MC68332 DEVICE INFORMATION
(Issue 6 - 19 April, 1994)
Rev. P Silicon
All D87M and E80J Mask Sets

The following information and errata pertain to Revision P samples of the 68332 microcontroller. This revision contains the following module versions: CPU32V11, SIMV11, TPUV06, QSMV08, TPUSRAMV04. The phrase "This is an erratum" following each item below identifies characteristics of the current silicon which are expected to be improved . "This is information only" refers to enhancements, clarifications, and changes to the documented descriptions of the microcontroller and the modules within.

GENERAL INFORMATION:

1. On 20.97 MHz versions of this device, the following errata applies: IM:79 - If a Chip Select is synchronized with the E-clock by setting the MODE bit to a "1" in the Chip Select Option Register, then the MCU may stop normal operation. This is an erratum

WORKAROUND: Do not synchronize Chip Select operation with E-clock. CSOR[10:0] "MODE" bit must be "O".

MODULE DETAILS:

68332:05

Mod Date: 3/17/94

INTEGRATION: After power-up (Vdd Vdd min.) of the MCU, input/output and output-only port pins on modules other than the integration module may be in an indeterminate state for up to 15 ms (depends on ramp up conditions). Input/output pins on these modules may be in output mode (instead of high impedance) for a short time, which may create a conflict with external drive logic. This is information only.

68332:05

WORKAROUND: If a known state is required on these pins, before the 15 ms port initialization period, external reset control logic must condition these lines.

CPU32:46

Mod Date: 3/22/94

CPU32: An incorrectly timed prefetch bus cycle may occur when there is a change of flow type of instruction (e.g. BRA, JSR) in progress when an interrupt request is recognized. With external DSACK logic, the effect is to make the minimum bus cycle time 5 clocks instead of the normal 3 clocks. With chip selects enabled, the effect is that the bus cycle is stretched by two clocks, regardless of the number of wait states selected. The incorrect bus cycle will not usually happen with zero wait states, however, a bus request (BR asserted) can cause the problem for any number of wait states. If external timing circuits expect the prefetch cycle to end when it should, i.e. a specific number of clocks after the assertion of DSACK, there may be a conflict. The extended cycle will be either immediately before or immediately after the IACK cycle. Some customers have reported that their logic appears to interpret the extra long bus cycle as a normal bus cycle followed by the start of an abnormally short bus cycle. This item usually causes harm when the external logic is "counting clocks" after a DSACK. This item is fixed on CPU32V12. This is an erratum.

CPU32:46

Workaround: External logic should wait for the rising edge of AS or DS to terminate the cycle. If external logic asserts DSACK and it is timed for 2 or more wait states, operation should occur correctly, since DSACK will assert at the proper time for the normal cycle termination point. Products using chips selects alone should not experience a problem, although the bus cycle will be stretched by two clocks.

CPU32:51

Mod Date: 4/8/94

CPU32: If the internal bus is arbitrated away just before the last cycle of a MOVEM to memory, and the MOVEM is followed by other instructions which write to external memory (ex: MOVE Reg,-(An) instruction), then the data written by the last cycle of the MOVEM will be the same data as for the subsequent instruction (ex:MOVE). In other words, the next instruction (MOVE) started execution too early. Internal bus arbitration can occur in the following situations:

1. An external bus request occurs with the SHEN[1:0] = %11 or
2. If an on-chip alternate master (ex: DMA) module takes mastership of the IMB or
3. An external bus request occurs when Factory Test (Slave) mode is enabled.

This is an erratum.

CPU32:51

Workaround:

1. If external bus requests may occur, ensure SHEN[1:0] are anything except %11.
2. If an alternate master (DMA) is present on-chip, and operating, arrange instruction sequences to prevent the occurrence of MOVEM followed by another instruction which writes to external memory (ex:MOVE Reg,-(An)). It is recommended to place a NOP after the MOVEM.
3. Factory Test (Slave) mode should not be used.

CPU32:45

Mod Date: 3/23/94

CPU32: If an IACK cycle (with AVEC asserted) is terminated by a normal BERR, a Spurious Interrupt exception will be taken as expected. If the BERR of the IACK cycle is a LATE BUS ERROR (BERR asserted after AVEC), then a Type 4 bus error exception (Faults during exception processing) will be taken, in this case the Status Register in the stack frame will be corrupted and there can be no recovery from the exception. This is an erratum.

CPU32:45

Workaround: Do not allow LATE BERR during IACK cycles that are terminated with AVEC.

CPU32:53

Mod Date: 3/14/94

CPU32: When there is a bus error on the second word of a released longword write cycle immediately before a TAS instruction, then the RR and RM bits in the special status word (SSW) are incorrect. The RR and RM bits represent the TAS cycle instead of the released write cycle. The bus cycle which had the error will not be re-run on the return from exception. Most applications do not experience this problem since software is not usually designed to to interpret and recover from bus errors. The common reaction to a bus error is to abort the program flow, report the error occurrence, and then attempt to restart the application. Applications designed that way need not be concerned with this problem. This is an erratum.

CPU32:53

Workaround: Sometimes it is not possible to determine this fault condition from the stack and program contents. When the released write happens to be a byte write to supervisor space, the SSW looks exactly like a bus error on the TAS write cycle, and if the released write address is the same as the TAS address, the fault address is not different between the two cases. The bus error exception handling software may be able to resolve the situation in most cases by:
(a) looking at the size, R/W, and function code information, or when that fails;
(b) examining the faulted address and determining whether it is the same as the address that would be generated by the TAS instruction at the return program counter. When it is the case of a released write bus error, then setting RR and clearing RM should be the correct action (with RTE), after resolving the original cause of the bus error.

CPU32:56

Mod Date: 3/14/94

CPU32: When a jump or branch to an odd address occurs and there is a pending bus error, the address error exception is incorrectly taken and the bus error exception is not taken. This is an erratum.

CPU32:56

Workaround: Do not jump or branch to odd addresses when bus errors can occur.

IM:85

Mod Date: 3/23/94

INTEGRATION: At power-up, chip select pins may drive low (asserted) until the first CLKOUT edge occurs. False writes may result on power up if the R/W pin does not have a pull-up resistor. The R/W line is in a high impedance state at power up and while reset is asserted. This is an erratum.

IM:85

WORKAROUND: Insure external writes cannot occur on power up by pulling up the R/W pin.

IM:82

Mod Date: 4/6/94

INTEGRATION: If PIT is used in either PLL mode or external clock mode with the PIT prescalar enabled (PTP bit in PITR register set), the PIT clock is 32 KHz (crystal or EXTAL/512 if in external clock mode). If the PITM field of the PITR register is written to zero, followed immediately by a nonzero value, and then LPSTOP is entered before 1 full period of the 32 KHz PIT clock source, the PIT timer will never decrement which may prevent exiting LPSTOP using the PIT timeout. This is an erratum.

IM:82

WORKAROUND: If the user wishes to stop the PIT and restart with a new time value by writing zero followed by the new value to the PITM field, the user should delay entering LPSTOP by at least one 32kHz PIT clock period after writing the new value. Alternatively, the user can update the PITM field with the new value without first stopping the counter (writing zero to the PITM field). In this case, the timer will continue counting down to the original time value before updating to the new value.

IM:77

Mod Date: 3/28/94

INTEGRATION: In some cases, a loss of clock reference to the PLL is not detected and the MCU system clock stops running. When the clock reference restarts, the system clock restarts and the MCU, begins running from where the system clock stopped. (Note: Since the RESET pin is gated by the internal system clock, the reset is not accepted by the MCU, until the system clock recovers. The RESET pin is gated by the clock since many applications require write cycles to complete prior to reset being accepted by the MCU.) This is an erratum.

IM:77

WORKAROUND: Do not rely upon the loss of clock feature.

IM:90

Mod Date: 3/28/94

INTEGRATION: The current documentation for the Phase Lock Loop (PLL) shows a single 0.1 uF capacitor from the XFC pin to the Vddsyn supply pin. This filter is part of the overall loop filter for the PLL. An improvement to the filter will enhance PLL stability under some noisy system conditions. Noise can be induced by external sources, which can be filtered by proper supply layout techniques and low pass filtering. Additionally, internal supply current transients may result from rapidly changing bus activity. Possible sources of repetitive internal noise are:
A) arbitrating the bus away,
B) entering and exiting STOP on the CPU32 or WAI on CPU16, or LPSTOP if the VCO is active (STSIM = 1). This internal noise may be sufficient to affect PLL stability, if the improved filter is not present. This improved filter replaces the single 0.1 uF capacitor between VDDsyn and XFC with a combination of components consisting of an 18 K Ohm resistor in series with a 0.1 uF capacitor. This series combination is then connected in parallel with a 0.01 uF capacitor and placed between VDDsyn and XFC. Current systems that are operating correctly may not require this filter change. It is recommended that this filter be used in new designs. The use of reference frequencies very different from those stated may require different filter values to optimize PLL stability. There are versions of the integration module that are configured for either a 32.768 KHz or a 4.194 MHz crystal source. These filter component values are based on using these standard frequencies. If the PLL is not enabled (MODCK = 0 at RESET) then the XFC filter is not required and the pin may be left unconnected but VDDsyn must be connected to VDD. This is information only

IM:90

IM:86

Mod Date: 3/23/94

INTEGRATION: At power-up, integration module I/O pins should initialize to the high impedance state. The following pins may, however, drive as outputs until the first CLKOUT edge occurs to initialize the internal logic into the high impedance state. Port E, D[15:0] and HALT (open Drain). This is an erratum.

IM:86

WORKAROUND: Insure external writes cannot occur on power up by pulling up the R/W pin.

IM:88

Mod Date: 3/11/94

INTEGRATION: If E-Clock synchronized chip selects (MODE=1) are used in combination with peripherals which can retry bus cycles an addressing problem may occur. Chip selects can be set up for synchronous E-clock support, configured for an 8 bit port and used with a peripheral connected to the upper data bus (D15:8 and BYTE=upper) . A problem occurs during a word access of the 8 bit peripheral, in combination with a retry (terminate cycle with BERR and HALT) being requested by the device on the second cycle of the access. Under this condition, the second cycle will be retried, but [A0] will be incorrect. This is information only.

IM:88

WORKAROUND: Do not attempt retry of chip selects supporting synchronous E-clock cycles on 8 bit ports.

IM:92

Mod Date: 3/31/94

Integration: Chip Selects may assert while the external bus is granted away if show cycles and external arbitration is enabled (SHEN bits of MCR set to %10). This behavior will only occur when BR is asserted during an IACK cycle terminated by AVEC (either external or with a chip select). The sequence of events are as follows:

1. External IRQ is asserted.
2. IACK cycle is run by the MCU.
3. BR is asserted on AS of IACK cycle. External AVEC is tied low to terminate IACK cycle or chip select configured for AVEC support.
4. BG is driven by the MCU.
5. BGACK is asserted in response to BG, and BR is negated.
6. BGACK is held asserted. During this time the MCU asserts a Chip Select (in response to address $1FFE on IMB, first stack cycle). Data bus drives unknown data.
7. After BGACK is negated, the $1FF8 stack cycle is run. The stacking is corrupted.

This is an erratum.

IM:92

WORKAROUND:

1. Assert BR one clock cycle after the assertion of AS on the IACK cycle instead of coincident with AS.

or;

2. Set the SHEN bits in the MCR to 11 instead of 10. This prevents internal bus cycles on the IMB while the external bus is granted away.

SRAM:10

Mod Date: 4/6/94

SRAM: When the VSTBY pin has power applied, and VDD is being powered up or down (supply transient conditions), the standby current (ISB) may temporarily draw current in excess of normal operating (VDD=VSTBY=5.0 V) or standby (VDD=0 V) limits. Once VDD is below 0.5 Volts or above (VSTBY - 0.5 Volts), the current returns to specified levels. The slower VDD transitions, the longer the transient current is present. Please refer to the RAM Standby Current transient specification in the users manual for this device. This is information only.

SRAM:10

Archived: 4/19/94

Revision 0.0, 20 NOV 96

last update: Mar 26 1998