3 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 **Equation 4**.

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

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

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.

4.4 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.

4.4.1 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**, *External Clock Timing*, on page 2-5 for 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.

4.4.3 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.

4.5 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.

```
        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
;
;************************************************************************
```