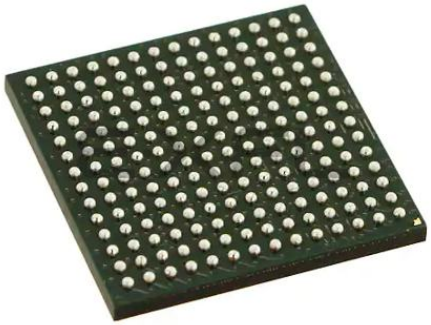


Welcome to E-XFL.COM

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	200MHz
Non-Volatile Memory	ROM (576B)
On-Chip RAM	576kB
Voltage - I/O	3.30V
Voltage - Core	1.60V
Operating Temperature	-40°C ~ 100°C (TJ)
Mounting Type	Surface Mount
Package / Case	196-BGA
Supplier Device Package	196-MAPBGA (15x15)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/dsp56321vl200r2

Table of Contents

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”	Means that a high true (active high) signal is high or that a low true (active low) signal is low
“deasserted”	Means that a high true (active high) signal is low or that 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

Table 1 lists the features of the DSP56321 device.

Table 1. DSP56321 Features

Feature	Description
High-Performance DSP56300 Core	• 275 million multiply-accumulates per second (MMACS) (550 MMACS using the EFCOP in filtering applications) with a 275 MHz clock at 1.6 V core and 3.3 V I/O • Object code compatible with the DSP56000 core with highly parallel instruction set • Data arithmetic logic unit (Data ALU) with fully pipelined 24 × 24-bit parallel Multiplier-Accumulator (MAC), 56-bit parallel barrel shifter (fast shift and normalization; bit stream generation and parsing), conditional ALU instructions, and 24-bit or 16-bit arithmetic support under software control • Program control unit (PCU) with position independent code (PIC) support, addressing modes optimized for DSP applications (including immediate offsets), internal instruction cache controller, internal memory-expandable hardware stack, nested hardware DO loops, and fast auto-return interrupts • Direct memory access (DMA) with six DMA channels supporting internal and external accesses; one-, two-, and three-dimensional transfers (including circular buffering); end-of-block-transfer interrupts; and triggering from interrupt lines and all peripherals • Phase-lock loop (PLL) allows change of low-power divide factor (DF) without loss of lock and output clock with skew elimination • Hardware debugging support including on-chip emulation (OnCE) module, Joint Test Action Group (JTAG) test access port (TAP)

Enhanced Filter Coprocessor (EFCOP)	• Internal 24 × 24-bit filtering and echo-cancellation coprocessor that runs in parallel to the DSP core
	• Operation at the same frequency as the core (up to 275 MHz)
	• Support for a variety of filter modes, some of which are optimized for cellular base station applications:
	  • Real finite impulse response (FIR) with real taps
	  • Complex FIR with complex taps
	  • Complex FIR generating pure real or pure imaginary outputs alternately
	  • A 4-bit decimation factor in FIR filters, thus providing a decimation ratio up to 16
	  • Direct form 1 (DFI) Infinite Impulse Response (IIR) filter
	  • Direct form 2 (DFII) IIR filter
	  • Four scaling factors (1, 4, 8, 16) for IIR output
	  • Adaptive FIR filter with true least mean square (LMS) coefficient updates
	  • Adaptive FIR filter with delayed LMS coefficient updates

Internal Peripherals	• Enhanced 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

Signals/Connections 1

The DSP56321 input and output signals are organized into functional groups as shown in **Table 1-1**. **Figure 1-1** diagrams the DSP56321 signals by functional group. The remainder of this chapter describes the signal pins in each functional group.

Table 1-1. DSP56321 Functional Signal Groupings

Functional Group		Number of Signals
Power (V_{CC})		20
Ground (GND)		66
Clock		2
Address bus	Port A[1]	18
Data bus		24
Bus control		10
Interrupt and mode control		6
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. Eight signal lines are not connected internally. These are designated as no connect (NC) in the package description (see **Chapter 3**). There are also two reserved lines.

Note: This chapter refers to a number of configuration registers used to select individual multiplexed signal functionality. See the *DSP56321 Reference Manual* for details on these configuration registers.

1.6 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.

1.6.1 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 1-9**.

Table 1-9. 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.

1.6.2 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 1-10. 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 2-5. External Clock Operation (Continued)

No.	Characteristics	Symbol	200 MHz		220 MHz		240 MHz		275 MHz	
			Min	Max	Min	Max	Min	Max	Min	Max
4	EXTAL cycle time[3] • With DPLL disabled • With DPLL enabled	ET_C	 5.0 ns 5.0 ns	 ∞ 62.5 ns	 4.55 ns 4.55 ns	 ∞ 62.5 ns	 4.17 ns 4.17 ns	 ∞ 62.5 ns	 3.64 ns 3.64 ns	 ∞ 62.5 ns
7	Instruction cycle time = $I_{CYC} = ET_C$ • With DPLL disabled • With DPLL enabled	I_{CYC}	 10 ns 5.0 ns	 ∞ 1.6 µs	 9.09 ns 4.55 ns	 ∞ 1.6 µs	 8.33 ns 4.17 ns	 ∞ 1.6 µs	 7.28 ns 3.64 ns	 ∞ 1.6 µs

Notes:
1. The rise and fall time of this external clock should be 2 ns maximum.
2. Refer to **Table 2-6** for a description of PDF and PDFR.
3. Measured at 50 percent of the input transition.
4. 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.

Note: If an externally-supplied square wave voltage source is used, disable the internal oscillator circuit after boot-up by setting XTLD (PCTL Register bit 2 = 1—see the *DSP56321 Reference Manual*). The external square wave source connects to EXTAL and XTAL is not used. **Figure 2-2** shows the EXTAL input signal.

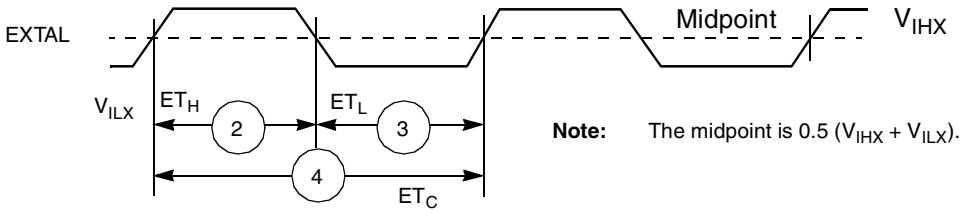


Figure 2-2. External Input Clock Timing

2.4.3 Clock Generator (CLKGEN) and Digital PLL (DPLL) Characteristics

Table 2-6. CLKGEN and DPLL Characteristics

Characteristics	Symbol	200 MHz		220 MHz		240 MHz		275 MHz		Unit
		Min	Max	Min	Max	Min	Max	Min	Max	
Predivision factor	PDF[1]	1	16	1	16	1	16	1	16	—
Predivider output clock frequency range	PDFR	16	32	16	32	16	32	16	32	MHz
Total multiplication factor[2]	MF	5	15	5	15	5	15	5	15	—
Multiplication factor integer part	MFI[1]	5	15	5	15	5	15	5	15	—
Multiplication factor numerator[3]	MFN	0	127	0	127	0	127	0	127	—
Multiplication factor denominator	MFD	1	128	1	128	1	128	1	128	—
Double clock frequency range	DDFR	160	400	160	440	160	480	160	550	MHz
Phase lock-in time[4]	DPLT	6.8[5]	150[6]	6.8[5]	150[6]	6.8[5]	150[6]	6.8[5]	150[6]	µs

Table 2-7. Reset, Stop, Mode Select, and Interrupt Timing[5] (CONTINUED)

No.	Characteristics	Expression	200 MHz		220 MHz		240 MHz		275 MHz		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	
20	Delay from $\overline{RD}$ assertion to interrupt request deassertion for level sensitive fast interrupts[1, 6, 7]	(WS + 3.25) × T_C – 10.94	—	Note 7	—	Note 7	—	Note 7	—	Note 7	ns
21	Delay from $\overline{WR}$ assertion to interrupt request deassertion for level sensitive fast interrupts[1, 6, 7] • SRAM WS = 3 • SRAM WS ≥ 4	 (WS + 3) × T_C – 10.94 (WS + 2.5) × T_C – 10.94	 — —	 Note 7 Note 7	 — —	 Note 7 Note 7	 — —	 Note 7 Note 7	 — —	 Note 7 Note 7	 ns ns
24	Duration for $\overline{IRQA}$ assertion to recover from Stop state		8.0	—	8.0	—	8.0	—	8.0	—	ns
25	Delay from $\overline{IRQA}$ assertion to fetch of first instruction (when exiting Stop)[2, 3] • DPLL is not active during Stop (PCTL Bit 1 = 0) and Stop delay is enabled (Operating Mode Register Bit 6 = 0) • DPLL is not active during Stop (PCTL Bit 1 = 0) and Stop delay is not enabled (Operating Mode Register Bit 6 = 1) • DPLL is active during Stop (PCTL Bit 1 = 1; Implies No Stop Delay)	 DPLT + (128K × T_C) DPLT + (23.75 ± 0.5) × T_C (10.0 ± 1.75) × T_C	 662.2 µs 6.9 41.25	 209.9 ms 188.8 58.8	 662.2 µs 6.9 37.5	 209.9 ms 188.8 53.3	 662.2 µs 6.9 34.4	 209.9 ms 188.8 49.0	 662.2 µs 6.9 30.0	 209.9 ms 188.8 43.0	 — µs ns
26	Duration of level sensitive $\overline{IRQA}$ assertion to ensure interrupt service (when exiting Stop)[2, 3] • DPLL is not active during Stop (PCTL bit 1 = 0) and Stop delay is enabled (Operating Mode Register Bit 6 = 0) • DPLL is not active during Stop (PCTL bit 1 = 0) and Stop delay is not enabled (Operating Mode Register Bit 6 = 1) • DPLL is active during Stop ((PCTL bit 1 = 0; implies no Stop delay)	 DPLT + (128 K × T_C) DPLT + (20.5 ± 0.5) × T_C 5.5 × T_C	 805.4 150.1 27.5	 — — —	 805.4 150.1 25	 — — —	 805.4 150.1 22.9	 — — —	 805.4 150.1 20.0	 — — —	 µs µs ns
27	Interrupt Request Rate • HI08, ESSI, SCI, Timer • DMA • $\overline{IRQ}$, $\overline{NMI}$ (edge trigger) • $\overline{IRQ}$, $\overline{NMI}$ (level trigger)	 12T_C 8T_C 8T_C 12T_C	 — — — —	 60.0 40.0 40.0 60.0	 — — — —	 54.6 36.4 36.4 54.6	 — — — —	 50.0 33.4 33.4 50.0	 — — — —	 43.7 29.2 29.2 43.7	 ns ns ns ns
28	DMA Request Rate • Data read from HI08, ESSI, SCI • Data write to HI08, ESSI, SCI • Timer • $\overline{IRQ}$, $\overline{NMI}$ (edge trigger)	 6T_C 7T_C 2T_C 3T_C	 — — — —	 30.0 35.0 10.0 15.0	 — — — —	 27.3 31.9 9.1 13.7	 — — — —	 25.0 29.2 8.3 12.5	 — — — —	 21.84 25.48 7.28 10.92	 ns ns ns ns
29	Delay from $\overline{IRQA}$, $\overline{IRQB}$, $\overline{IRQC}$, $\overline{IRQD}$, $\overline{NMI}$ assertion to external memory (DMA source) access address out valid	4.25 × T_C + 2.0	23.25	—	21.34	—	19.72	—	17.45	—	ns

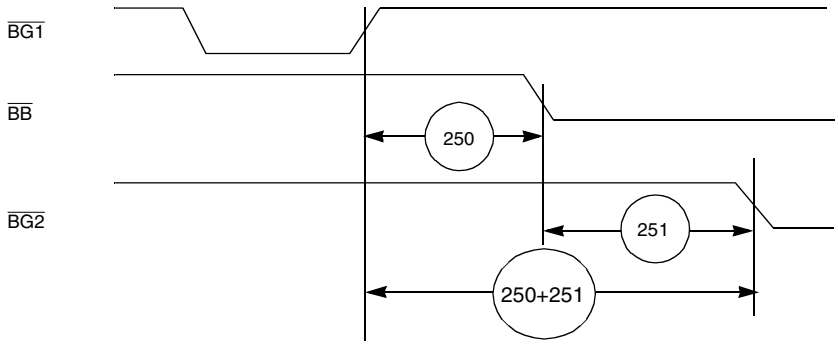


Figure 2-12. Asynchronous Bus Arbitration Timing

The asynchronous bus arbitration is enabled by internal synchronization circuits on $\overline{BG}$ and $\overline{BB}$ 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 $\overline{BB}$, for some time after $\overline{BG}$ is deasserted. This is the reason for timing 250.

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

2.4.6 Host Interface Timing

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

No.	Characteristic[10]	Expression	200 MHz		220 MHz		240 MHz		275 MHz		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	
317	Read data strobe assertion width[5] $\overline{HACK}$ assertion width	$T_C + 4.95$	9.95	—	9.05	—	8.3	—	7.77	—	ns
318	Read data strobe deassertion width[5] $\overline{HACK}$ deassertion width		4.95	—	4.5	—	4.13	—	4.0	—	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] $\overline{HACK}$ deassertion width after "Last Data Register" reads[8,11]	$2.5 \times T_C + 3.3$	15.8	—	14.7	—	13.7	—	12.39	—	ns
320	Write data strobe assertion width[6]		6.6	—	6.0	—	5.5	—	5.1	—	ns
321	Write data strobe deassertion width[8] $\overline{HACK}$ write deassertion width • after ICR, CVR and "Last Data Register" writes	$2.5 \times T_C + 3.3$	15.8	—	14.7	—	13.7	—	12.39	—	ns
	• after IVR writes, or after TXH:TXM:TXL writes (with HLEND= 0), or after TXL:TXM:TXH writes (with HLEND = 1)		8.25	—	7.5	—	6.88	—	6.28	—	ns
322	$\overline{HAS}$ assertion width		4.95	—	4.5	—	4.13	—	4.0	—	ns

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

No.	Characteristic[10]	Expression	200 MHz		220 MHz		240 MHz		275 MHz		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	
323	$\overline{HAS}$ deassertion to data strobe assertion[4]		0.0	—	0.0	—	0.0	—	0.0	—	ns
324	Host data input setup time before write data strobe deassertion[6]		4.95	—	4.5	—	4.13	—	4.0	—	ns
325	Host data input hold time after write data strobe deassertion[6]		1.65	—	1.5	—	1.38	—	1.23	—	ns
326	Read data strobe assertion to output data active from high impedance[5] $\overline{HACK}$ assertion to output data active from high impedance		1.65	—	1.5	—	1.38	—	1.23	—	ns
327	Read data strobe assertion to output data valid[5] $\overline{HACK}$ assertion to output data valid		—	14.78	—	13.45	—	12.32	—	10.2	ns
328	Read data strobe deassertion to output data high impedance[5] $\overline{HACK}$ deassertion to output data high impedance		—	4.95	—	4.5	—	4.13	4.0	—	ns
329	Output data hold time after read data strobe deassertion[5] Output data hold time after $\overline{HACK}$ deassertion		1.65	—	1.5	—	1.38	—	1.23	—	ns
330	$\overline{HCS}$ assertion to read data strobe deassertion[5]	$T_C + 4.95$	9.95	—	9.05	—	8.3	—	7.77	—	ns
331	$\overline{HCS}$ assertion to write data strobe deassertion[6]		8	—	8	—	8	—	8	—	ns
332	$\overline{HCS}$ assertion to output data valid		—	17	—	16	—	15	—	14	ns
333	$\overline{HCS}$ hold time after data strobe deassertion[4]		0.0	—	0.0	—	0.0	—	0.0	—	ns
334	Address (HAD[0–7]) setup time before $\overline{HAS}$ deassertion (HMUX=1)		2.31	—	2.1	—	1.93	—	1.76	—	ns
335	Address (HAD[0–7]) hold time after $\overline{HAS}$ deassertion (HMUX=1)		1.65	—	1.5	—	1.38	—	1.23	—	ns
336	HA[8–10] (HMUX=1), HA[0–2] (HMUX=0), HR/$\overline{W}$ setup time before data strobe assertion[4] • Read • Write		 0 2.31	 — —	 0 2.1	 — —	 0 1.93	 — —	 0 1.76	 — —	 ns ns
337	HA[8–10] (HMUX=1), HA[0–2] (HMUX=0), HR/$\overline{W}$ hold time after data strobe deassertion[4]		1.65	—	1.5	—	1.38	—	1.23	—	ns
338	Delay from read data strobe deassertion to host request assertion for “Last Data Register” read[5, 7, 8]	$T_C + 2.64$	7.64	—	7.19	—	6.81	—	6.28	—	ns
339	Delay from write data strobe deassertion to host request assertion for “Last Data Register” write[6, 7, 8]	$1.5 \times T_C + 2.64$	10.14	—	9.47	—	8.9	—	8.1	—	ns
340	Delay from data strobe assertion to host request deassertion for “Last Data Register” read or write (HROD=0)[4, 7, 8]		—	12.14	—	11.04	—	10.12	—	9.0	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	—	300.0	—	300.0	—	300.0	ns

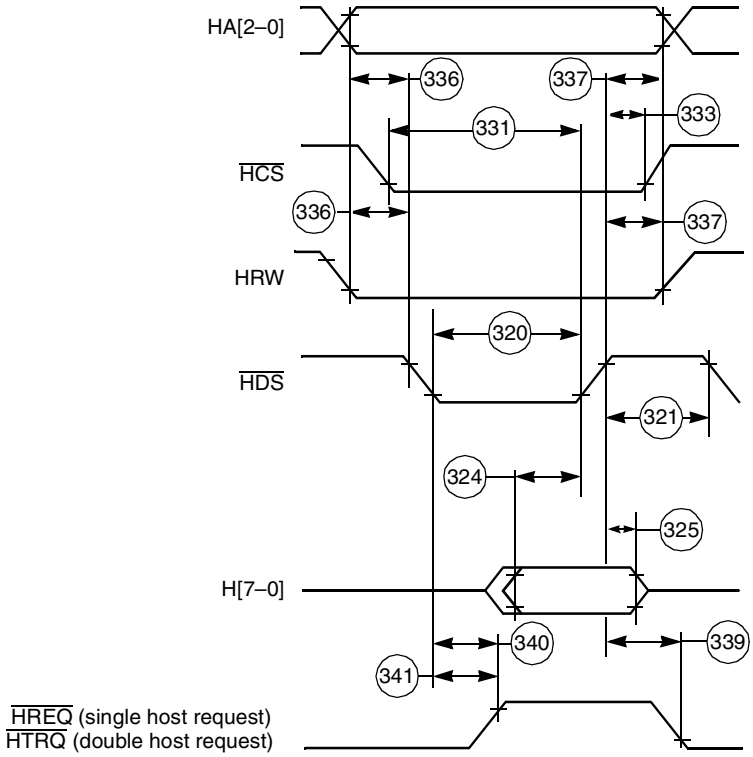


Figure 2-16. Write Timing Diagram, Non-Multiplexed Bus, Single Data Strobe

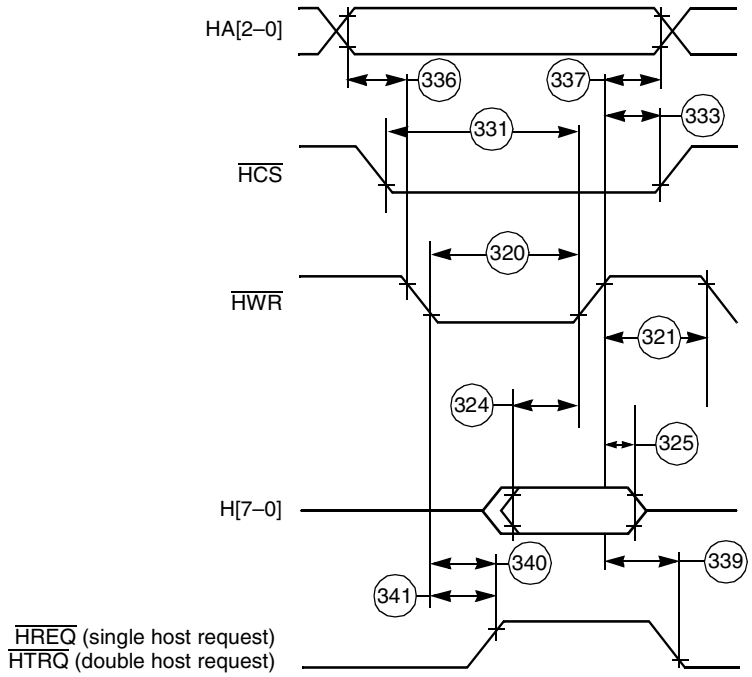


Figure 2-17. Write Timing Diagram, Non-Multiplexed Bus, Double Data Strobe

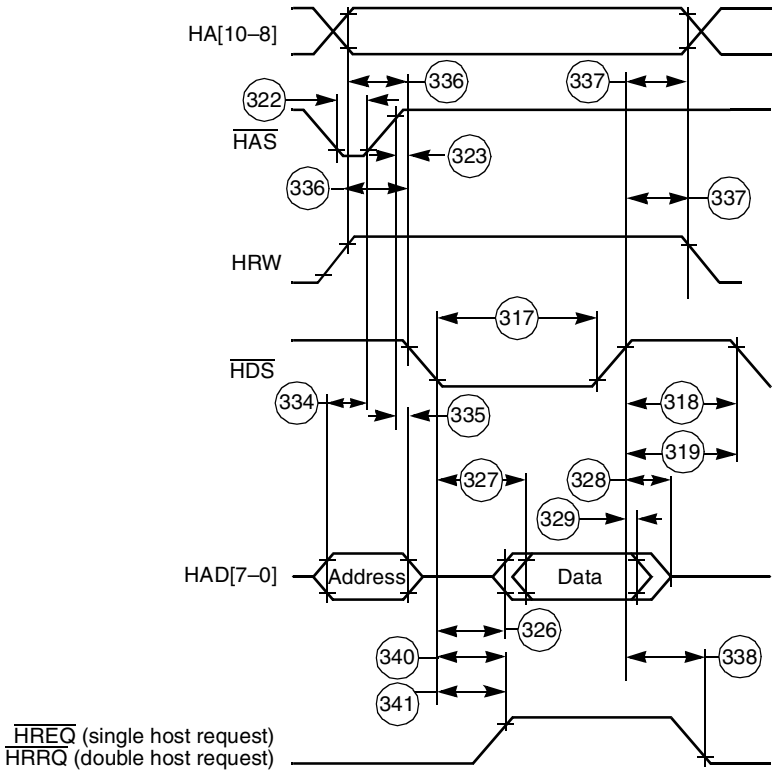


Figure 2-18. Read Timing Diagram, Multiplexed Bus, Single Data Strobe

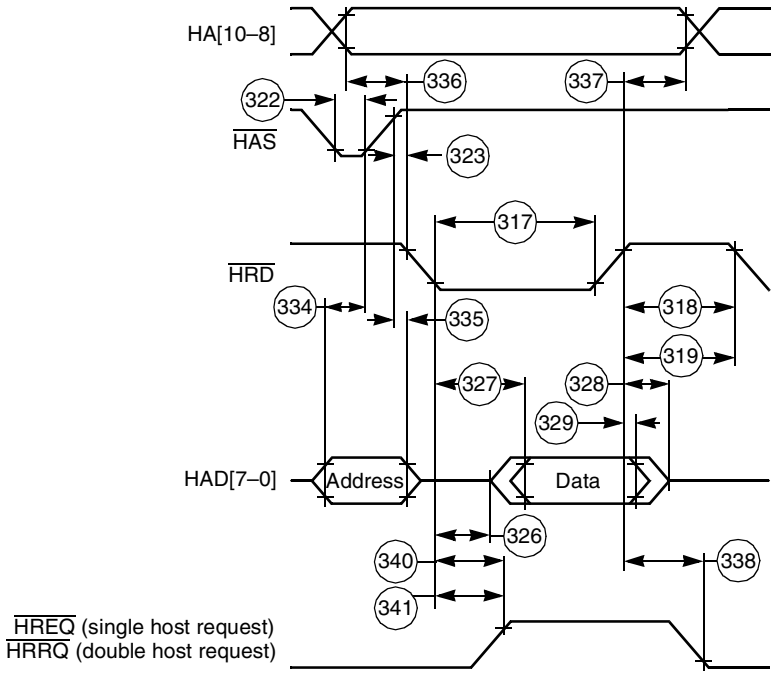


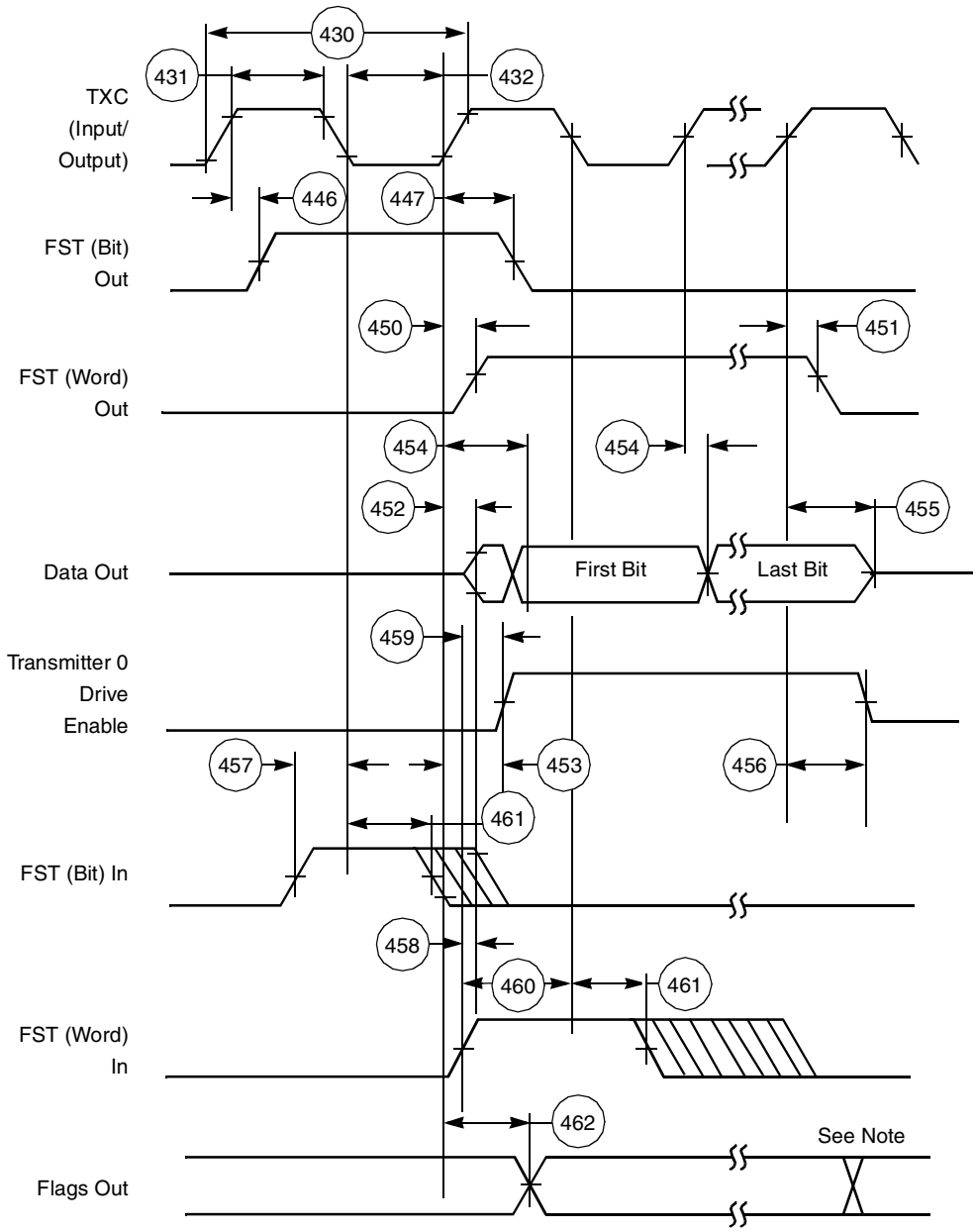
Figure 2-19. Read Timing Diagram, Multiplexed Bus, Double Data Strobe

2.4.7 SCI Timing

Table 2-11. SCI Timings

No.	Characteristics[1]	Symbol	Expression	200 MHz		220 MHz		240 MHz		275 MHz		Unit
				Min	Max	Min	Max	Min	Max	Min	Max	
400	Synchronous clock cycle	t_{SCC}[2]	$16 \times T_C$	80.0	—	72.8	—	66.7	—	58.0	—	ns
401	Clock low period		$t_{SCC}/2 - 10.0$	30.0	—	26.4	—	23.4	—	19.0	—	ns
402	Clock high period		$t_{SCC}/2 - 10.0$	30.0	—	26.4	—	23.4	—	19.0	—	ns
403	Output data setup to clock falling edge (internal clock)		$t_{SCC}/4 + 0.5 \times T_C - 17.0$	5.5	—	3.5	—	1.76	—	–0.68	—	ns
404	Output data hold after clock rising edge (internal clock)		$t_{SCC}/4 - 1.5 \times T_C$	13	—	11.5	—	10	—	9.04	—	ns
405	Input data setup time before clock rising edge (internal clock)		$t_{SCC}/4 + 0.5 \times T_C + 25.0$	47.5	—	45.5	—	43.8	—	41.32	—	ns
406	Input data not valid before clock rising edge (internal clock)		$t_{SCC}/4 + 0.5 \times T_C - 5.5$	—	17.0	—	15.0	—	13.8	—	10.81	ns
407	Clock falling edge to output data valid (external clock)			—	32.0	—	32.0	—	32.0	—	32.0	ns
408	Output data hold after clock rising edge (external clock)		$T_C + 8.0$	13.0	—	12.6	—	12.2	—	11.64	—	ns
409	Input data setup time before clock rising edge (external clock)			0.0	—	0.0	—	0.0	—	0.0	—	ns
410	Input data hold time after clock rising edge (external clock)			9.0	—	9.0	—	9.0	—	9.0	—	ns
411	Asynchronous clock cycle	t_{ACC}[3]	$64 \times T_C$	320.0	—	291.2	—	266.9	—	232.0	—	ns
412	Clock low period		$t_{ACC}/2 - 10.0$	150.0	—	135.6	—	123.5	—	106.0	—	ns
413	Clock high period		$t_{ACC}/2 - 10.0$	150.0	—	135.6	—	123.5	—	106.0	—	ns
414	Output data setup to clock rising edge (internal clock)		$t_{ACC}/2 - 30.0$	130.0	—	115.6	—	103.5	—	86.0	—	ns
415	Output data hold after clock rising edge (internal clock)		$t_{ACC}/2 - 30.0$	130.0	—	115.6	—	103.5	—	86.0	—	ns

Notes:
1. $V_{CCQH} = 3.3\ V \pm 0.3\ V$, $V_{CCQL} = 1.6\ V \pm 0.1\ V$; $T_J = -40°C$ 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).
4. In the timing diagrams that follow, the SCLK is drawn using the clock falling edge as a the first reference. Clock polarity is programmable in the SCI Control Register (SCR). Refer to the *DSP56321 Reference Manual* for details.



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 2-24. ESSI Transmitter Timing

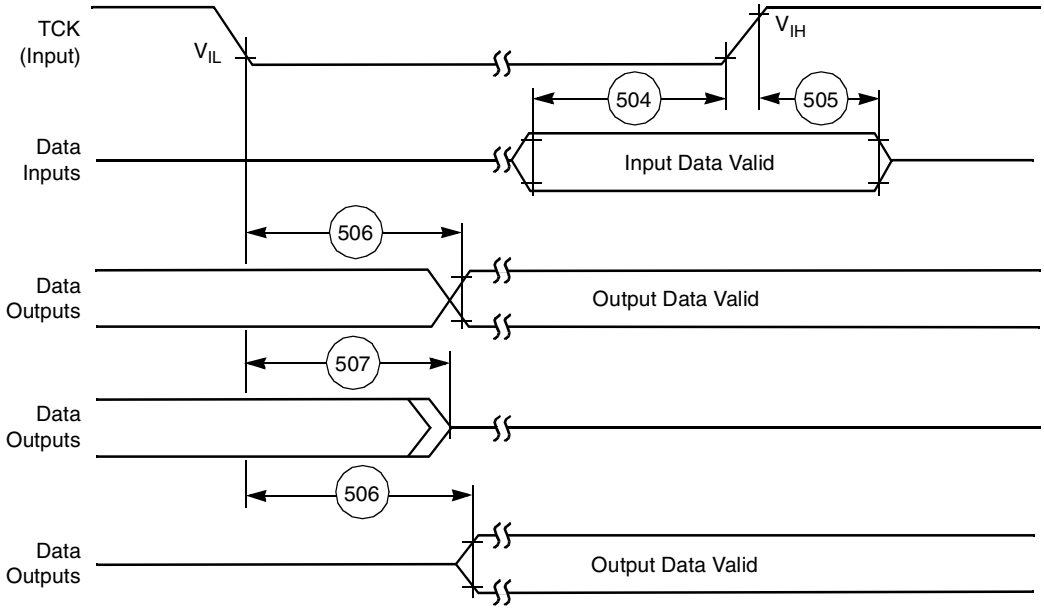


Figure 2-29. Boundary Scan (JTAG) Timing Diagram

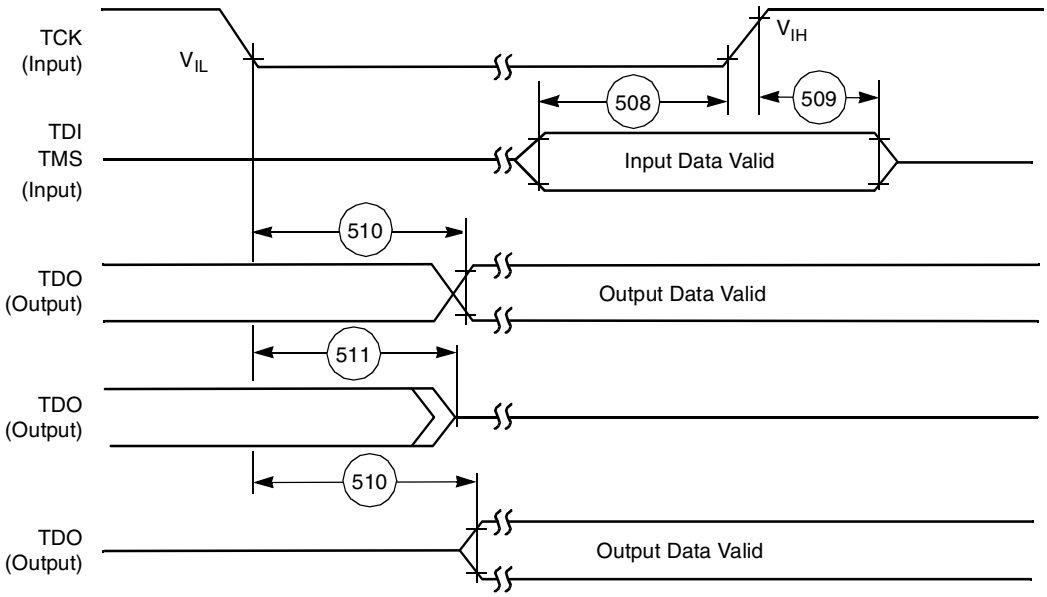


Figure 2-30. Test Access Port Timing Diagram

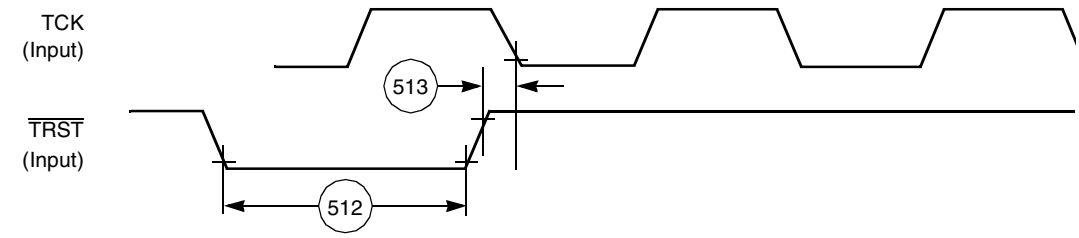


Figure 2-31. $\overline{\text{TRST}}$ Timing Diagram

3.1 Package Description

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

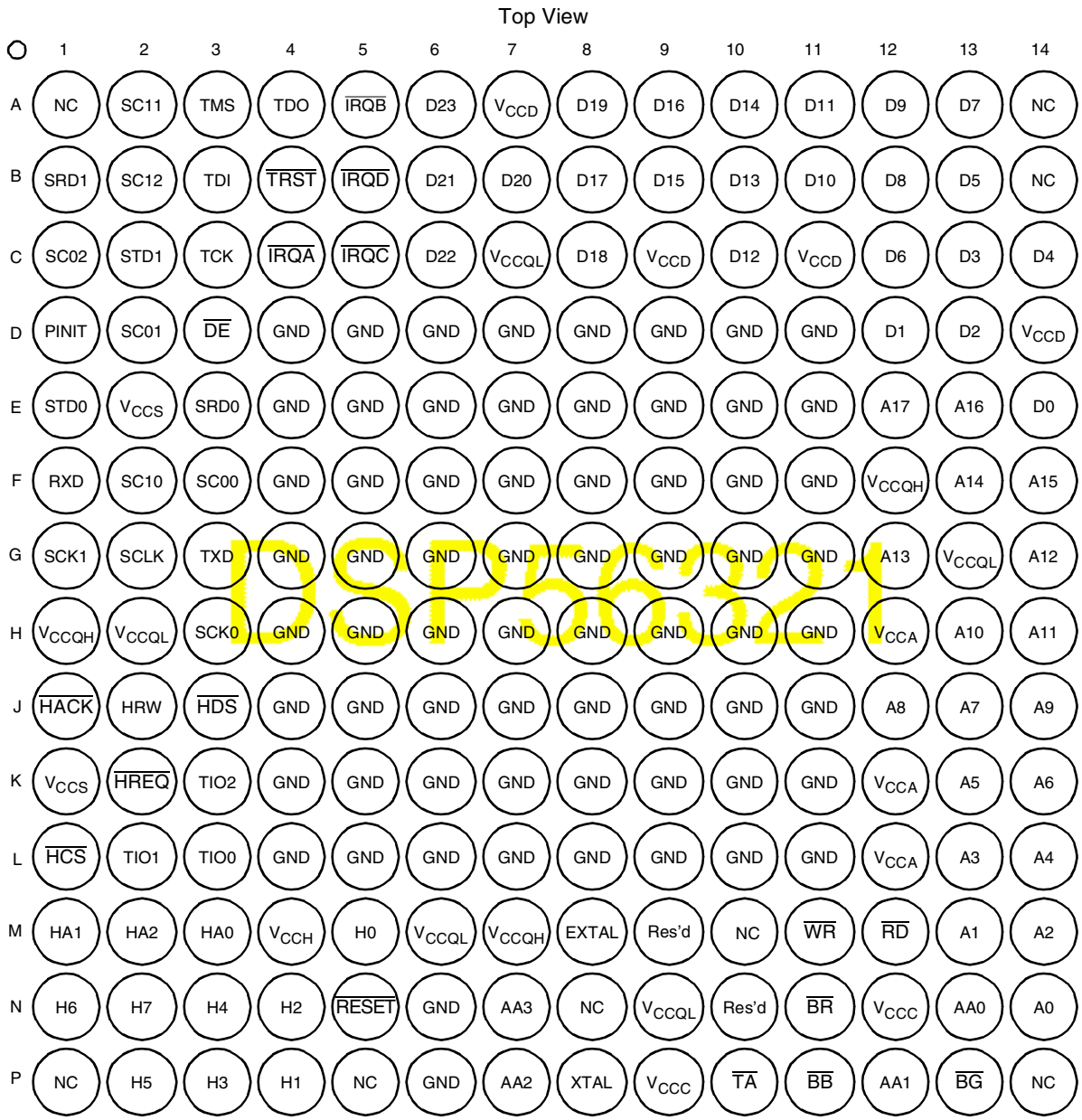


Figure 3-1. DSP56321 MAP-BGA Package, Top View

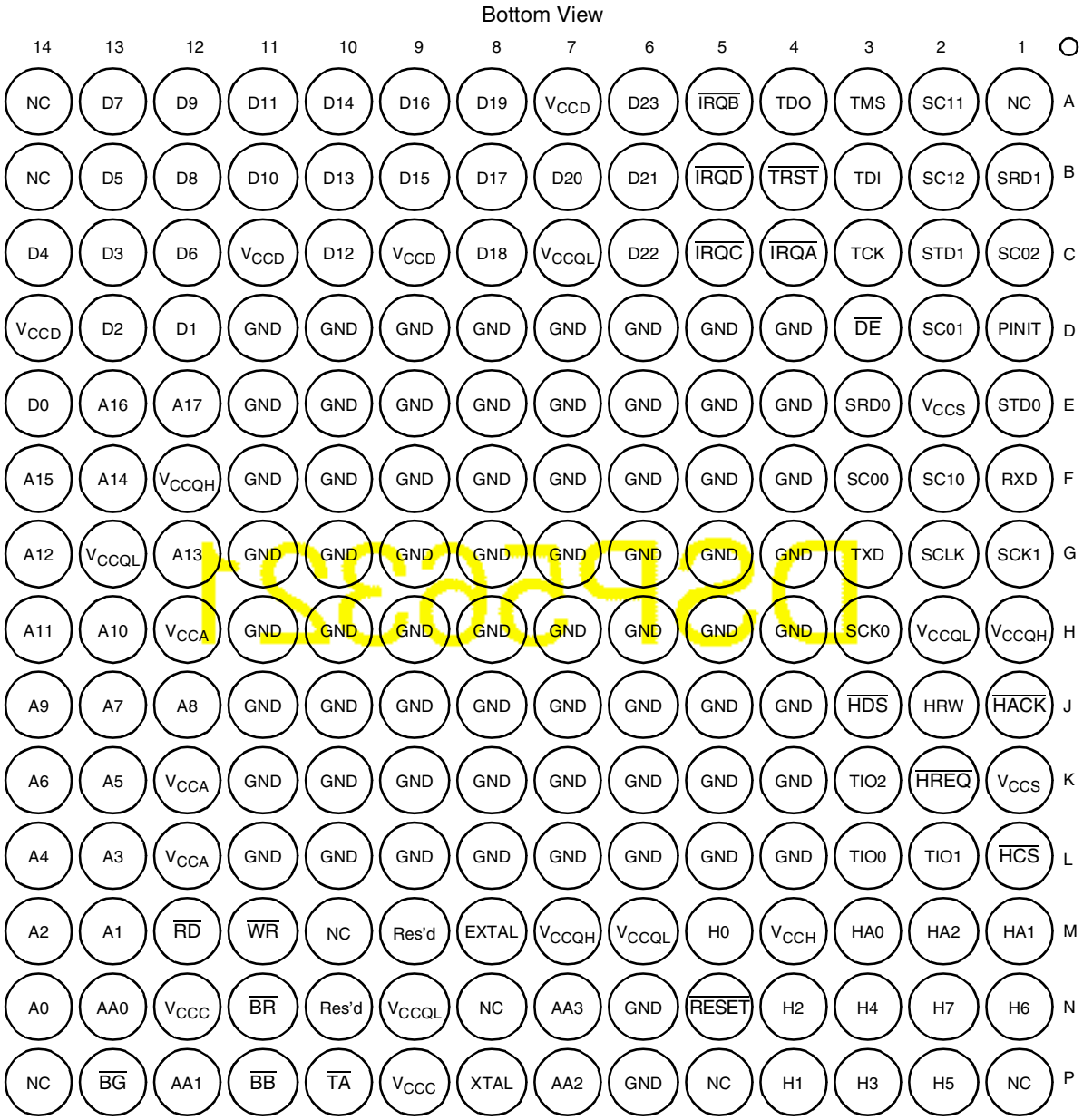


Figure 3-2. DSP56321 MAP-BGA Package, Bottom View

Table 3-1. Signal List by Ball Number

Ball No.	Signal Name	Ball No.	Signal Name	Ball No.	Signal Name
A1	Not Connected (NC)	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	$\overline{\text{DE}}$	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 3-2. Signal List by Signal Name (Continued)

Signal Name	Ball No.	Signal Name	Ball No.	Signal Name	Ball No.
GND	F9	GND	K4	HA1	M1
GND	F10	GND	K5	HA10	L1
GND	F11	GND	K6	HA2	M2
GND	G4	GND	K7	HA8	M1
GND	G5	GND	K8	HA9	M2
GND	G6	GND	K9	$\overline{\text{HACK}}$/HACK	J1
GND	G7	GND	K10	HAD0	M5
GND	G8	GND	K11	HAD1	P4
GND	G9	GND	L4	HAD2	N4
GND	G10	GND	L5	HAD3	P3
GND	G11	GND	L6	HAD4	N3
GND	H4	GND	L7	HAD5	P2
GND	H5	GND	L8	HAD6	N1
GND	H6	GND	L9	HAD7	N2
GND	H7	GND	L10	$\overline{\text{HAS}}$/HAS	M3
GND	H8	GND	L11	$\overline{\text{HCS}}$/HCS	L1
GND	H9	GND	N6	$\overline{\text{HDS}}$/HDS	J3
GND	H10	GND	P6	$\overline{\text{HRD}}$/HRD	J2
GND	H11	H0	M5	$\overline{\text{HREQ}}$/HREQ	K2
GND	J4	H1	P4	$\overline{\text{HRRQ}}$/HRRQ	J1
GND	J5	H2	N4	HRW	J2
GND	J6	H3	P3	$\overline{\text{HTRQ}}$/HTRQ	K2
GND	J7	H4	N3	$\overline{\text{HWR}}$/HWR	J3
GND	J8	H5	P2	$\overline{\text{IRQA}}$	C4
GND	J9	H6	N2	$\overline{\text{IRQB}}$	A5
GND	J10	H7	N2	$\overline{\text{IRQC}}$	C5
GND	J11	HA0	M3	$\overline{\text{IRQD}}$	B5

```
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_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
```