MOTOROLA

Chip Errata
DSP56301 Digital Signal Processor
Mask: 2F48S

General remark: In order to prevent the usage of instructions or sequences of instructions that do not operate correctly, the user is encouraged to use the “lint563” program to identify such cases and use alternative sequences of instructions. This program is available as part of the Motorola DSP Tools CLAS package.

Silicon Errata

Errata Number	Errata Description	Applies to Mask
ES16	Description (added 6/26/1997): When the chip is powered up with PLL enabled (PINIT = 1), the skew between EXTAL and CLKOUT after the PLL locks cannot be guaranteed at high frequency (over 50 MHz, not 100% tested). Workaround: If skew between EXTAL and CLKOUT is needed, power up with PINIT = 0, and then enable the PLL by software.	2F48S
ES30	Description (added 6/26/1997): 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.	2F48S

Errata Number	Errata Description	Applies to Mask
ES37	Description (added 9/2/1997): In PCI mode, improper HI32 operation may result if the HTXR/HRXS registers are accessed by the PCI master at byte address Base_Address + (N × 2048 + 16), where N is an integer from 1–31. Workaround: Not available.	2F48S
ES38	Description (added 9/11/1997): 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.	2F48S
ES39	Description (added 9/11/1997): When the ESSI transmits data in the On-Demand mode (i.e., 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.	2F48S

Errata Number	Errata Description	Applies to Mask
ES40	Description (added 9/11/1997; modified 9/15/1997): Programming the ESSI to use an internal frame sync (i.e., 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 GPI 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 GPI.	2F48S
ES41	Description (added 9/15/1997): The HCLK pin of the HI32 presents an input capacitive load of almost 30 pF, which exceeds the permissible maximum load of 12 pF as specified in the PCI Specification Version 2.1. This may cause improper HI32 operation in PCI systems. **Note:** The effect of this extra load may vary from system to system, depending on PCI clock driver strength. Workaround: Use a zero-propagation-delay external PLL device (e.g., CY2305) to buffer the PCI clock signal. This solution does not enable spread-spectrum PCI clocking.	2F48S

Errata Number	Errata Description	Applies to Mask
ES45	Description (added 3/3/ 98): When the Host Command Vector Register (HCVR) is written in Peripheral Component Interconnect (PCI) mode while the Receive Buffer Lock Enable (RBLE) bit is set in the DSP PCI Control Register (DPCR), the Host Data Transfer Complete (HDTC) status bit in DSP PCI Status Register (DPSR) may be set falsely, thus also causing an HDTC interrupt if that interrupt has been enabled by the Transfer Complete Interrupt Enable (TCIE) bit in the DPCR. Workaround: Use either one of the following alternatives: • Clear HDTC, if it is set, by writing it with 1 in the Host Command Interface Status Register (ISR). • Clear HDTC, if it is set, by writing it with 1; use software-dependent information to distinguish between a false and true HDTC setting. For example, you do either of the following: - Alter the destination address pointer if the DSP Receive Data Register (DRXR) data is being transferred by the DSP core. The pointer is changed if the HDTC setting is true. - Alter the destination address or counter registers of the DMA channel if the DRXR data is being transferred by the DMA. The registers are changed if the HDTC setting is true.	2F48S

Errata Number	Errata Description	Applies to Mask
ES46	Description (added 6/26/1997): When a DMA controller is in a mode that clears $\overline{DE}$ (i.e., 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 general 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 (i.e., 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 (i.e., 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.	2F48S

Errata Number	Errata Description	Applies to Mask
ES46 cont.	(Errata #46 continued) For systems using the HI32 and DMA interface, in which the host processor stores the exact number (N) of words to receive or transmit, the following three examples indicate how the workarounds are used: Transfers from the HI32 to the DSP; DMA reads from DRXR: a. The host processor writes N words to HTXR with the DMA channel working in mode 5 (TM = 101) and programmed to receive N words (DCO initial value equals N –1) with the DMA interrupt enabled. b. After the DMA has read the N words, it enters the ISR, which disables the DMA, updates the pointers, and re-enables the DMA. **Note:**This is based on Workaround #2 above. Transfers from the DSP to the HI32; DMA writes to DTXS: a. The host processor is required to read N words with the DMA channel working in mode 5 (TM =101) and programmed to transmit N + 6 words (DCO initial value equals N + 5) with the DMA interrupt enabled. b. By the time the host processors completes reading of the N words from HRXS, the DMA has filled the FIFO and entered the DMA interrupt. The DMA ISR should disable the DMA, update the pointers, and generate a software reset to the HI32 by writing 000 to HM in the DCTR. After this is complete (i.e., HACT in DSR is cleared), the ISR can re-enable the HI32 and the DMA controller. Transfers from the DSP to the HI32; DMA writes to DTXM: a. The host processor is required to read N words with the DMA channel working in mode 5 (TM =101) and programmed to transmit N + 8 words (DCO initial value equals N +7) with the DMA interrupt enabled. b. By the time the host processors completes reading of the N words from HRXS, the DMA has filled the FIFO and entered the DMA interrupt. The DMA ISR should disable the DMA, update the pointers, and generate a software reset to the HI32 by writing 000 to HM in the DCTR. After this is complete (i.e., HACT in DSR is cleared), the ISR can re-enable the HI32 and the DMA controller. **Note:**This is the same as a DMA write to DTXS, except for the number of words for which the DMA is programmed.	

Errata Number	Errata Description	Applies to Mask
ES47	Description (added 6/26/1997): 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 (i.e., 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`	2F48S

Errata Number	Errata Description	Applies to Mask
ES48	Description (added 6/26/1997): **Note:**This is a subset of Errata # 46 (i.e., in every case that errata # 10 occurs, errata # 46 occurs, but not vice versa). When a DMA controller is in a mode that clears $\overline{DE}$ (i.e., 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 (i.e., TM = 1xx) and activate the DMA interrupt. b. In the DMA ISR, clear DE, update the address registers, and set DE. Workaround 2: a. Use a DMA mode that does not clear DE (i.e., 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`	2F48S

Errata Number	Errata Description	Applies to Mask
ES48 Cont.	(Errata #10 continued) For systems using the HI32 and DMA interface, in which the host processor stores the exact number (N) of words to receive or transmit, the following three examples indicate how the workarounds are used: Transfers from the HI32 to the DSP; DMA reads from DRXR: a. The host processor writes N words to HTXR with the DMA channel working in mode 5 (TM = 101) and programmed to receive N words (DCO initial value equals N –1) with the DMA interrupt enabled. b. After the DMA has read the N words, it enters the ISR, which disables the DMA, updates the pointers, and re-enables the DMA. **Note:**This is based on Workaround #1 above. Transfers from the DSP to the HI32; DMA writes to DTXS: a. The host processor is required to read N words with the DMA channel working in mode 5 (TM =101) and programmed to transmit N + 6 words (DCO initial value equals N + 5) with the DMA interrupt enabled. b. By the time the host processors completes reading of the N words from HRXS, the DMA has filled the FIFO and entered the DMA interrupt. The DMA ISR should disable the DMA, update the pointers, and generate a software reset to the HI32 by writing 000 to HM in the DCTR. After this is complete (i.e., HACT in DSR is cleared), the ISR can re-enable the HI32 and the DMA controller. Transfers from the DSP to the HI32; DMA writes to DTXM: a. The host processor is required to read N words with the DMA channel working in mode 5 (TM =101) and programmed to transmit N + 8 words (DCO initial value equals N +7) with the DMA interrupt enabled. b. By the time the host processors completes reading of the N words from HRXS, the DMA has filled the FIFO and entered the DMA interrupt. The DMA ISR should disable the DMA, update the pointers, and generate a software reset to the HI32 by writing 000 to HM in the DCTR. After this is complete (i.e., HACT in DSR is cleared), the ISR can re-enable the HI32 and the DMA controller. **Note:**This is the same as a DMA write to DTXS, except for the number of words for which the DMA is programmed.	

Errata Number	Errata Description	Applies to Mask
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 (i.e., 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 (i.e., Instruction Register, Boundary Scan Register, or ID Register). — Before using any other JTAG instruction, load one of the other BYPASS instruction codes (i.e., any code from 1000–1100 ($8–$C) or 1110 ($E)) into the instruction register. Then, any other JTAG instruction may be used.	2F48S

Errata Number	Errata Description	Applies to Mask
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.	2F48S
ES72	Description (added 10/30/1997): During external memory accesses, noise may be generated by the Bus Strobe ($\overline{BS}$) signal and placed on the Test Clock (TCK) signal, causing the JTAG Test Access Port (TAP) controller to change states unpredictably. This problem may be more severe at higher speeds or in applications using multiple DSP56300 families. NOTE: Applies to PBGA package only. Workaround: Bypass the $\overline{BS}$ line to signal ground with a 10 pF capacitor.	2F48S
ES81	Description (added 5/3/98): The HI32 may generate a wrong PAR signal. Workaround: If possible, the system should ignore parity errors generated in such a case.	2F48S

Errata Number	Errata Description	Applies to Mask
ES82	Description (added 5/13/98): The $\overline{BL}$ pin may operate improperly when two consecutive manipulation instructions (bset/bclr/bchg) use external memory as the destination. Example of the sequence: `bset #5,x:(r0) ;; r0 is a pointer on an external memory address` `bclr #7,x:(r3) ;; r3 is a pointer on an external memory address` Workaround : Separate the consecutive bit manipulation instructions by any other instruction, as in the following example: `bclr #7,x:(r3) ;; r3 is a pointer on an external memory address` `nop` `bset #5,x:(r0) ;; r0 is a pointer on an external memory address`	2F48S

Errata Number	Errata Description	Applies to Mask
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`	2F48S

Errata Number	Errata Description	Applies to Mask
ES86	Description (added 4/23/98): If the HI32 performs a write transaction as a PCI master and the transaction is disconnected by the target, the value of the MTRQ status bit in the DPSR register may be wrong. Workaround: Do not use an MTRQ status bit-related interrupt or polling. (The related DMA functionality is not affected by this issue.)	2F48S
ES87	Description (added 5/28/98): When the HI32 is an active PCI target, it does not set the DPE bit in the CSTR register if an address parity error occurs. Workaround : The Host can get information about the Address Parity status either by reading the SSE bit (in the CSTR) or by indirectly reading the (e.g. via Host Command) the APER bit in the DPSR register.	2F48S
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.	2F48S

Errata Number	Errata Description	Applies to Mask
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, i.e., $\overline{BG}$ 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 - i.e. not valid if packing mode is used, and workaround 4 is valid only to section 3 b of the errata - i.e., 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.	2F48S

Errata Number	Errata Description	Applies to Mask
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.	2F48S
ES101	Description (added 10/26/98): If the reset mode is expanded mode (for example, mode 0 or mode 8 on the DSP5630x), A MOVE (not a PROGRAM FETCH) from internal P memory to any destination may not work properly. Workaround: After each reset ($\overline{\text{RESET}}$) negation and before the first move from internal program memory, execute the following sequence: `BSET #M_CE,sr` `NOP` `NOP` `NOP` `BCLR #M_CE,sr`	2F48S

Errata Number	Errata Description	Applies to Mask
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, i.e. when the DACT bit in the DSTR register is cleared. **Note:** To perform this operation most efficiently, all other DMA channels should be disabled.	2F48S
ES107	Description (added 12/8/98): The HDTC status bit (relevant only if the RBLE control bit is set) may not be set properly when both of the following conditions apply: a) DSP software clears the HDTC bit while the PCI bus is parked on the HI32. b) The PCI master read transaction is initiated by the HI32 while the bus is still parked on the HI32. Workaround: Use one of the following alternatives: 1. Avoid bus parking on the HI32. 2. Enter the Personal Software Reset (HM[2:0]=0) in HDTC ISR. 3. Poll the MRRQ and SRRQ status bits before the start of each master read transaction (e.g. in MARQ ISR). Start this transaction only when both MRRQ and SRRQ are cleared. The HDTC status bit should be cleared by the DSP software as defined in the specification.	2F48S

Errata Number	Errata Description	Applies to Mask
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 (i.e., 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.	2F48S
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.`	2F48S

Errata Number	Errata Description	Applies to Mask
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)	2F48S
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.	2F48S

Errata Number	Errata Description	Applies to Mask
ES124	Description (added 9/11/99) (reclassified from documentation to silicon errata 11/11/99): When an external PCI master executes a configuration space read from the HI32 with an odd number of byte lanes enabled (for example, $\overline{BE3}$ – $\overline{BE0}$ = 1000), the DSP drives the parity signal (HPAR) with the wrong value. This is because the $\overline{BE3}$ – $\overline{BE0}$ signals are ignored (erroneously) when generating the parity value during configuration space reads. Workaround: None. **Pertains to**: The HI32 (PCI) chapter of the user's manual, in the section on PCI Mode (DCTR[HM]=$1). In Revision 2 of the *DSP56301 User's Manual*, this section is 6.5.2 on page 6-14. The information should accompany the bullet on Memory-Space and configuration transactions as a target. NOTE: Was documentation errata, ED39.	2F48S

Documentation Errata

	Document Update	Applies to Mask
ED1	1. Description (revised 11/9/98): XY memory data move does not work properly if 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 OR 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` This is not a bug, but a documentation update. 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.	2F48S
ED2	Description (added 6/26/1997): $\overline{BL}$ pin timings T198 and T199 in the Data Sheet are changed, improving the arbitration latency: T198 is 5 ns (max), T199 is 0 ns (min). This is not a bug, but a documentation update.	2F48S
ED3	Description (added 6/26/1997): A one word conditional branch instruction at LA-1 is not allowed. This is not a bug, but a documentation update.	2F48S

ED4	Description (added 6/26/1997): The following instructions should not start at address LA: MOVE to/from Program space {MOVEM, MOVEP (only the P space options)} This is not a bug but a documentation update (Appendix B, DSP56300 Family Manual).	2F48S
ED6	Description (added 4/9/98) When the $\overline{\text{HIRQ}}$ pin is used in pulse mode (HIRH=0 in DCTR), the LT[7:0] value (in CLAT) should not be zero. This is not a bug but a documentation update.	2F48S
ED7	Description (added 1/27/98): When activity is passed 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. This is not a bug, but a specification update.	2F48S
ED8	Description (added 7/7/1997): The timing for HSAK is no longer qualified by the data strobe. The new timing numbers are: a. **T318**—HSAK assertion from HA0–HA10 and HAEN valid is 30.0 ns maximum. b. **T319**—HSAK assertion hold from HA0-HA10 and NAEN not valid is 2.0 ns minimum. This is not a bug, but a documentation update of a specification change.	2F48S

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. This is not a bug, but a specification update.	2F48S
ED10	Description (added 5/13/98): The HI32 may operate improperly in PCI mode when the TWSD bit is set in the HCTR register. Workaround: Do not set the TWSD bit in the HCTR register; this bit is reserved. This is a documentation change.	2F48S
ED12	Description (added 5/13/98): When the HI32 is in PCI mode, the HTF control bits affect the address insertion (the IAE bit is set in the DPCR register) in the same way they affect the transferred data. Address as appears on the PCI bus: $12345678 **HTF[1:0]** — **Inserted Address** 00 — $005678, $001234 01 — $345678 10 — $345678 11 — $123456 Workaround: This is a documentation update.	2F48S
ED13	Description (added 5/15/98): When the HI32 is in PCI mode, the Insert Address Enable control bit (IAE=1) can be set only with the Receive Buffer Lock Enable control bit set (RBLE=1 in the DPCR register.)	2F48S

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 - 4.0 [WS = 1] 1.5ns TC - 2.0 [2 ≤ WS ≤ 3] 6ns 2.5 x TC - 4.0 [4 ≤ WS ≤ 7] 21ns 3.5 x TC - 4.0 [WS Š 8] 31ns **Pertains to**: Data Sheets; timing 114 is in the table on SRAM Read and Write Accesses in the Port A section entitled “External Memory Expansion Port (Port A). The table number is 2-8. Timings 321 and 319 are in the section on Host Interface Timing, Table 2-20.	2F48S

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.	2F48S
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	2F48S
ED18	Description (added 11/2/98): The PCI host must not change the values of the HBE[3:0] bits during PCI read transactions from the HI32 as a PCI target.	2F48S
ED19	Description (added 11/9/98): To guarantee the proper HI32 operation, the DMA should service the HI32 under the following restrictions: • Two DMA channels should not service the DRXR FIFO if master and slave data is mixed there. • The DMA data transfers should not be concurrent with the 56300 Core data transfers to/from the same HI32 data FIFO.	2F48S
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.	2F48S

ED21	Description (added 11/24/98): In the Technical Datasheet Iol should be listed as 1.6 mA, not as 3.0 mA. Workaround: This is a documentation update.	2F48S
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.	2F48S

ED25	Description (added 12/16/98): Current definition: HDTC is set if SRRQ and MRRQ are cleared (i.e. the host-to-DSP data path is emptied by DSP56300 core reads) under one of the following conditions: • a non-exclusive PCI write transaction to the HTXR terminates or completes • HLOCK is negated after the completion of an exclusive write access to the HTXR • the HI32 initiates a read transaction. The HI32 disconnects (retry or disconnect-C) forthcoming write accesses to the HTXR as long as HDTC is set. New definition: HDTC is set if SRRQ and MRRQ are cleared (i.e. the host-to-DSP data path is emptied by DSP56300 Core reads) under one of the following conditions: • a non-exclusive PCI write transaction to the HTXR terminates or completes • HLOCK is negated after the completion of an exclusive write access to the HTXR. The HI32 disconnects (retry or disconnect-C) forthcoming write accesses to the HTXR as long as HDTC is set. Note: The HDTC bit is not set after a read transaction initiated by the HI32 as a PCI master. Workaround: NTR	2F48S
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.”	2F48S

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 (i.e., 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"	2F48S
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."	2F48S

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 (i.e. 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` `.....` `.....`	2F48S

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	2F48S

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	2F48S

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	2F48S

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.”	2F48S

ED37	Description (added 4/19/99): In paragraph 6.1.1.11 on page 6-12 of the 301 User's Manual, there is an error, as follows: "HIRQ_ is asserted by the HI32 when a host interrupt request (recieve and/or transmit) is generated in the HI32" Workaround/correction: Should be: "HIRQ_ is asserted by the HI32 when a host interrupt request (receive and/or transmit) is generated in the HI32 (as described in paragraphs 6.2.1.1, 6.2.1.1 and 6.2.1.4)."	2F48S
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 (i.e. BAT = 01) or to the DRAM access type (i.e. 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).	2F48S

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."	2F48S

ED42	Description (added on 3/22/2000) The DMA End-of-Block-Transfer interrupt cannot be used if DMA is operating in the mode in which DE is not cleared at the end of the block transfer (DTM = 100 or 101). **Pertains to:** *DSP56300 Family Manual*, Rev. 2, Section 10.4.1.2, "End-of-Block-Transfer Interrupt." Also, Section 10.5.3.5, "DMA Control Registers (DCR[5–0]," discussion of bits 21 – 19 (DTM bits).	2F48S
ED46		2F48S

NOTES

1. An over-bar (i.e., $\overline{xxxx}$) indicates an active-low signal.
2. The letters seen to the right of the errata tell which DSP56301 mask numbers apply.
3. The Motorola DSP website has additional documentation updates that can be accessed at the following URL:

   http://www.motorola-dsp.com/documentation/index.html

-end-