

Freescale Semiconductor, Inc.



Chip Errata DSP56321 Digital Signal Processor Mask: 0K91M

General remark: In order to prevent the use of instructions or sequences of instructions that do not operate correctly, we encourage you 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 <u>to Mask</u>
ES136	Description (added 3/25/2002)(Modified 5/9/2002):	0K91M
	There is an issue in the current DPLL/CLKGEN design if the DPLL is programmed in frequency phase lock (FPL) mode with integer MF (MFN = 0).	
	Workaround: If one needs to program the DPLL in FPL, represent MF as a dummy integer. For example:	
	MF = MFI + MFN/MFD 8 = 7 + 63/63,	
	15 = 14 + 102/102.	
	Additionally, always program the CBRMO bit with a value of 1 so that the bit rate modulator (BRM) operates as second order, decreasing the jitter.	
ES137	Description (added 5/9/2002):	0K91M
	JTAG Boundary Scan error: The EXTEST and SAMPLE/PRELOAD commands do not provide correct data. This error will be fixed in future revisions.	



Documentation Errata

	Description (revised 11/9/98):	0K91M
	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):	
ED1	Example 1:	
EDI	<pre>move #\$12,y0 move x0,x:(r7) y0,y:(r3) (while x:(r7) is a peripheral).</pre>	
	Example 2:	
	<pre>mac x1,y0,a x1,x:(r1)+ y:(r6)+,y0 move y0,y1</pre>	
	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.	
	Description (added before 2/18/1996):	0K91M
ED3	$\overline{\mathrm{BL}}$ 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).	
	Pertains to : Data Sheet, Synchronous Timings (SRAM) table, Table 2- 17.	



Chip Errata DSP56321 Digital Signal Processor Mask:0K91M

	Description (added 10/13/1997):	0K91M
	The following instructions should not start at address LA:	
ED4	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).	
	Description (added 1/27/98):	0K91M
ED7	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.	
	Description (added 1/27/98):	0K91M
	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.	
ED9	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.	
	Description (added 7/21/98):	0K91M
	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.	
ED15	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.	



Chip Errata DSP56321 Digital Signal Processor Mask:0K91M

	Description (added 9/28/98):	0K91M
ED17	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	
	Description (added 11/24/98):	0K91M
	In the Technical Datasheet Voh-TTL should be listed at 2.4 Volts, not as:	
ED20	TTL = Vcc-0.4	
	Workaround:	
	This is a documentation update.	
	Description (added 11/24/98):	0K91M
ED24	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.	
	Description (added 1/6/99):	0K91M
	The specification DMA Chapter is wrong.	
ED26	"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."	



	Description (added 1/7/1997; identified as Documentation Errata	0K91M
	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:	
ED28	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 _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."	
	Description (added $9/12/1997$; identified as a Documentation errata $2/1/99$):	0K91M
ED29	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.	



DSP56321 Digital Signal Processor Mask:0K91M

	Description (added $9/12/1997$; identified as a Documentation errata $2/1/99$):	0K91M
	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.	
ED30	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."	
	Description (added $9/12/1997$; modified $9/15/1997$; identified as a Documentation errata $2/1/99$):	0K91M
	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.	
ED31	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"	



DSP56321 Digital Signal Processor Mask:0K91M

-		
	Description (added $11/9/98$; identified as a Documentation errata $2/1/99$):	0K91M
	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.	
ED32	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."	



Description (added 12/16/98; identified as a Documentation errata 0K91M 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 **ED33** do #M,label2 BRKcc label2 label1 Will be replaced by: do #N, label1 do #M, label2 fix brk routine Jcc



DSP56321 Digital Signal Processor

Mask:0K91M

0K91M 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 ED33 cont. do #N,label2 ENDDO label2 label1 Will be replaced by: do #M, label1 do #N, label2 JMP fix_enddo_routine



Mask:0K91M

0K91M 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 _____ ED33 cont. BRKcc _ _ _ _ _ Original code: do #M,label1 do forever, label2 BRKcc label2 label1



Mask:0K91M

Will be replaced by: 0K91M do #M,label1 do forever, label2 fix_brk_forever_routine ; <---</pre> JScc note: JScc and not Jcc nop before label2 nop ; This instruction must be NOP. label2 ED33 cont. label1 fix brk forever routine move $\mathtt{ssh}, \mathtt{x:<} \ldots \mathtt{>}$; <...> is some reserved not used address (for temporary data) move #nop_before_label2,ssh bclr #16,ssl ; move #1,lc ; <---- note: "rti" and not "rts" ! rti ENDDO _ _ _ _ _ _ Original code: do #M,label1



DSP56321 Digital Signal Processor

Mask:0K91M

0K91M do forever, label2 ENDDO label2 label1 Will be replaced by: do #M,label1 do forever, label2 fix_enddo_routine ; <--- note:</pre> JSR ED33 cont. JSR and not JMP nop_after_jmp NOP ; This instruction should be NOP label2 label1 fix enddo routine nop move #1,1c bclr #16,ssl move #nop after jmp,la ; <--- note: "rti" and not "rts" rti Pertains to: DSP56300 Family Manual, Section B-4.2, "General Do Restrictions."



Description (added 1/5/99; identified as a Documentation errata 0K91M 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 (i.e. jclr, brclr, jset, brset, btst, bsset, jsset, bsclr, jsclr). For the second executed instruction: move to SSH or bit manipulation on SSH (i.e. 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 (i.e., bset, bclr, bchg, jclr, brclr, jset, brset, btst, bsset, jsset, bsclr, jsclr). **ED34** Workaround: Add two NOP instructions before the third executed instruction. Case 2: For the first executed instruction: bit manipulation on SSH (i.e. 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 (i.e., 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.



	Description (added 7/14/99):	0K91M
ED38	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: <i>DSP56300 Family Manual</i> , 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).	



0K91M 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 " $x(r_0)+x_0$ " doesn't happen. On the first iteration before the last, the X0 register **ED40** is fixed with the " $x(r_0)$ +, x_0 ", but the "tfr x_0 ,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) + , x0move 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."



1		1
ED42	Description (added on 3/22/2000)	0K91M
	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 = $100 \text{ or } 101$).	
	Pertains to:	
	<i>DSP56300 Family Manual</i> , 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).	
	Description: (added 1/19/2002):	0K91M
ED47	When DMA line-by-line block transfers are used with the EFCOP to perform IIR filtering with two or fewer IIR coefficients, the first output of the IIR filter is lost. The rest of the outputs are shifted and inaccurate.	
	Workaround:	
	Instead of DMA line-by-line block transfers, use DMA word-by- word block transfers.	
	Description (added 9/10/1996 as ES29; reclassified as a documentation erratum on 8/2/2002):	0K91M
ED50	When the SCI transmitter is used in Synchronous mode, the last bit of the transmitted byte might be truncated to the half of the serial cycle.	
	Workaround: Not available.	

Motorola and in are registered trademarks of Motorola, Inc. OnCE is a trademark of Motorola, Inc.

NOTES

- 1. An over-bar (i.e., \overline{xxxx}) indicates an active-low signal.
- 2. The letters in the right column tell which DSP56321 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/

-end-

For More Information On⁶This Product, Go to: www.freescale.com