Functional Differences Between The 1F90S Mask of the DSP56302 and the 0J17D Mask of the DSP56309

This document summarizes the differences between the DSP56302 (1F90S mask revision) and the DSP56309, Rev A (0J17D mask revision). Rev A of the DSP56309 is functionally compatible with the DSP56302, allowing seamless transition to the higher-performance DSP56309. The 0J17D mask of the DSP56309 uses the communications design rules (CDR2) process. The 0J17D mask set also has new I/O and a new PLL, with the requisite change in the PLL capacitor equation (PCAP).

CONTENTS

1 PLL Input Capacitor

The process change results in a changed requirement for computing the size of C_{PCAP}, the capacitor used with the PCAP input. lists the new formulas for computing the value of this input capacitor for the DSP56309 Rev A.

2 ESD Protection

The DSP56309 0J17D mask set incorporates improved I/O structures to ensure Electro Static Discharge (ESD) protection to the Freescale MC qualification levels of 2000 volts Human Body Model (HBM) and 200 volts Machine Model (MM).

3 Differences Overview

The primary functional differences between the DSP56302 and the Rev A of the DSP56309 are due to inherent differences between the two design technologies. **Table 1** compares these two products.

Table 1. Functional Comparison of the DSP56302 and Rev A of the DSP56309

Feature	DSP56302	DSP56309, Rev A
Operating frequency	up to 66 MHz	up to 100 MHz
Technology	0.5 micron	0.32 micron
Input power	V_{CC} = 3.0-3.6 V combined core and I/O power and ground	Split power: • Core V_{CC} (3.0–3.6 V) • I/O V_{CC} (3.0–3.6 V) A pinout change is required to support the split power configuration. See **Section 4** for more information and a description of the pinout change for the 144-pin TQFP package.
Package	144-pin TQFP	144-pin TQFP or 196-pin PBGA
PLL input capacitor (C_{PCAP})	Uses the following rules: • For MF ≤ 4: C_{PCAP} = [(500 × MF) – 150] pF • For MF > 4: C_{PCAP} = (690 × MF) pF	Uses the following rules (recommended): • For MF ≤ 4: C_{PCAP} = [(680 × MF) – 120] pF • For MF > 4: C_{PCAP} = (1100 × MF) pF
Operating modes		See **Table 4** for details.

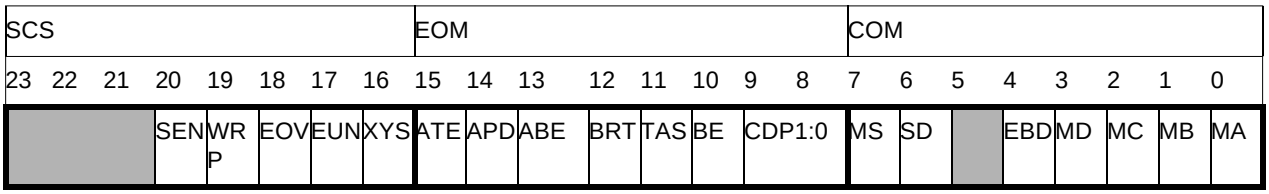
4 Input Power Changes

The DSP56309 Rev A uses a split-power design and therefore requires a modification in the chip pinout. A top view of the DSP56309 Rev A TQFP package is shown in **Figure 4**. **Table 3** lists the pin differences between the DSP56302 and the DSP56309.

Note: The power input for the core logic is designated as V_{CCQ} for the DSP56302. For the DSP56309, the independent core logic input voltage is designated as V_{CCQL}, while the independent I/O input voltage is designated as V_{CCQH}. V_{CCQL} should connect to the core input power supply. V_{CCQH} and all other input power (V_{CCA}, V_{CCC}, V_{CCD}, V_{CCH}, V_{CCP}, and V_{CCS}) should connect to the external input power supply.

5 Asynchronous Bus Arbitration

In the Operating Mode Register (OMR) of the 0J17D is a new Asynchronous Bus Arbitration (ABE) bit to support external access at high frequencies. As **Figure 1** shows, bit 13 in the OMR enables Asynchronous Bus Arbitration. For convenience, **Table 2** defines all the OMR bits available in the 0J17D mask set of the DSP56309. The ABE bit is highlighted in a bold red font.



ATE—Address Trace Enable

APD—Address Attribution Priority Disable

ABE—Async. Bus Arbitration Enable

BRT—Bus Release Timing

TAS—$\overline{\text{TA}}$ Synchronize Select

BE—Burst Mode Enable

CDP1—Core-DMA Priority 1

CDP0—Core-DMA Priority 0

MS—Memory Switch Mode

SD—Stop Delay

EBD—External Bus Disable

MD—Operating Mode D

MC—Operating Mode C

MB—Operating Mode B

MA—Operating Mode A

SEN—Stack Extension Enable

WRP—Extended Stack Wrap Flag

EOV—Extended Stack Overflow Flag

EUN—Extended Stack Underflow Flag

XYS—Stack Extension Space Select

—Reserved bit; read as zero; should be written with zero for future compatibility

Figure 1. Operating Mode Register (OMR)

Table 2. Operating Mode Register (OMR) Bit Definitions

Bit Number	Bit Name	Reset Value	Description
23–21		0	Reserved. Set to 0 for future compatibility.
20	SEN	0	**Stack Extension Enable** Enables/disables the stack extension in data memory. If the SEN bit is set, the extension is enabled. Hardware reset clears this bit, so the default out of reset is a disabled stack extension.
19	WRP	0	Stack Extension Wrap Flag Set when copying from the on-chip hardware stack (System Stack Register file) to the stack extension memory begins. You can use this flag during the debugging phase of the software development to evaluate and increase the speed of software-implemented algorithms. The WRP flag is a *sticky bit* (that is, cleared only by hardware reset or by an explicit MOVE operation to the OMR).
18	EOV	0	Stack Extension Overflow Flag Set when a stack overflow occurs in Stack Extended mode. Extended stack overflow is recognized when a push operation is requested while SP = SZ (Stack Size register), and the Extended mode is enabled by the SEN bit. The EOV flag is a *sticky bit* (that is, cleared only by hardware reset or by an explicit MOVE operation to the OMR). The transition of the EOV flag from zero to one causes a Priority Level 3 (Non-maskable) stack error exception.
17	EUN	0	**Stack Extension Underflow Flag** Set when a stack underflow occurs in Extended Stack mode. Extended stack underflow is recognized when a pull operation is requested, SP = 0, and the SEN bit enables Extended mode. The EUN flag is a *sticky bit* (that is, cleared only by hardware reset or by an explicit MOVE operation to the OMR). Transition of the EUN flag from zero to one causes a Priority Level 3 (Non-maskable) stack error exception. NOTE: While the chip is in Extended Stack mode, the UF bit in the SP acts like a normal counter bit.

Table 2. Operating Mode Register (OMR) Bit Definitions (Continued)

Bit Number	Bit Name	Reset Value	Description
16	XYS	0	**Stack Extension XY Select** Determines whether the stack extension is mapped onto X or Y memory space. If the bit is clear, then the stack extension is mapped onto the X memory space. If the XYS bit is set, the stack extension is mapped to the Y memory space.
15	ATE	0	**Address Trace Enable** When set, the Address Trace Enable (ATE) bit enables Address Trace mode. The Address Trace mode is a debugging tool that reflects internal memory accesses at the external address lines.
14	APD	0	**Address Attribution Priority Disable** Disables the priority assigned to the Address Attribute signals (AA0-AA3). When APD = 0 (default setting), the four Address Attribute signals each have a certain priority: AA3 has the highest priority, AA0 has the lowest priority. Therefore, only one AA signal can be active at one time. This allows continuous partitioning of external memory; however, certain functions, such as using the AA signals as additional address lines, require the use of additional interface hardware. When APD is set, the priority mechanism is disabled, allowing more than one AA signal to be active simultaneously. Therefore, the AA signals can be used as additional address lines without the need for additional interface hardware.
13	**ABE**	**0**	**Asynchronous Bus Arbitration Enable** Eliminates the setup and hold time requirements (with respect to CLKOUT) for $\overline{BB}$ and $\overline{BG}$, and substitutes a required non-overlap interval between the deassertion of one $\overline{BG}$ input to a DSP56300 family device and the assertion of a second $\overline{BG}$ input to a second DSP56300 family device on the same bus. When the ABE bit is set, the $\overline{BG}$ and $\overline{BB}$ inputs are synchronized. This synchronization causes a delay between a change in $\overline{BG}$ or $\overline{BB}$ until this change is actually accepted by the receiving device.
12	BRT	0	**Bus Release Timing** Selects between fast or slow bus release. If BRT is cleared, a Fast Bus Release mode is selected (that is, no additional cycles are added to the access and BB is not guaranteed to be the last Port A pin that is tri-stated at the end of the access). If BRT is set, a Slow Bus Release mode is selected (that is, an additional cycle is added to the access, and BB is the last Port A pin that is tri-stated at the end of the access).
11	TAS		**TA Synchronize Select** Selects the synchronization method for the input Port A pin—$\overline{TA}$ (Transfer Acknowledge). If TAS is cleared, you are responsible for asserting the $\overline{TA}$ pin in synchrony with the chip clock, as described in the technical data sheet. If TAS is set, the $\overline{TA}$ input pin is synchronized inside the chip, thus eliminating the need for an off-chip synchronizer. Note that the TAS bit has no effect when the $\overline{TA}$ pin is deasserted: you are responsible for deasserting the **TA** pin in synchrony with the chip clock, regardless of the value of TAS.
10	BE	0	**Cache Burst Mode Enable** Enables/disables Burst mode in the memory expansion port during an instruction cache miss. If the bit is cleared, Burst mode is disabled and only one program word is fetched from the external memory when an instruction cache miss condition is detected. If the bit is set, Burst mode is enabled, and up to four program words are fetched from the external memory when an instruction cache miss is detected.
9 – 8	CPD	1	**Core-DMA Priority** Specify the priority of core and DMA accesses to the external bus. 00 — Determined by comparing status register CP[1:0] to the active DMA channel priority 01 — DMA accesses have higher priority than core accesses 10 — DMA accesses have the same priority as the core accesses 11 — DMA accesses have lower priority than the core accesses

Table 2. Operating Mode Register (OMR) Bit Definitions (Continued)

Bit Number	Bit Name	Reset Value	Description
7	MS	0	Memory Switch Mode Allows some internal data memory (X, Y, or both) to become part of the chip internal Program RAM. **Notes:** **1.** Program data placed in the Program RAM/Instruction Cache area changes its placement after the MS bit is set (that is, the Instruction Cache always uses the highest internal Program RAM addresses). **1.** To ensure proper operation, place six NOP instructions after the instruction that changes the MS bit. **2.** To ensure proper operation, do not set the MS bit while the Instruction Cache is enabled (CE bit is set in SR).
6	SD	0	**Stop Delay Mode** Determines the length of the delay invoked when the core exits the Stop state. The STOP instruction suspends core processing indefinitely until a defined event occurs to restart it. If SD is cleared, a 128K clock cycle delay is invoked before a STOP instruction cycle continues. However, if SD, the delay before the instruction cycle continues is 16 clock cycles. The long delay allows a clock stabilization period for the internal clock to begin oscillating and to stabilize. When a stable external clock is used, the shorter delay allows faster start-up of the DSP56300 core.
5		0	Reserved. Set to 0 for future compatibility.
4	EBD	0	**External Bus Disable** Disables the external bus controller to reduce power consumption when external memories are not used. When EBD is set, the external bus controller is disabled and external memory cannot be accessed. When EBD is cleared, the external bus controller is enabled and external access can be performed. Hardware reset clears the EBD bit.
3–0	MD–MA		**Chip Operating Mode** Indicate the operating mode of the DSP56300 core. On hardware reset, these bits are loaded from the external mode select pins, MODD, MODC, MODB, and MODA, respectively. After the DSP56300 core leaves the Reset state, MD, MC, MB, and MA can be changed under program control.

6 Identification Registers

Values in the Device Identification register (IDR) and JTAG Identification (ID) register are changed to reflect the new 0J17D mask set, as shown in **Figure 2** and **Figure 3**. The IDR is a 24-bit, read-only factory-programmed register that identifies DSP56300 family members. It specifies the derivative number and revision number of the device. This information is used in testing or by software. **Figure 2** shows the contents of the IDR for mask set 0J17D. Revision numbers are assigned as follows: $0 is revision 0, $1 is revision A, and so on.

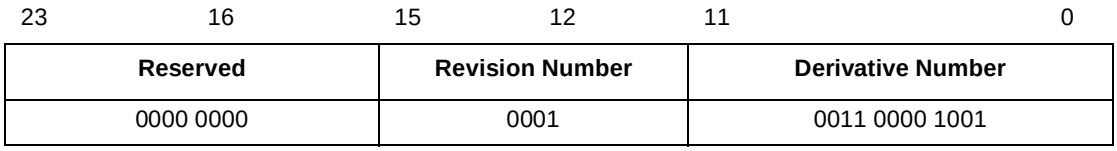


Figure 2. Identification Register Configuration (Revision 3)

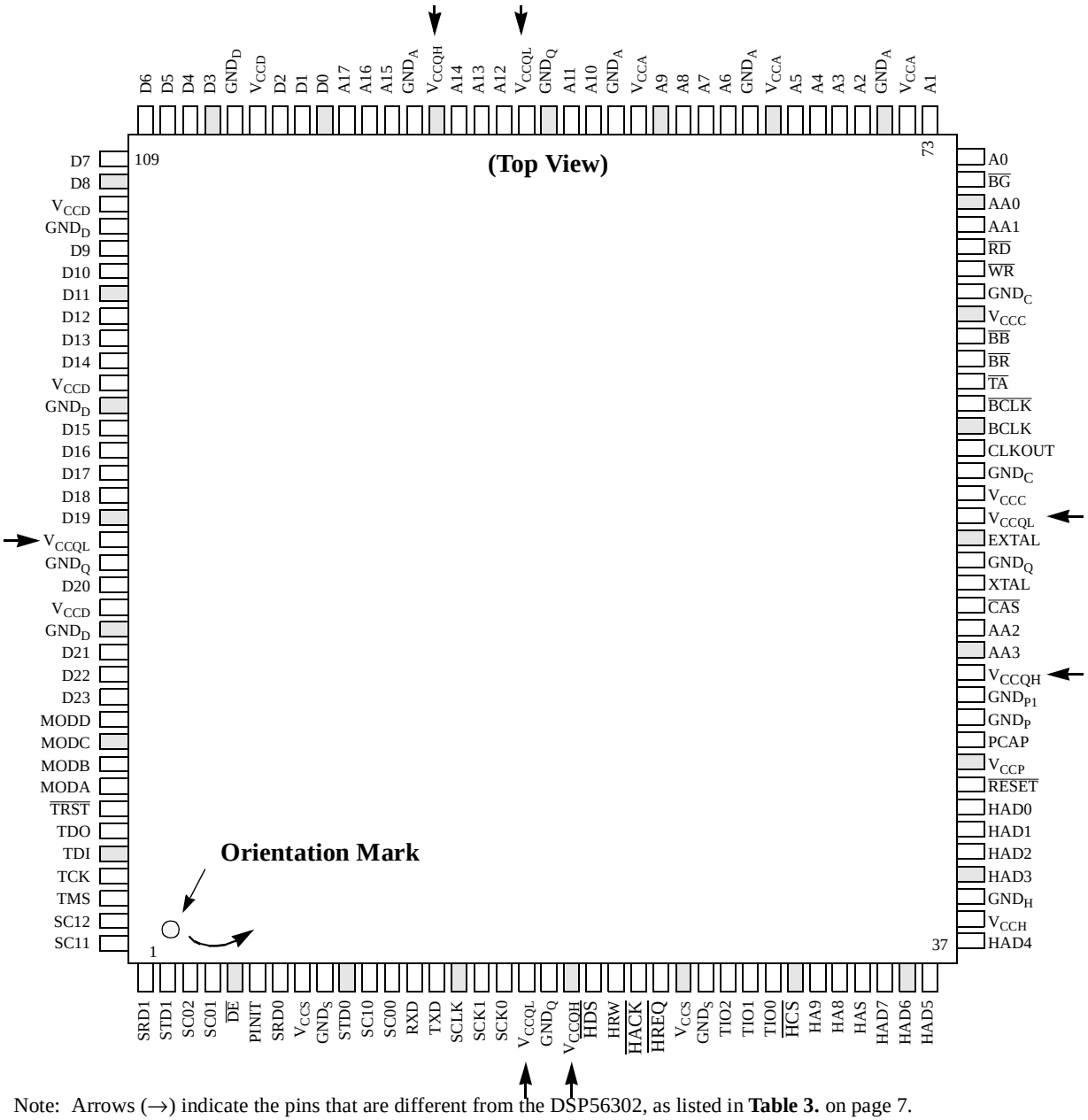
The JTAG ID register is a 32-bit read-only factory-programmed register that distinguishes the component on a board according to the IEEE 1149.1 standard. **Figure 3** shows the JTAG ID register configuration for mask set 0J17D. Version information corresponds to the revision number ($0 for revision 0, $1 for revision A, and so forth).

31 28	27 22	21 12	11 1	0
Version Information	**Customer Part Number**	**Sequence Number**	**Manufacturer Identity**	**1**
0001	000110	0000001001	000 0000 1110	1

Figure 3. JTAG Identification Register Configuration (Revision 3)

7 Package and Mechanicals

This section shows a view of the DSP56309 Thin Quad Flat Pack and summarizes the differences between the DSP56302 and DSP56309, Rev A packages.



Note: Arrows (→) indicate the pins that are different from the DSP56302, as listed in **Table 3.** on page 7.

Figure 4. DSP56309 Thin Quad Flat Pack (TQFP), Top View

Table 3. Pin Differences between DSP56302 and DSP56309 Rev A (144-pin TQFP Package)

Pin	Pin Name	
	DSP56302	DSP56309 Rev A
18	V_{CCQ}	V_{CCQL}
20	NC	V_{CCQH}
49	NC	V_{CCQH}
56	V_{CCQ}	V_{CCQL}
91	V_{CCQ}	V_{CCQL}
95	V_{CCA}	V_{CCQH}
126	V_{CCQ}	V_{CCQL}

Notes:
1. V_{CCQ} = input voltage for core logic
3. V_{CCQL} = independent input voltage for core logic
2. NC = not connected
3. V_{CCQH} = independent input voltage for I/O lines
4. V_{CCA} = voltage for external address lines
5. Unlisted pins are the same for both chips.

A pinout for the 196-pin PBGA package is included in the *DSP56309 Technical Data Sheet*. This package includes the split power configuration described for the 144-pin TQFP package.

8 Operating Modes

The operating modes of the DSP56302 are documented in the *DSP56302 User's Manual*. **Table 4.** shows the operating modes of the DSP56309, Rev A. Modes that differ from those of the DSP56302 are highlighted in the table.

Table 4. DSP56309 Operating Modes

Mode	MODD	MODC	MODB	MODA	Reset Vector	Description
0	0	0	0	0	$C00000	Expanded mode
1	0	0	0	1	$FF0000	Reserved
2	0	0	1	0	$FF0000	Reserved
3	0	0	1	1	$FF0000	Reserved
4	0	1	0	0	$FF0000	Reserved
5	0	1	0	1	$FF0000	Reserved
6	0	1	1	0	$FF0000	Reserved
7	0	1	1	1	$FF0000	Reserved
8	1	0	0	0	$008000	Expanded mode
9	1	0	0	1	$FF0000	Boot from byte-wide memory
A	1	0	1	0	$FF0000	Boot from SCI
B	1	0	1	1	$FF0000	Reserved
C	1	1	0	0	$FF0000	HI08 bootstrap in ISA mode
D	1	1	0	1	$FF0000	HI08 bootstrap in HC11 non-multiplexed mode

Table 4. DSP56309 Operating Modes

Mode	MODD	MODC	MODB	MODA	Reset Vector	Description
E	1	1	1	0	$FF0000	HI08 bootstrap in 8051 multiplexed bus mode
F	1	1	1	1	$FF0000	HI08 bootstrap in MC68302 mode

9 DSP56302 Silicon Errata

This section lists the silicon errata for mask 1F90S of the DSP56302, which have been fixed on the DSP56309, Rev A.

Errata Number	Errata Description	Status
ES27	Description (added 10/24/1996): If the chip is in the Debug mode and the $\overline{\text{RESET}}$ pin is asserted to bring the chip into a normal execution mode without asserting $\overline{\text{TRST}}$ at the same time, then a subsequent read of the chip status through the JTAG port shows the chip to be in the Debug mode instead of the expected User mode. Workaround: Assert the $\overline{\text{TRST}}$ pin whenever the $\overline{\text{RESET}}$ pin is asserted.	**Fixed on DSP56309, Rev A**
ES28	Description (added 10/24/1996): If the chip is in the Debug mode and the $\overline{\text{TRST}}$ pin is asserted (without asserting $\overline{\text{RESET}}$), then a subsequent read of the chip status through the JTAG port shows the chip to be in the User mode instead of the expected Debug mode. Workaround: Execute the following JTAG commands before reading the JTAG status: a) Enable OnCE b) DEBUG request After executing these commands, the status bits will reflect the actual status of the chip and the $\overline{\text{DE}}$ pin will acknowledge "re-entering" the Debug mode.	**Fixed on DSP56309, Rev A**
ES29	Description (added 10/24/1996): When SCI transmitter is used in Synchronous mode, the last bit of the transmitted byte may be truncated to the half of the serial cycle. Workaround: Not available.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status
ES30	Description (added 11/18/1996): After the $\overline{BB}$ pin output is driven high and released, the pin output voltage level may not reach V_{CC}. The issue depends on the application board layout and the parameters of the chip process. Workaround: Use a restricted board layout that includes a 1 kΩ pull-up resistor connected to the $\overline{BB}$ pin with a 100 Ω resistor connected in series with, and as close as possible to, the pin. The board route from the $\overline{BB}$ pin to any component should guarantee the following parameters: a. Route inductance < 40 nH b. Route capacitance < 15 pF c. Input capacitance < 8 pF Such restrictions guarantee that when $\overline{BB}$ is driven high (deasserted), the output voltage level will be above 2.25 V at V_{CC} = 3.3 V.	**Fixed on DSP56309, Rev A**
ES33	Description (added 3/3/1997): When using the JTAG instructions SAMPLE/PRELOAD, EXTEST, and CLAMP, erroneous data may be driven out on the parallel pins and TDO. Data cannot be shifted through the Boundary Scan Register (BSR) using the SAMPLE/PRELOAD instruction. Because the BSR must be preloaded using the SAMPLE/PRELOAD instruction, the EXTEST and CLAMP instructions cannot be used for testing the board connections. Workaround: None available.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status
ES36	Description (added 4/7/1997): If the stack extension is enabled, the instructions listed below should not be placed as the next-to-last or as the last instruction of a DO loop (that is, should not appear at LA-1 or LA). The instructions are: XY Memory Data Move (A-6.76) X Memory Move (A-6.71) Y Memory Move (A-6.73) Long Memory Data Move (A-6.75) Immediate Short Data Move (A-6.68) Register to Register Data Move (A-6.69) Address Register Update (A-6.70) X Memory and Register Data Move (A-6.72) Y Memory and Register Data Move (A-6.74) Arithmetic Instructions that allow Parallel Moves listed above IFcc and IFcc.U (A-6.41) Workaround: Insert a NOP or other instruction not listed above as the next-to-last and last instructions in the DO loop.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status
ES46	Description (added 3/3/1997): When a DMA controller is in a mode that clears $\overline{DE}$ (that is, TM = 0xx), if the core performs an external access with wait states or there is a transfer stall (see Appendix B, Section B.3.4.2 in the DSP56300 Family Manual) or a conditional transfer interlock (see Appendix B, Section B.3.5.1) during the last DMA channel transfer, there will be one additional DMA word transfer. Workaround: There are three system-dependent workarounds for this problem. The user should test the system using these workarounds to determine which one to use in the particular system to overcome this problem. The workarounds are: Workaround 1: a. Prepare one additional memory word in the source and destinations buffers. This data should be ignored. b. Activate a DMA Interrupt Service Routine (ISR) or poll the DTD bit to ensure block transfer completeness. In the DMA ISR or the handler routine after status polling, reload the values of the address registers. Workaround 2: a. Use a DMA mode that does not clear DE (that is, TM = 1xx) and activate the DMA interrupt. b. In the ISR, execute the following operations in the order listed: clear DE, update the address registers, and set DE. Workaround3: a. Use a DMA mode that does not clear DE (that is, TM = 1xx). b. Change the address mode from linear addressing to 2D or from 2D to 3D and use an offset register to update the address automatically at the end of the block. **Note:** If the user can not use one of these workarounds, there may be other possible system-dependent workarounds.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status

ES47	Description (added 3/3/1997):
	
	If the DMA channel and the core access the same 1/4 K internal X data, Y data, or program memory page, and the DMA interrupt is enabled, a false interrupt may occur in addition to the correct one.
	
	Workaround: Ensure that the channel's DTD status bit in the DSTR is set before jumping to the Interrupt Service Routine (that is, the interrupt is correct only when DTD is set).
	
	Example:
	
	`ORG P:I_DMA0`
	`JSSET #M_DTD0,X:M_DSTR,ISR_ ; ISR_ is the Interrupt Service`
	`; Routine label for DMA channel 0`
		Fixed on DSP56309, Rev A

ES48	Description (added 4/7/1997; modified 7/7/1997): **Note:** This is a subset of Errata # 46 (that is, in every case that errata # 48 occurs, errata # 46 occurs, but not vice versa). When a DMA controller is in a mode that clears $\overline{DE}$ (that is, TM = 0xx), and it transfers data to an external memory with two or more wait states, and the DSP core performs an external access with wait states or there is a transfer stall (see Appendix B, Section B.3.4.2 in the DSP56300 Family Manual) or a conditional transfer interlock (see Appendix B, Section B.3.5.1) during the last DMA channel transfer, the destination pointer for a subsequent DMA transfer may not be reprogrammed correctly. There are two defined workarounds to prevent the occurrence of this condition and one recovery code that should be used if the workarounds can not be used in a specific system: Workaround 1: a. Use a DMA mode that does not clear DE (that is, TM = 1xx) and activate the DMA interrupt. b. In the DMA ISR, clear DE, update the address registers, and set DE.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status

ES48 Cont.	Description (added 4/7/1997; modified 7/7/1997):
	
	Workaround 2:
	
	a. Use a DMA mode that does not clear DE (that is, TM = 1xx).
	
	b. Change the address mode from linear addressing to 2D or 2D to 3D and use an offset register to update the address automatically at the end of the block.
	
	Recovery (to recover if the condition occurs):
	
	a. Enable the DMA interrupt.
	
	b. Use the following code in the DMA ISR:
	
	`movep #dummy_source, x:M_DSRi`
	`movep #dummy_dest, x:M_DDRi`
	`movep #0, x:M_DEOi`
	`movep #9E0240, x:M_DCRi ; initiate one dummy transfer`
	`; if the bug occurred, the`
	`; transfer will be to the`
	`; old_block_last_dest + 1`
	`; and not to the dummy_dest`
	`nop`
	`nop`
	`jclr #M_DTDi, x:M_DSRT,*`
	`... ; update the registers for`
	`; new block.
		Fixed on DSP56309, Rev A

Errata Number	Errata Description	Status
ES53	Description (added 9/25/1997): Using the JTAG instruction code 1111 ($F) or 1101 ($D) for the BYPASS instruction may cause the chip to enter Debug mode (which then correctly sets the Status bits (OS[1:0]) in the OnCE Status and Control Register (OSCR[7:6]) and asserts the $\overline{DE}$ output to acknowledge the Debug mode status). Workaround: Use one of the following alternatives: a. If possible, do not use instruction code 1111 ($F) or 1101 ($D) for the BYPASS instruction. Use one of the other defined BYPASS instruction codes (that is, any code from 1000–1100 ($8–$C) or 1110 ($E)). b. If you must use instruction code 1111 ($F) or 1101 ($D), use the following procedure: — While the $F or $D instruction code is in the Instruction Register, ensure that the JTAG Test Access Port (TAP) state machine does not pass through the JTAG Test-Logic-Reset state while accessing any JTAG registers (that is, Instruction Register, Boundary Scan Register, or ID Register). — Before using any other JTAG instruction, load one of the other BYPASS instruction codes (that is, any code from 1000–1100 ($8–$C) or 1110 ($E)) into the instruction register. Then, any other JTAG instruction may be used.	**Fixed on DSP56309, Rev A**
ES54	Description (added 1/27/98): When a DMA channel is configured using its DMA Control Register (DCR) in the following manner: • Line Transfer mode is selected (DTM[2:0] = 010) • Non-Three-Dimensional Address mode is selected (D3D = 0) • Destination Address Offset Register DOR1 or DOR3 is selected (DAM[5:3] = 001 or 011) • No Source Address Offset is selected (DAM[2:0] = 100 or 101) The DMA transfer does not function as intended. Workaround: Select Destination Address Offset Register DOR0 or DOR2 by setting DAM[5:3] = 000 or 010.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status
ES64	Description (added 10/24/1996): When using the 5 V tolerant pin in open drain mode (that is, $\overline{DE}$, $\overline{HREQ}$/$\overline{HTRQ}$, $\overline{HACK}$/$\overline{HRRQ}$ and TXD), the chip clamps the voltage at the pin to about V_{CC} + 0.4 V. Workaround: Not available.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status

ES84	Description (added 5/13/98):
	
	When software disables a DMA channel (by clearing the DE bit of the DCR) , the DTD status bit of the channel may not be set if any of the following events occur:
	
	a. Software disables the DMA channel just before a conditional transfer stall (Described by App B-3.5.1,UM).
	
	b. Software disables the DMA channel at the end of the block transfer (that is after the counter is loaded with its initial value and transfer of the last word of the block is completed).
	
	As a result, the Transfer Done interrupt might not be generated.
	
	Workaround:
	Avoid using the instruction sequence causing the conditional transfer stall (See DSP56300 UM, App B-3.5.1 for description) in fast interrupt service routines. Every time the DMA channel needs to be disabled by software, the following sequence must be used :
	
	`bclr #DIE,x:M_DCR ; not needed if DIE is cleared`
	`bclr #DE,x:M_DCR`
	`; instead of two instructions above, one 'movep' instruction may be used`
	`; to clear DIE and DE bits`
	`movep #DCR_Dummy_Value,x:M_DCR`
	`bclr #DE,x:M_DCR`
	`nop`
	`nop`
	
	Here, the `DCR_Dummy_value` is any value of the DCR register that complies with the following requirements:
	
	• DE is set;
	• DIE is set if Transfer Done interrupt request should be generated and cleared otherwise;
	• DRS[4:0] bits must encode a reserved DMA request source (see the following list of reserved DRS values);
	
	List of reserved DRS[4:0] values (per device):
	
	• DSP56302, DSP56309, DSP56303, DSP56304, DSP56362 — `10101-11111`
	• DSP56305 — `11011`
	• DSP56301 — `10011-11011`
	• DSP56307 — `10111-11111`
		Fixed on DSP56309, Rev A

Errata Number	Errata Description	Status
ES85	Description (added 5/3/98): If both the DMA channel and the core simultaneously access the same 1/4K page of internal memory (X, Y, or program), an improper DMA channel operation may occur. Workaround: Avoid simultaneous DMA and core accesses to the same 1/4K page of internal memory.	**Fixed on DSP56309, Rev A**
ES89	Description (added 6/25/98): If the SCI Receiver is programmed to work with a different serial clock than the SCI Transmitter so that either the Receiver or Transmitter is using the external serial clock and the other is using the internally-generated serial clock—RCM and TCM in the SCCR are programmed differently)—then the internal serial clock generator will not operate and the SCI portion (Receiver or Transmitter) clocked by the internal clock will be stuck. Workaround: Do not use SCI with the two SCI portions (Receiver and Transmitter) clocked by different serial clocks; use either both externally or both internally clocked. Or: When using both portions of the SCI (Receiver & Transmitter), do not program different values on RCM and TCM in the SCCR.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status
ES90	Description (added 6/25/98)/Modified 4/19/99: A deadlock occurs during DMA transfers if all the following conditions exist: 1. DMA transfers data between internal memory and external memory through port A. 2. DMA and the core access the same internal 0.25K memory module. 3. One of the following occurs: a. The bus arbitration system is active, that is, $\overline{BG}$ is changing, not tied to ground. b. Packing mode (bit 7 in the AAR[3–0] registers) is active for DMA transfers on Port A. Workaround: One of the following, but workarounds 2, and 3 are valid ONLY to section 3 a of the errata—that is, not valid if packing mode is used, and workaround 4 is valid only to section 3 b of the errata—that is, not valid if bus arbitration is active. 1. Use intermediate internal memory on which there is no contention with the core. 2. Tie $\overline{BG}$ to ground, or have an external arbiter that asserts $\overline{BG}$ even if BR is not asserted. 3. Set the BCR[BRH] bit, whenever BR must be active. 4. Avoid using packing mode.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status

ES91	Description (added 7/22/98):
	
	If the Core reads data from the HRX while instructions are fetched from the memory Expansion Port (Port A) using 2 or more wait states, data may be lost.
	
	Workaround :
	
	There are three possible workarounds:
	
	1) The host should guarantee that there is no more than one word in the TXH:TXM:TXL-HRX data path at any time. This can be achieved if the host writes a word to the HI08 only when the TRDY flag is set (that is, the data path is empty).
	
	2) Use a service routine running from fast (that is, one wait state) external memory or internal memory to read the HRX read code; ensure that code that is fetched from slow (that is, more than 1 wait state) external memory is located at least 4 instructions after the HRX register is read. For example:
	
	`READ_HRX_DATA`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	
	Note:
	a) Interrupt requests that fetch instructions from slow external memory should be masked during this service routine. Nonmaskable interrupt (NMI) request routines must not be in external memory.
	
	b) If running from fast external memory and if a DMA channel accessing external memory is used, then the DMA may cause extra wait states to the core. Thus, the DMA should have a lower priority than the core so that the core can access the external memory with no more than 1 wait state.
	
	3) Read the HRX using one of the channels of the on-chip DMA controller.
		Fixed on DSP56309, Rev A

Errata Number	Errata Description	Status
ES95	Description (added 8/15/98): If more than a single DMA channel is enabled while the DSP stays in the WAIT processing state, and triggering one of the DMA channels causes an exit from the WAIT state (See A-6.115, UM), triggering another DMA channel might cause improper DMA operation. Workaround: Assure that only a single DMA channel can be triggered during DSP WAIT state. If the application cannot guarantee this, other DMA channels should be disabled before the WAIT processing state is entered and then reenabled after WAIT state is exited.	**Fixed on DSP56309, Rev A**
ES104	Description: (added 11/24/98): An improper operation may occur when all the following conditions apply: • The DMA channel is in a mode that does not automatically clear the DE bit at the end of the block (DTM[2:0] = 1xx in DCR). • This channel is disabled by software (by clearing DE in DCR) while it is triggered for a new transfer. • The previous operation is not yet completed. Workaround: The DMA channel should be disabled only when it is not triggered for a new transfer, that is, when the DACT bit in the DSTR register is cleared. **Note:** To perform this operation most efficiently, all other DMA channels should be disabled.	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status

ES105	Description (added 11/25/98):
	
	If the core clears HCIE bit on HCR register, while the interrupt was issued, and the vector was read by the interrupt controller, then when the interrupt is serviced, HCP will not be cleared, since the clear equation is conditioned by HCIE=1.
	
	Workaround :
	
	There are two possible workarounds:
	
	1) If only host commands are used as possible interrupt source to the CORE (that is, HTIE and HRIE are both 0), then instead of bit-set and bit-clear to HCIE, do bit-set and bit-clear instructions on IPRP register for the HIE bit.
	
	2) If option "1" can not be used, then the user should first turn off host interrupt requests (all the possible sources) by clearing HIE bit on IPRP register, then issue 6 NOP instructions, then clear HCIE bit on HCR, issue another 6 NOP instructions and finally re-enable the HIE bit on IPRP as shown below:
	
	`;; Clear the relevant bits on IPRP register acording`
	`;; to the current host interrupt priority settings`
	
	`BCLR #M_HPL0,x:M_IPRP`
	`BCLR #M_HPL1,x:M_IPRP`
	`;; Issue 6 NOP instructions`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	`;; Clear the HCIE bit on HCR register to turn off host commands`
	`BCLR #M_HCIE,x:M_HCR`
	`;; Issue 6 NOP instructions`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	`NOP`
	`;; Restore the required host interrupt level`
	`BSET #M_HPL0,x:M_IPRP`
	`BSET #M_HPL1,x:M_IPRP`
		Fixed on DSP56309, Rev A

Errata Number	Errata Description	Status
ES114	Description (added 4/19/99, revised 4/30/99): A DMA channel may operate improperly when the address mode of this channel is defined as three-dimensional (D3D=1) and DAM[5:0] = 1xx 1 10 or DAM[5:0] = 01xx 10 (that is, triple counter mode is E). Workaround: Use the triple counter modes C(DAM[1:0]=00) or D(DAM[1:0]=01) instead of the E(DAM[1:0]=10) mode.	**Fixed on DSP56309, Rev A**

ES115	Description (added 4/19/99):
	
	When a DMA channel (called channel A) is disabled by software clearing the channel's DCR[DE] bit, the DTD bit may not get set, and the DMA end of the block interrupt may not happen if one of the following occurs:
	
	1. There is another channel (channel B) executing EXTERNAL accesses, and the DE bit of channel A is being cleared by software at the end of the channel B word transfer—if channel B is in Word transfer mode, or at the end of the channel B line transfer—if channel B is in Line Transfer mode, or at the end of the channel B block transfer—if channel B is in Block transfer mode.
	
	2. This channel (A) is executing EXTERNAL accesses, and the DE bit of this channel (A) is being cleared by software at the end of the channel B word transfer—if channel B is in Word transfer mode, or at the end of the channel B line transfer—if channel B is in Line transfer mode.
	
	Workaround:
	
	Avoid executing a DMA external access when any DMA channel should be disabled. This can be done as follows. Every time the DMA channel needs to be disabled by software, the following sequence must be used:
	
	`;; initialize an unused DMA channel "C"`
	`movep #DSR_swflag, x:M_DSRC ;; here DSR_swflag is an`
	`;; unused X, Y or P memory`
	`;; location, should`
	`;; be initialized to`
	`;; $800000`
	`;; M_DSRC - address of the`
	`;; channel C DSR register.`
		Fixed on DSP56309, Rev A

Errata Number	Errata Description	Status
ES115 cont.	movep #DDR_swflag, x:M_DDRC ;; DDR_swflag is an unused ;; X, Y or P memory ;; location, should be ;; initialized to $000000 ;; M_DDRC - ;; address of the channel C ;; DDR register . movep #TR_LENGTH, x:M_DCOC ;; see below the definition ;; of the TR_LENGTH value, ;; M_DCOC - address ;; of the channel C DCO register .movep #1f0240, x:M_DCRC ;; M_DCRB - address of the ;; channel C DCR register. ;; Set transfer mode - ;; block transfer, ;; triggered by ;; software highest ;; priority, continuous ;; mode on no-update ;; source and destination ;; address mode X memory ;; location for source ;; and destination (can be ;; chosen by ;; user accordingly to ;; DSR_swflag/DDR_swflag)	**Fixed on DSP56309, Rev A**

Errata Number	Errata Description	Status
ES115 cont.	;; disable DMA channel "A" ori #3, mr ;; mask all interrupts bset #23, x:M_DCRC ;; enable DMA channel C bclr #23,x:DDR_swflag,* ;; wait until DMA channel C ;; begin transfer bclr #23, x:M_DCRA ;; disable DMA channel A nop nop jclr #M_DTDA, x:M_DSTR,* ;; polling DTD bit of the DMA ;; channel A, The TR_LENGTH value can be defined as the maximum length of the external DMA transfer—from the length of the read DMA cycle and from the length of the write DMA cycle. The length of the external read/write DMA cycle can be defined as the length of the PORTA external access. The length of the internal read/write DMA cycle can be defined in the errata case as 2 DSP clock cycles. The TR_LENGTH can be found as sum of the lengths of the DMA read and DMA write cycles.	Fixed on DSP56309, Rev A

10 DSP56309 Silicon Errata

This section lists all known silicon and documentation errata for Rev A of the DSP56309 (mask 0J17D).

Table 5. Silicon Errata for DSP56309, Rev A (Mask 0j17D)

Errata Number	Errata Description
	None Known

11 Common Document Errata

This section lists the documentation errata that are present on both the DSP56302 Rev A (mask 0F90S) and the DSP56309 Rev A (mask 0J17D).

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D)

Errata Number	Document Update	Applies to Mask

ED1	Description (revised 11/9/98):
	
	XY memory data move does not work properly under one of the following two situations:
	
	1. The X-memory move destination is internal I/O and the Y-memory move source is a register used as destination in the previous adjacent move from non Y-memory
	
	2. The Y-memory move destination is a register used as source in the next adjacent move to non Y-memory.
	
	Here are examples of the two cases (where x:(r1) is a peripheral):
	
	Example 1:
	
	`move #$12,y0`
	`move x0,x:(r7) y0,y:(r3) (while x:(r7) is a peripheral).`
	
	Example 2:
	
	`mac x1,y0,a x1,x:(r1)+ y:(r6)+,y0`
	`move y0,y1`
	
	Any of the following alternatives can be used:
	
	a. Separate these two consecutive moves by any other instruction.
	
	b. Split XY Data Move to two moves.
	
	Pertains to: DSP56300 Family Manual, Section B-5 "Peripheral pipeline restrictions.
		0J17D and 1F90S

ED3	Description (added 5/7/1996): A one-word conditional branch instruction at LA-1 is not allowed. **Pertains to:** DSP56300 Family Manual, Appendix B, Section B.4.1.3	0J17D and 1F90S
ED4	Description (added 11/11/1996): The following instructions should not start at address LA: MOVE to/from Program space {MOVEM, MOVEP (only the P space options). **Pertains to:** DSP56300 Family Manual, Appendix B	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED7	Description (added 1/27/98): When activity passes from one DMA channel to another and the DMA interface accesses external memory (which requires one or more wait states), the DACT and DCH status bits in the DMA Status Register (DSTR) may indicate improper activity status for DMA Channel 0 (DACT = 1 and DCH[2:0] = 000). Workaround: None. **Pertains to:** DSP56300 Family Manual, Sections 8.1.6.3 and 8.1.6.4	0J17D and 1F90S
ED9	Description (added 1/27/98): When the SCI is configured in Synchronous mode, internal clock, and all the SCI pins are enabled simultaneously, an extra pulse of 1 DSP clock length is provided on the SCLK pin. Workaround: a. Enable an SCI pin other than SCLK. b. In the next instruction, enable the remaining SCI pins, including the SCLK pin. **Pertains to:** UM, SCI Chapter (Use the 302 UM as your reference, Section 8.4.2, “SCI Initialization”)	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED14	The data sheets of the various DSP56300 host interfaces (HI32 excluded) must be modified to make the HI08/HDI08 compatible with PortA timing 114, which is included here as a reference. Timing 321 "Write data strobe deassertion width" should be split (similar to timing 319 "Read data strobe deassertion width"), as described here: Write data strobe deassertion width: • after HCTR, HCVR and "Last Data Register" writes 2.5*Tc+10.0 @66MHz 2.5*Tc+8.3 @80MHz 2.5*Tc+6.6 @100MHz • after TXH:TXM writes (with HBE=0), TXM:TXL writes (with HBE=1) 25 @66MHz 20.6 @80MHz 16.5 @100MHz That is, a minimum of 4 WS for PortA is required for 100 MHz operation. Reference: Timing 114 @ 100MHz 114 WR_ deassertion time 0.5 x TC - 3.5 [WS = 1] 1.5ns TC - 3.5 [2 <= WS <= 3] 6.5ns 2.5 x TC - 3.5 [4 <= WS <= 7] 21.5ns 3.5 x TC - 3.5 [WS >= 8] 31.5ns	0J17D and 1F90S
ED15	Description (added 7/21/98): The DRAM Control Register (DCR) should not be changed while refresh is enabled. If refresh is enabled only a write operation that disables refresh is allowed. Workaround: First disable refresh by clearing the BREN bit, than change other bits in the DCR register, and finally enable refresh by setting the BREN bit.	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED17	Description (added 9/28/98): In all DSP563xx technical datasheets, a note is to be added under "AC Electrical Characteristics" that although the minimum value for "Frequency of Extal" is 0MHz, the device AC test conditions are 15MHz and rated speed. Workaround: N/A	0J17D and 1F90S
ED20	Description (added 11/24/98): In the Technical Datasheet Voh-TTL should be listed at 2.4 Volts, not as: `TTL = Vcc-0.4` Workaround: This is a documentation update.	0J17D and 1F90S
ED21	Description (added 11/24/98): In the Technical Datasheet Iol should be listed as 1.6 mA, not as 3.0 mA. Workaround: **Pertains to:** Data sheet, under “DC Electrical Characteristics” near beginning of Chapter 2. Typically, the table number is 2-3. See Output Low Voltage in the “Characteristics” column.	0J17D and 1F90S
ED24	Description (added 11/24/98): The technical datasheet supplies a maximum value for internal supply current in Normal, Wait, and Stop modes. These values will be removed because we will specify only a "Typical" current. Workaround: This is a documentation update.	0J17D and 1F90S
ED26	Description (added 1/6/99): The specification DMA Chapter is wrong. “Due to the DSP56300 Core pipeline, after DE bit in DCRx is set, the corresponding DTDx bit in DSTR will be cleared only after two instruction cycles.” Should be replaced with: “Due to the DSP56300 Core pipeline, after DE bit in DCRx is set, the corresponding DTDx bit in DSTR will be cleared only after three instruction cycles.”	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask

ED28	Description (added 1/7/1997; identified as Documentation Errata 2/1/99):
	
	When two consecutive LAs have a conditional branch instruction at LA-1 of the internal loop, the part does not operate properly. For example, the following sequence may generate incorrect results:
	
	`DO #5, LABEL1`
	`NOP`
	`DO #4, LABEL2`
	`NOP`
	`MOVE (R0) +`
	`BSCC _DEST ; conditional branch at LA-1 of internal loop`
	`NOP ; internal LA LABEL2`
	`NOP ; external LA LABEL1`
	`NOP`
	`NOP`
	`_DEST NOP`
	`NOP`
	`RTS`
	`Workaround: Put an additional NOP between LABEL2 and LABEL1.`
	
	Pertains to: DSP56300 Family Manual, Appendix B, Section B-4.1.3, "At LA-1."
		0J17D and 1F90S

ED29	Description (added 9/12/1997; identified as a Documentation errata 2/1/99): When the ESSI transmits data with the CRA Word Length Control bits (WL[2:0]) = 100, the ESSI is designed to duplicate the last bit of the 24-bit transmission eight times to fill the 32-bit shifter. Instead, after shifting the 24-bit word correctly, eight 0s are being shifted. Workaround: None at this time. **Pertains to:** UM, Section 7.4.1.7, "CRA Word Length Control." The table number is 7-2.	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED30	Description (added 9/12/1997; identified as a Documentation errata 2/1/99): When the ESSI transmits data in the On-Demand mode (that is, MOD = 1 in CRB and DC[4:0] = $00000 in CRA) with WL[2:0] = 100, the transmission does not work properly. Workaround: To ensure correct operation, do not use the On-Demand mode with the WL[2:0] = 100 32-bit Word-Length mode. **Pertains to:** UM, Section 7.5.4.1, "Normal/On-Demand Mode Selection."	0J17D and 1F90S
ED31	Description (added 9/12/1997; modified 9/15/1997; identified as a Documentation errata 2/1/99): Programming the ESSI to use an internal frame sync (that is, SCD2 = 1 in CRB) causes the SC2 and SC1 signals to be programmed as outputs. If however, the corresponding multiplexed pins are programmed by the Port Control Register (PCR) to be GPIOs, then the GPIO Port Direction Register (PRR) chooses their direction, but this causes the ESSI to use an external frame sync if GPIO is selected. **Note:** This errata and workaround apply to both ESSI0 and ESSI1. Workaround: To assure correct operation, either program the GPIO pins as outputs or configure the pins in the PCR as ESSI signals. **Note:** The default selection for these signals after reset is GPIO. **Pertains to:** UM, Section 7.4.2.4, "CRB Serial Control Direction 2 (SCD2) Bit 4"	0J17D and 1F90S

ED32	Description (added 11/9/98; identified as a Documentation errata 2/1/99):
	
	When returning from a long interrupt (by RTI instruction), and the first instruction after the RTI is a move to a DALU register (A, B, X, Y), the move may not be correct, if the 16-bit arithmetic mode bit (bit 17 of SR) is changed due to the restoring of SR after RTI.
	
	Workaround: Replace the RTI with the following sequence:
	
	`movec ssl,sr`
	`nop`
	`rti`
	
	Pertains to: DSP56300 Family Manual. Add a new section to Appendix B that is entitled "Sixteen-Bit Compatibility Mode Restrictions."
		0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask

ED33	Description (added 12/16/98; identified as a Documentation errata 2/1/99):
	
	When Stack Extension mode is enabled, a use of the instructions BRKcc or ENDDO inside do loops might cause an improper operation.
	
	If the loop is non nested and has no nested loop inside it, the erratais relevant only if LA or LC values are being used outside the loop.
	
	Workaround:
	
	If Stack Extension is used, emulate the BRKcc or ENDDO as in the following examples. We split between two cases, finite loops and do forever loops.
	
	1) Finite DO loops (that is, not DO FOREVER loops)
	
	==========
	
	BRKcc
	
	Original code:
	
	`do #N,label1`
	`.....`
	`.....`
	`do #M,label2`
	`.....`
	`.....`
	`BRKcc`
	`.....`
	`.....`
	`label2`
	`.....`
	`.....`
	`label1`
	
	Will be replaced by:
	
	`do #N, label1`
	`.....`
	`.....`
	`do #M, label2`
	`.....`
	`.....`
	`Jcc fix_brk_routine`
	`.....`
	`.....`
		0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED33 cont.	nop_before_label2 nop ; This instruction must be NOP. label2 .... label1 .... fix_brk_routine move #1,lc jmp nop_before_label2 ENDDO ------ Original code: do #M,label1 .... do #N,label2 .... ENDDO .... label2 .... label1 Will be replaced by: do #M, label1 .... do #N, label2 .... JMP fix_enddo_routine	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED33 cont.	nop_after_jmp NOP ; This instruction must be NOP. .... label2 .... label1 .... fix_enddo_routine move #1,lc move #nop_after_jmp,la jmp nop_after_jmp 2) DO FOREVER loops ========== BRKcc ----- Original code: do #M,label1 .... do forever,label2 .... BRKcc .... label2 .... label1	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask

ED33 cont.	Will be replaced by:
	`do #M,label1`
	`.....`
	`.....`
	`do forever,label2`
	`.....`
	`.....`
	`JScc fix_brk_forever_routine ; <---`
	`note: JScc and not Jcc`
	`.....`
	`.....`
	`nop_before_label2`
	`nop ; This instruction must be NOP.`
	`label2`
	`.....`
	`.....`
	`label1`
	`....`
	`....`
	`fix_brk_forever_routine`
	`move ssh,x:<..> ; <..> is some reserved not used address (for temporary data)`
	`move #nop_before_label2,ssh`
	`bclr #16,ssl ;`
	`move #1,lc`
	`rti ; <---- note: "rti" and not "rts" !`
	`ENDDO`
	`------`
	`Original code:`
	`do #M,label1`
	`.....`
	`.....`
		0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED33 cont.	do forever,label2 .... ENDDO .... label2 .... label1 Will be replaced by: do #M,label1 .... do forever,label2 .... JSR fix_enddo_routine ; <--- note: JSR and not JMP nop_after_jmp NOP ; This instruction should be NOP .... label2 .... label1 .... fix_enddo_routine nop move #1,lc bclr #16,ssl move #nop_after_jmp,la rti ; <--- note: "rti" and not "rts" **Pertains to:** DSP56300 Family Manual, Section B-4.2, “General Do Restrictions.”	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED34	Description (added 1/5/99; identified as a Documentation errata 2/1/99): When stack extansion is enabled, the read result from stack may be improper if two previous executed instructions cause sequential read and write operations with SSH. Two cases are possible: Case 1: For the first executed instruction: move from SSH or bit manipulation on SSH (that is, jclr, brclr, jset, brset, btst, bsset, jsset, bsclr, jsclr). For the second executed instruction: move to SSH or bit manipulation on SSH (that is, jsr, bsr, jscc, bscc). For the third executed instruction: an SSL or SSH read from the stack result may be improper—move from SSH or SSL or bit manipulation on SSH or SSL (that is, bset, bclr, bchg, jclr, brclr, jset, brset, btst, bsset, jsset, bsclr, jsclr). Workaround: Add two NOP instructions before the third executed instruction. Case 2: For the first executed instruction: bit manipulation on SSH (that is, bset, bclr, bchg). For the second executed instruction: an SSL or SSH read from the stack result may be improper—move from SSH or SSL or bit manipulation on SSH or SSL (that is, bset, bclr, bchg, jclr, brclr, jset, brset, btst, bsset, jsset, bsclr, jsclr). Workaround: Add two NOP instructions before the second executed instruction. **Pertains to:** DSP56300 Family Manual, Appendix B, add a new section called “Stack Extension Enable Restrictions.” Cover all cases. Also, in Section 6.3.11.15, add a cross reference to this new section.	0J17D and 1F90S
ED36	Description (added 4/19/99): Vih on $\overline{BB}$ should be kept above 2.3V, not 2.0V as shown in the 56302 datasheet. Workaround: Specification Update. Pertains to: Datasheet,	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask
ED38	Description (added 7/14/99): If Port A is used for external accesses, the BAT bits in the AAR3-0 registers must be initialized to the SRAM access type (that is, BAT = 01) or to the DRAM access type (that is, BAT = 10). To ensure proper operation of Port A, this initialization must occur even for an AAR register that is not used during any Port A access. Note that at reset, the BAT bits are initialized to 00. **Pertains to:** *DSP56300 Family Manual*, Port A Chapter (Chapter 9 in Revision 2), description of the BAT[1–0] bits in the AAR3–AAR0 registers. Also pertains to the core chapter in device-specific user's manuals that include a description of the AAR3–AAR0 registers with bit definitions (usually Chapter 4).	0J17D and 1F90S

Table 6. Documentation Errata for DSP56309, Rev A (Mask 0J17D) (Continued)

Errata Number	Document Update	Applies to Mask

ED40	Description (added 11/11/99):
	When an instruction with all the following conditions follows a repeat instruction, then the last move will be corrupted.:
	1. The repeated instruction is from external memory.
	2. The repeated instruction is a DALU instruction that includes 2 DAL registers, one as a source, and one as destination (e.g. tfr, add).
	3. The repeated instruction has a double move in parallel to the DALU instruction: one move's source is the destination of the DALU instruction (causing a DALU interlock); the other move's destination is the source of the DALU instruction.
	Example:
	rep #number
	tfr x0,a x(r0)+,x0 a,y0 ; This instruction is from external memory
	|__|_____|------|----------> This is condition 3 second part.
	|_____|----------> This is condition 3, first part—DALU interlock
	In this example, the second iteration before the last, the "x(r0)+,x0" doesn't happen. On the first iteration before the last, the X0 register is fixed with the "x(r0)+,x0", but the "tfr x0,a" gets the wrong value from the previous iteration's X0. Thus, at the last iteration the A register is fixed with "tfr x0,a", but the "a,y0" transfers the wrong value from the previous iteration's A register to Y0.
	Workaround:
	1. Use the DO instruction instead; mask any necessary interrupts before the DO.
	2. Run the REP instructions from internal memory.
	3. Don't make DALU interlocks in the repeated instruction. After the repeat make the move. In the example above, all the "move a,y0" are redundant so it can be done in the next instruction:
	`rep #number`
	`tfr x0,a x(r0)+,x0`
	`move a,y0`
	If no interrupts before the move is a must, mask the interrupts before the REP. **Pertains to:** *DSP56300 Family Manual*, Rev. 2, Section A.3, "Instruction Sequence Restrictions."
		0J17D and 1F90S

12 DSP56309 Document Errata

This section lists documentation errata that are present on the DSP56309 but not on the DSP56302.

Errata Number	Description	Status
ED35	Description (added 2/23/99): There is an error in the 56309 user's manual on page 4-17, Section 4.6, OMR description. Bit 13 of the OMR is described as ABE—Asynchronous Bus arbitration Enable. This is not true in revision 0 of the DSP56309 (this bit will be added in the next revision). This bit was still reserved when revision 0 of the DSP56309 was designed.	0J17D

How to Reach Us:

Home Page:
www.freescale.com

E-mail:
support@freescale.com

USA/Europe or Locations not listed:
Freescale Semiconductor
Technical Information Center, CH370
1300 N. Alma School Road
Chandler, Arizona 85224
+1-800-521-6274 or +1-480-768-2130
support@freescale.com

Europe, Middle East, and Africa:
Freescale Halbleiter Deutschland GMBH
Technical Information Center
Schatzbogen 7
81829 München, Germany
+44 1296 380 456 (English)
+46 8 52200080 (English)
+49 89 92103 559 (German)
+33 1 69 35 48 48 (French)
support@freescale.com

Japan:
Freescale Semiconductor Japan Ltd.
Headquarters
ARCO Tower 15F
1-8-1, Shimo-Meguro, Meguro-ku,
Tokyo 153-0064, Japan
0120 191014 or +81 3 5437 9125
support.japan@freescale.com

Asia/Pacific:
Freescale Semiconductor Hong Kong Ltd.
Technical Information Center
2 Dai King Street
Tai Po Industrial Estate
Tai Po, N.T. Hong Kong
+800 2666 8080

For Literature Requests Only:
Freescale Semiconductor Literature Distribution Center
P.O. Box 5405
Denver, Colorado 80217
1-800-441-2447 or 303-675-2140
Fax: 303-675-2150
LDCForFreescaleSemiconductor@hibbertgroup.com

Document Order No.: EB345
Rev. 2
10/2005