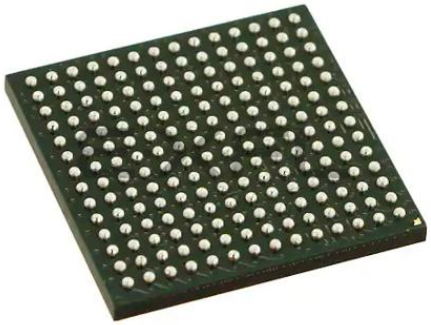


Welcome to E-XFL.COM

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	275MHz
Non-Volatile Memory	ROM (576B)
On-Chip RAM	576kB
Voltage - I/O	3.30V
Voltage - Core	1.60V
Operating Temperature	-40°C ~ 105°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/spakdsp321vf275

Table 1-7. External Bus Control Signals (Continued)

Signal Name	Type	State During Reset, Stop, or Wait	Signal Description
$\overline{TA}$	Input	Ignored Input	**Transfer Acknowledge**—If the DSP56321 is the bus master and there is no external bus activity, or the DSP56321 is not the bus master, the $\overline{TA}$ input is ignored. The $\overline{TA}$ input is a data transfer acknowledge (DTACK) function that can extend an external bus cycle indefinitely. Any number of wait states (1, 2. . .infinity) can be added to the wait states inserted by the bus control register (BCR) by keeping $\overline{TA}$ deasserted. In typical operation, $\overline{TA}$ 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 $\overline{TA}$ is asserted synchronous to CLKOUT. The number of wait states is determined by the $\overline{TA}$ 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. To use the $\overline{TA}$ functionality, the BCR must be programmed to at least one wait state. A zero wait state access cannot be extended by $\overline{TA}$ deassertion; otherwise, improper operation may result.
$\overline{BR}$	Output	Reset: Output (deasserted) State during Stop/Wait depends on BRH bit setting: • BRH = 0: Output (deasserted) • BRH = 1: Maintains last state (that is, if asserted, remains asserted)	**Bus Request**—Asserted when the DSP requests bus mastership. $\overline{BR}$ is deasserted when the DSP no longer needs the bus. $\overline{BR}$ may be asserted or deasserted independently of whether the DSP56321 is a bus master or a bus slave. Bus “parking” allows $\overline{BR}$ to be deasserted even though the DSP56321 is the bus master. (See the description of bus “parking” in the $\overline{BB}$ signal description.) The bus request hold (BRH) bit in the BCR allows $\overline{BR}$ to be asserted under software control even though the DSP does not need the bus. $\overline{BR}$ is typically sent to an external bus arbitrator that controls the priority, parking, and tenure of each master on the same external bus. $\overline{BR}$ is affected only by DSP requests for the external bus, never for the internal bus. During hardware reset, $\overline{BR}$ is deasserted and the arbitration is reset to the bus slave state.
$\overline{BG}$	Input	Ignored Input	**Bus Grant**—Asserted by an external bus arbitration circuit when the DSP56321 becomes the next bus master. When $\overline{BG}$ is asserted, the DSP56321 must wait until $\overline{BB}$ is deasserted before taking bus mastership. When $\overline{BG}$ 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. To ensure proper operation, the user must set the asynchronous bus arbitration enable (ABE) bit (Bit 13) in the Operating Mode Register. When this bit is set, $\overline{BG}$ and $\overline{BB}$ are synchronized internally. This adds a required delay between the deassertion of an initial $\overline{BG}$ input and the assertion of a subsequent $\overline{BG}$ input.
$\overline{BB}$	Input/ Output	Ignored Input	**Bus Busy**—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. $\overline{BB}$ is deasserted 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). **Notes:** **1.** See $\overline{BG}$ for additional information. **2.** $\overline{BB}$ requires an external pull-up resistor.

Table 1-10. Host Interface (Continued)

Signal Name	Type	State During Reset[1,2]	Signal Description
HA0	Input	Ignored Input	**Host Address Input 0**—When the HI08 is programmed to interface with a nonmultiplexed host bus and the HI function is selected, this signal is line 0 of the host address input bus.
$\overline{\text{HAS}}$/HAS	Input		**Host Address Strobe—**When the HI08 is programmed to interface with 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 ($\overline{\text{HAS}}$) following reset.
PB8	Input or Output		**Port B 8**—When the HI08 is configured as GPIO through the HI08 Port Control Register, this signal is individually programmed as an input or output through the HI08 Data Direction Register.
HA1	Input	Ignored Input	**Host Address Input 1**—When the HI08 is programmed to interface with 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 the HI08 is programmed to interface with 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 HI08 Port Control Register, this signal is individually programmed as an input or output through the HI08 Data Direction Register.
HA2	Input	Ignored Input	**Host Address Input 2**—When the HI08 is programmed to interface with 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 the HI08 is programmed to interface with 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 HI08 Port Control Register, this signal is individually programmed as an input or output through the HI08 Data Direction Register.
$\overline{\text{HCS}}$/HCS	Input	Ignored Input	**Host Chip Select**—When the HI08 is programmed to interface with 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 ($\overline{\text{HCS}}$) after reset.
HA10	Input		**Host Address 10**—When the HI08 is programmed to interface with 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 HI08 Port Control Register, this signal is individually programmed as an input or output through the HI08 Data Direction Register.
HRW	Input	Ignored Input	**Host Read/Write**—When the HI08 is programmed to interface with 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 the HI08 is programmed to interface with 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 HI08 Port Control Register, this signal is individually programmed as an input or output through the HI08 Data Direction Register.

1.11 JTAG and OnCE Interface

The DSP56300 family and in particular the DSP56321 support circuit-board test strategies based on the **IEEE® Std.** 1149.1™ 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. JTAG/OnCE Interface

Signal Name	Type	State During Reset	Signal Description
TCK	Input	Input	**Test Clock**—A test clock input signal to synchronize the JTAG test logic.
TDI	Input	Input	**Test Data Input**—A test data serial input signal 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**—A test data serial output signal for test instructions and data. TDO 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**—Sequences 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**—Initializes the test controller asynchronously. $\overline{\text{TRST}}$ has an internal pull-up resistor. $\overline{\text{TRST}}$ must be asserted during and after power-up (see EB610/D for details).
$\overline{\text{DE}}$	Input/ Output	Input	**Debug Event**—As an input, initiates Debug mode from an external command controller, and, as an open-drain output, acknowledges that the chip has entered Debug mode. As an input, $\overline{\text{DE}}$ causes the DSP56300 core to finish executing the current instruction, save the instruction pipeline information, enter 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 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 signal 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 Debug mode. All other interface with the OnCE module must occur through the JTAG port.

Specifications 2

The DSP56321 is fabricated in high-density CMOS with Transistor-Transistor Logic (TTL) compatible inputs and outputs.

2.1 Maximum Ratings

CAUTION

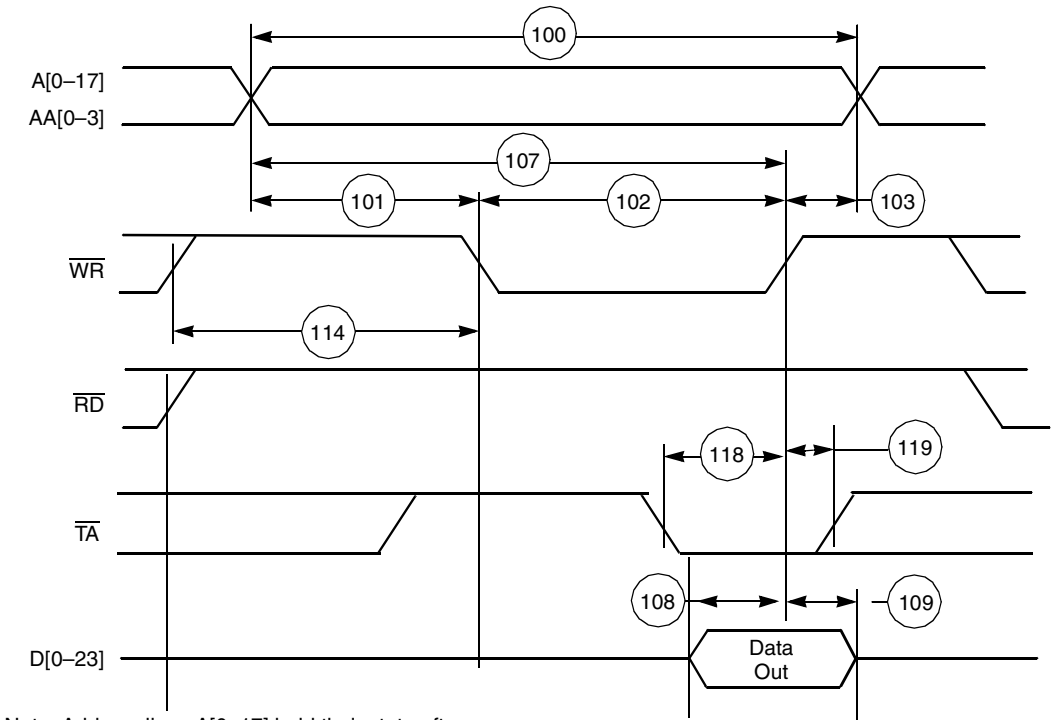
This device contains circuitry protecting against damage due to high static voltage or electrical fields; however, normal precautions should be taken to avoid exceeding maximum voltage ratings. Reliability is enhanced if unused inputs are tied to an appropriate logic voltage level (for example, either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

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. Absolute Maximum Ratings

Rating[1]	Symbol	Value[1, 2]	Unit
Supply Voltage[3]	V_{CCQL}	–0.1 to 2.25	V
Input/Output Supply Voltage[3]	V_{CCQH}	–0.3 to 4.35	V
All input voltages	V_{IN}	GND – 0.3 to V_{CCQH} + 0.3	V
Current drain per pin excluding V_{CC} and GND	I	10	mA
Operating temperature range	T_J	–40 to +100	°C
Storage temperature	T_{STG}	–55 to +150	°C

Notes:
1. GND = 0 V, V_{CCQL} = 1.6 V ± 0.1 V, V_{CCQH} = 3.3 V ± 0.3 V, T_J = –40°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 DSP56321 operation, V_{CCQH} voltage must always be higher or equal to V_{CCQL} voltage.



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

2.4.5.2 Asynchronous Bus Arbitration Timings

Table 2-9. Asynchronous Bus Timings

No.	Characteristics	Expression	200 MHz		220 MHz		240 MHz		275 Mhz		Unit
			Min	Max	Min	Max	Min	Max	Min	Max	
250	$\overline{BB}$ assertion window from $\overline{BG}$ input deassertion.	2.5 × Tc + 5	—	17.5	—	16.4	—	15.4	—	14.1	ns
251	Delay from $\overline{BB}$ assertion to $\overline{BG}$ assertion	2 × Tc + 5	15	—	14.1	—	13.3	—	12.27	—	ns

Notes:
1. Bit 13 in the Operating Mode Register must be set to enable Asynchronous Arbitration mode.
2. To guarantee timings 250 and 251, it is recommended that you assert non-overlapping $\overline{BG}$ inputs to different DSP56300 devices (on the same bus), as shown in **Figure 2-12**, where $\overline{BG1}$ is the $\overline{BG}$ signal for one DSP56300 device while $\overline{BG2}$ is the $\overline{BG}$ signal for a second DSP56300 device.

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

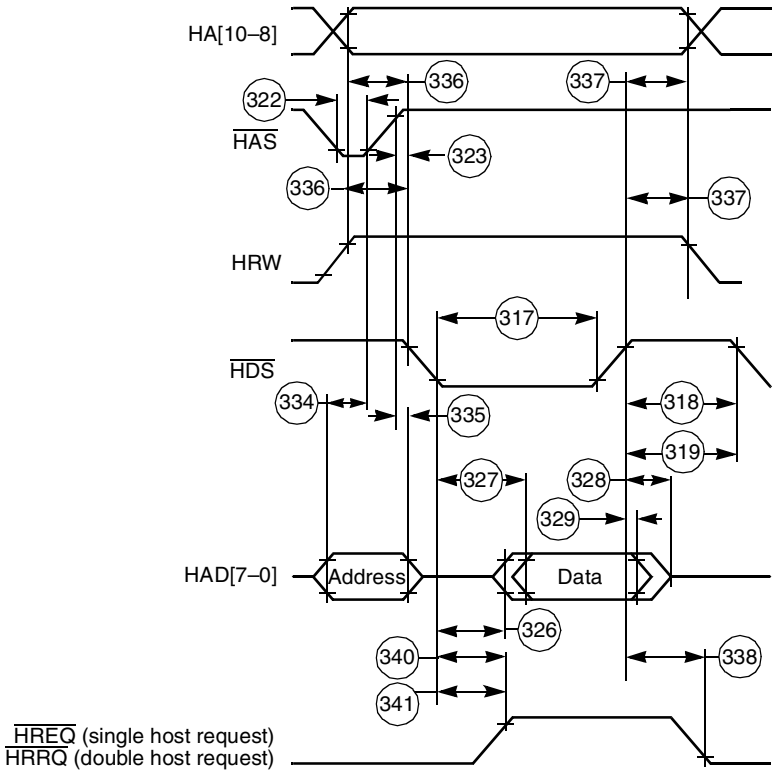


Figure 2-18. Read Timing Diagram, Multiplexed Bus, Single Data Strobe

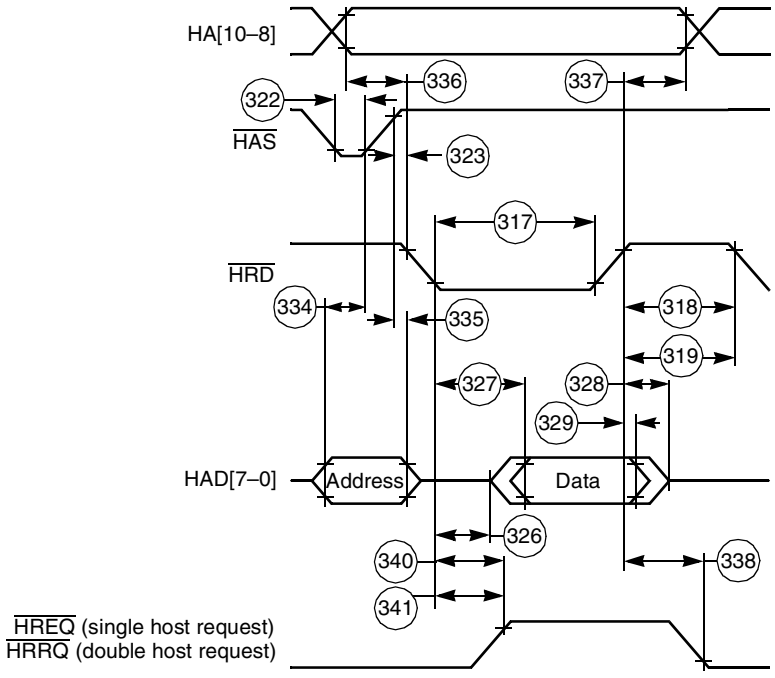


Figure 2-19. Read Timing Diagram, Multiplexed Bus, Double Data Strobe

2.4.8 ESSI0/ESSI1 Timing

Table 2-12. ESSI Timings

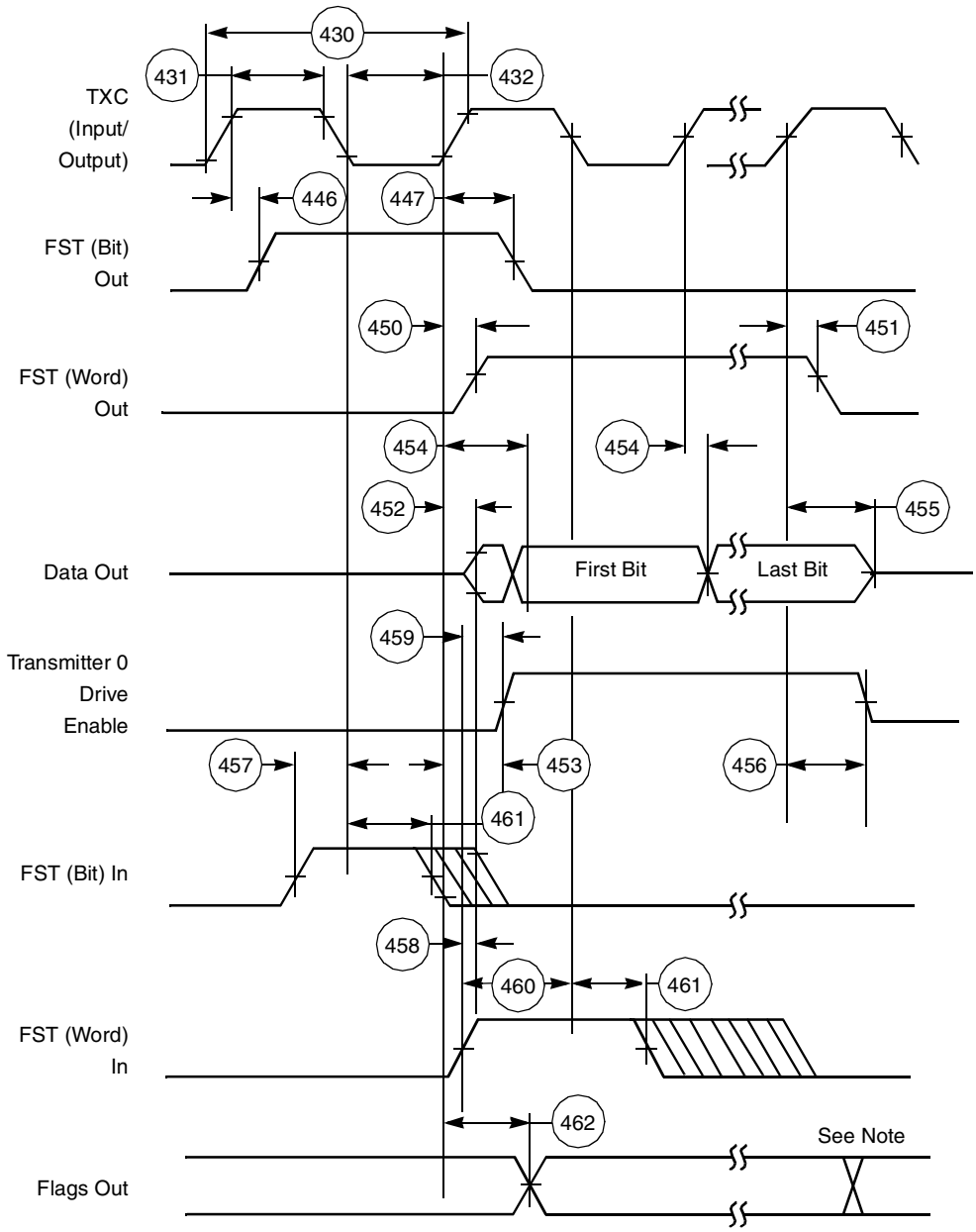
No.	Characteristics[4, 6]	Symbol	Expression	200 MHz		220 MHz		240 MHz		275 MHz		Condition[5]	Unit
				Min	Max	Min	Max	Min	Max	Min	Max		
430	Clock cycle[1]	T_{ECCX} T_{ECCI}	$6 \times T_C$ $8 \times T_C$	30.0 40.0	— —	27.3 36.6	— —	25.0 33.3	— —	21.5 25.0	— —	x ck i ck	ns ns
431	Clock high period • For internal clock • For external clock		 $T_{ECCX}/2 - 3.7$ $T_{ECCI}/2 - 10.0$	 11.3 10.0	 — —	 9.9 8.2	 — —	 8.8 6.7	 — —	 7.21 2.5	 — —		 ns ns
432	Clock low period • For internal clock • For external clock		 $T_{ECCX}/2 - 3.7$ $T_{ECCI}/2 - 10.0$	 11.3 10.0	 — —	 9.9 8.2	 — —	 8.8 6.7	 — —	 7.21 2.5	 — —		 ns ns
433	RXC rising edge to FSR out (bit-length) high			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck a	ns
434	RXC rising edge to FSR out (bit-length) low			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck a	ns
435	RXC rising edge to FSR out (word-length-relative) high[2]			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck a	ns
436	RXC rising edge to FSR out (word-length-relative) low[2]			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck a	ns
437	RXC rising edge to FSR out (word-length) high			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck a	ns
438	RXC rising edge to FSR out (word-length) low			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck a	ns
439	Data in setup time before RXC (SCK in Synchronous mode) falling edge			5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	x ck i ck	ns
440	Data in hold time after RXC falling edge			3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	x ck i ck	ns
441	FSR input (bl, wr) high before RXC falling edge[2]			5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	x ck i ck a	ns
442	FSR input (wl) high before RXC falling edge			5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	x ck i ck a	ns
443	FSR input hold time after RXC falling edge			3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	x ck i ck a	ns
444	Flags input setup before RXC falling edge			5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	x ck i ck s	ns
445	Flags input hold time after RXC falling edge			3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	x ck i ck s	ns
446	TXC rising edge to FST out (bit-length) high			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
447	TXC rising edge to FST out (bit-length) low			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
448	TXC rising edge to FST out (word-length-relative) high[2]			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
449	TXC rising edge to FST out (word-length-relative) low[2]			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
450	TXC rising edge to FST out (word-length) high			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns

Table 2-12. ESSI Timings (Continued)

No.	Characteristics[4, 6]	Symbol	Expression	200 MHz		220 MHz		240 MHz		275 MHz		Cond-ition[5]	Unit
				Min	Max	Min	Max	Min	Max	Min	Max		
451	TXC rising edge to FST out (word-length) low			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
452	TXC rising edge to data out enable from high impedance			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
453	TXC rising edge to Transmitter 0 drive enable assertion			— —	12.5 13.5	— —	12.5 13.5	— —	12.5 13.5	— —	12.5 13.5	x ck i ck	ns
454	TXC rising edge to data out valid			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
455	TXC rising edge to data out high impedance[3]			— —	30.0 8.3	— —	30.0 8.3	— —	30.0 8.3	— —	30.0 8.3	x ck i ck	ns
456	TXC rising edge to Transmitter 0 drive enable deassertion[3]			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns
457	FST input (bl, wr) setup time before TXC falling edge[2]			5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	x ck i ck	ns
458	FST input (wl) to data out enable from high impedance			— —	15.0 8.0	— —	15.0 8.0	— —	15.0 8.0	— —	15.0 8.0	x ck i ck	ns
459	FST input (wl) to Transmitter 0 drive enable assertion			— —	15.0 18.0	— —	15.0 18.0	— —	15.0 18.0	— —	15.0 18.0	x ck i ck	ns
460	FST input (wl) setup time before TXC falling edge			5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	5.0 10.0	— —	x ck i ck	ns
461	FST input hold time after TXC falling edge			3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	3.8 5.0	— —	x ck i ck	ns
462	Flag output valid after TXC rising edge			— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	— —	12.5 8.3	x ck i ck	ns

Notes:

1. For the internal clock, the external clock cycle is defined by the instruction cycle time (timing 7 in **Table 2-5** on page 2-4) and the ESSI control register. T_{ECCX} must be $\geq T_C \times 3$, in accordance with the note below Table 7-1 in the *DSP56321 Reference Manual*. T_{ECCI} must be $\geq T_C \times 4$, in accordance with the explanation of CRA[PSR] and the *ESSI Clock Generator Functional Block Diagram* shown in **Figure 7-3** of the *DSP56321 Reference Manual*.
2. The word-length-relative frame sync signal waveform operates the same way as the bit-length frame sync signal waveform, but spreads from one serial clock before the first bit clock (same as the Bit Length Frame Sync signal) until the one before last bit clock of the first word in the frame.
3. Periodically sampled and not 100 percent tested
4. V_{CCQH} = 3.3 V ±0.3 V, V_{CCQL} = 1.6 V ± 0.1 V; T_J = 0°C to +85°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. In the timing diagrams below, the clocks and frame sync signals are drawn using the clock falling edge as a the first reference. Clock and frame sync polarities are programmable in Control Register B (CRB). 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

2.4.11 JTAG Timing

Table 2-14. JTAG Timing

No.	Characteristics	All frequencies		Unit
		Min	Max	
500	TCK frequency of operation (1/(T_C × 3); absolute maximum 22 MHz)	0.0	22.0	MHz
501	TCK cycle time in Crystal mode	45.0	—	ns
502	TCK clock pulse width measured at 1.6 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_{CCQH} = 3.3 V ± 0.3 V, V_{CCQL} = 1.6 V ± 0.1 V; T_J = –40°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.

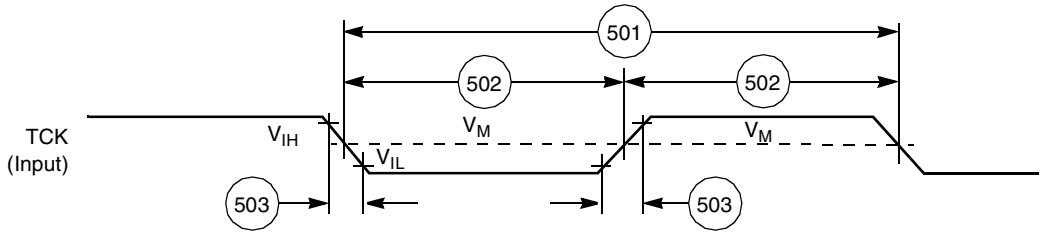


Figure 2-28. Test Clock Input Timing Diagram

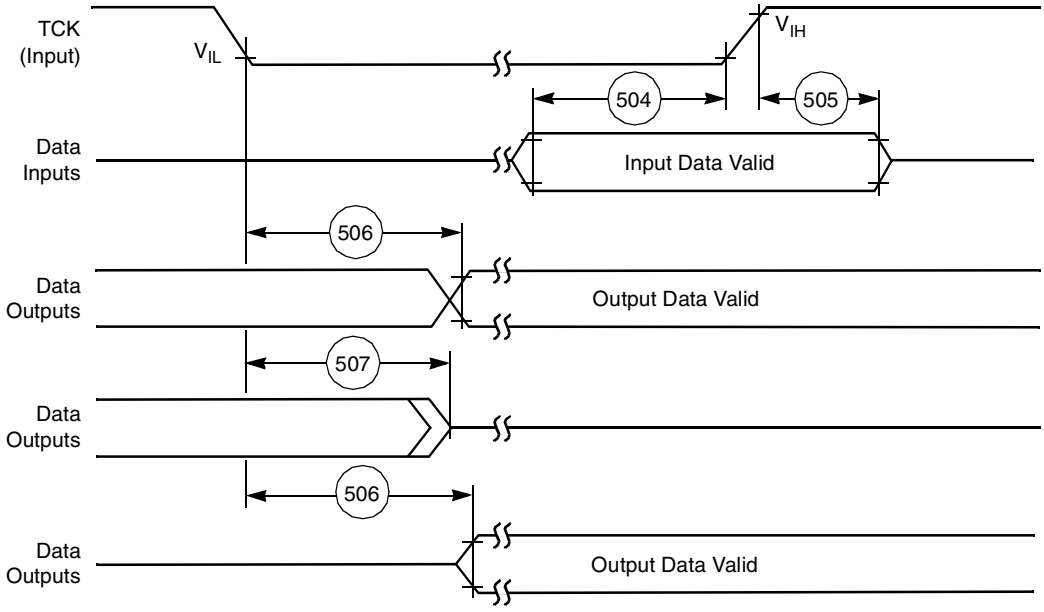


Figure 2-29. Boundary Scan (JTAG) Timing Diagram

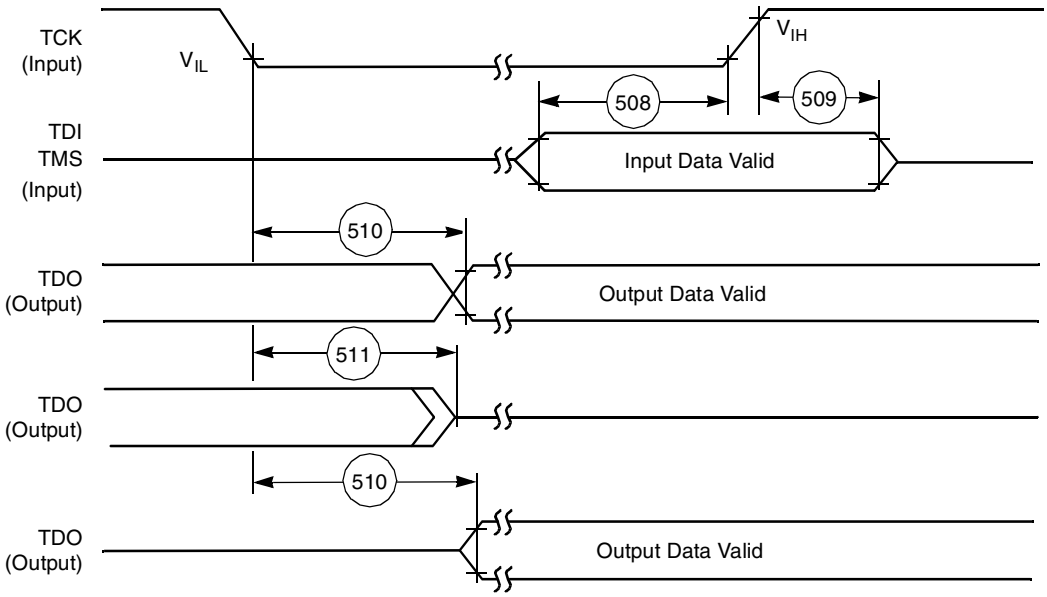


Figure 2-30. Test Access Port Timing Diagram

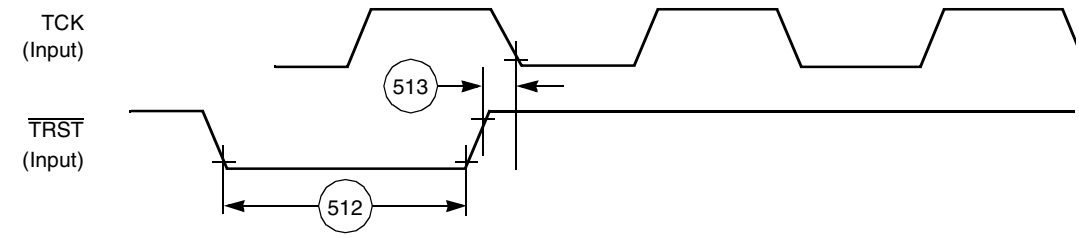


Figure 2-31. $\overline{\text{TRST}}$ Timing Diagram

2.4.12 OnCE Module TimIng

Table 2-15. OnCE Module Timing

No.	Characteristics	Expression	All Frequencies		Unit
			Min	Max	
500	TCK frequency of operation (1/(T_C × 3); maximum 22 MHz)	Max 22.0 MHz	0.0	22.0	MHz
514	$\overline{DE}$ assertion time in order to enter Debug mode	1.5 × T_C + 10.0	20.0	—	ns
515	Response time when DSP56321 is executing NOP instructions from internal memory	5.5 × T_C + 30.0	—	67.0	ns
516	Debug acknowledge assertion time	3 × T_C + 5.0	25.0	—	ns
Note:	V_{CCQH} = 3.3 V ± 0.3 V, V_{CCQL} = 1.6 V ± 0.1 V; T_J = –40°C to +100 °C, C_L = 50 pF				

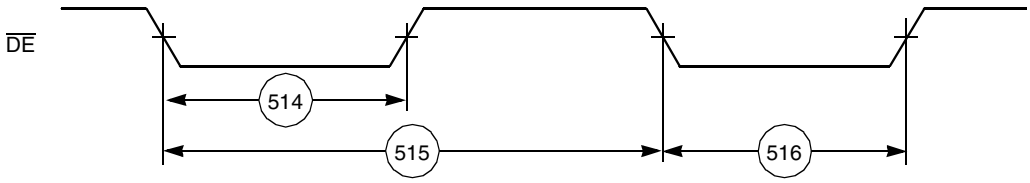


Figure 2-32. OnCE—Debug Request

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.

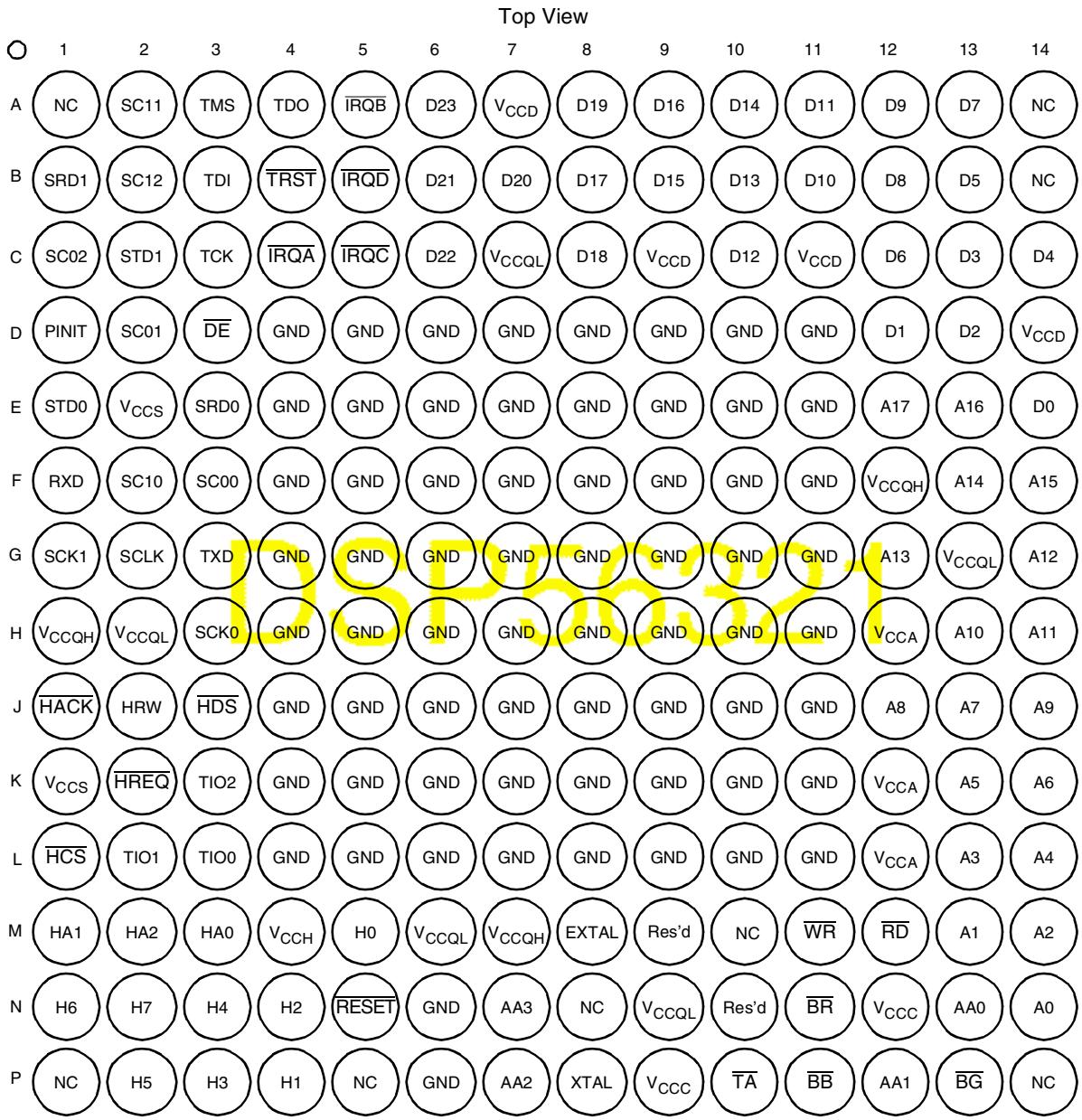


Figure 3-1. DSP56321 MAP-BGA Package, Top View

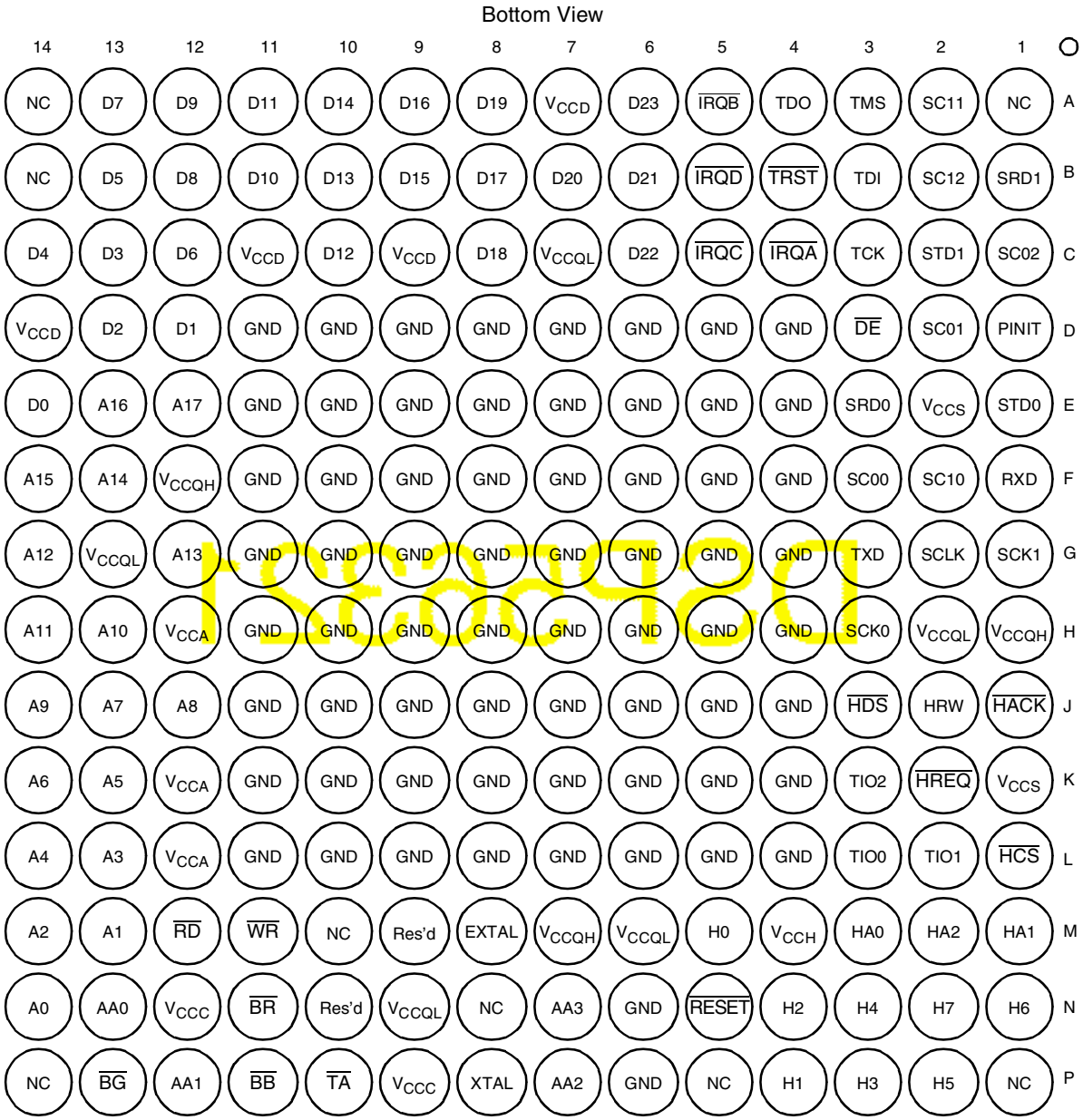


Figure 3-2. DSP56321 MAP-BGA Package, Bottom View

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.
MODA	C4	PB4	N3	Reserved	M9
MODB	A5	PB5	P2	Reserved	N10
MODC	C5	PB6	N1	$\overline{\text{RESET}}$	N5
MODD	B5	PB7	N2	RXD	F1
NC	A1	PB8	M3	SC00	F3
NC	A14	PB9	M1	SC01	D2
NC	B14	PC0	F3	SC02	C1
NC	M10	PC1	D2	SC10	F2
NC	N8	PC2	C1	SC11	A2
NC	P1	PC3	H3	SC12	B2
NC	P5	PC4	E3	SCK0	H3
NC	P14	PC5	E1	SCK1	G1
$\overline{\text{NMI}}$	D1	PD0	F2	SCLK	G2
PB0	M5	PD1	A2	SRD0	E3
PB1	P4	PD2	B2	SRD1	B1
PB10	M2	PD3	G1	STD0	E1
PB11	J2	PD4	B1	STD1	C2
PB12	J3	PD5	C2	$\overline{\text{TA}}$	P10
PB13	L1	PE0	F1	TCK	C3
PB14	K2	PE1	G3	TDI	B3
PB15	J1	PE2	G2	TDO	A4
PB2	N4	PINIT	D1	TIO0	L3
PB3	P3	$\overline{\text{RD}}$	M12	TIO1	L2

```
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 DSP56321 interrupts
;
;*************************************************************************

       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
```

```
;---------------------------------------------------------------------
; 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
```