

TRANSPORTATION SYSTEMS GROUP
MASK SET ERRATA AND INFORMATION SHEET
Part: HC812AV4.01H73K.A Mask Set:
Report Generated: Aug 06, 99 02:00

HC812AV4.01H73K.A Modules

Current Module Revision

ATD8B8C.2.7
BDM256.2.2
CDL_AV4.1.0
CORNER.2.3
CPU.3.3
EBI_AV4.1.1
EE4K.2.1
INT.1.3
IOB.2.2
KWU24C.1.2
LIM_AV4.2.0
MMI_AV4.1.0
MSI2C1P.2.0
MXS6X7S.1.2
PADS_AV4.2.0
PDVSB.1.0
RAM1K.1.6
ROC.2.0
TIM16B8C.1.3
VDD.3.1
VDDA.3.1
VDDX.3.1
VPP.3.0
VREF.1.0
VSS.3.1
VSSA.3.1
VSSX.2.1
WCR.1.1
XFC.3.1
XTAL.2.2

HC12_AR_380

Customer Erratum

HC812AV4.01H73K.A

DESCRIPTION:

The present STOP Idd test limit is set to a higher limit on the H73K mask sets which is greater than the expected specification limit of: 10uA -40C to 85C 25uA -40C to 105C 50uA -40C to 125C

WORKAROUND:

Designs should account for the possible higher than expected STOP Idd.

HC12_AR_374

Customer Erratum

ATD8B8C.2.7

DESCRIPTION:

ATD status bits and conversion counter are not reset properly when writing any ATD register while an active A/D conversion sequence is in the process of completion.

WORKAROUND:

When starting a new sequence, perform two writes to control registers 4/5 in quick succession. If the first write occurs when the status bit/conversion counter is not reset, the second write will correct ATD operation.

HC12_AR_311

Customer Information

ATD8B8C.2.7

DESCRIPTION:

The $(VRH-VRL)/2$ internal reference conversion result may be \$7F, \$80 or \$81.

WORKAROUND:

If the $(VRH-VRL)/2$ internal reference is used (perhaps for system diagnostics), expected pass result may be \$7F, \$80 or \$81.

HC12_AR_184

Customer Information

BDM256.2.2

DESCRIPTION:

There is a small chance that the BDMACT bit in the BDM STATUS register could be unintentionally changed from 1 to 0 by a WRITE_BD_BYTE serial command. This would cause a system runaway because it would disable the BDM firmware ROM while the CPU was executing BDM firmware.

WORKAROUND:

This workaround avoids most of the cases that could cause an unintended change of BDMACT from 1 to 0. To change the ENFIRM bit in STATUS, always perform a READ_BD_BYTE serial command, modify this value with OR #\$80 to set ENFIRM or AND #\$7F to clear ENFIRM, then use WRITE_BD_BYTE to write the modified result back to STATUS. This avoids most cases that could result in an unintended clearing of BDMACT. There is one case that could still occur (though it would be very rare). A BGND instruction (opcode \$00) could be encountered between the READ_BD_BYTE and the WRITE_BD_BYTE. Even if ENBDM is zero, the BDM ROM is enabled and the CPU executes a short sequence in the BDM firmware to check the state of ENFIRM - if it was zero, the firmware immediately executes an exit sequence to return to user code (user code sees this as a long NOP).

HC12_AR_287

Customer Erratum

CPU.3.3

DESCRIPTION:

The ETBL instruction will provide an incorrect result under EITHER of the following 2 conditions:1) Y2 less than Y1 and C-bit = 0 before ETBL instruction executed2) Y2 greater than Y1 and C-bit = 1 before ETBL instruction executed except for the cases where $(B * (Y2 - Y1)) = 0$ (i.e. B=0 or Y2=Y1)

WORKAROUND:

If a borrow is needed from the equation, then set C-bit=1 before executing the ETBL instruction; if a borrow is not needed than clear C-bit=0 before executing the ETBL instruction. Example: LDxy (idx); initialize index register to point to the start point LDD (Y2); get value of Y2 into ACCDCPD (Y1); D-M, Y2-Y1, This updates C-bit LDAB (B); accumulator B initialized with ratio ETBL (idx); Perform instruction

HC12_AR_288

Customer Erratum

CPU.3.3

DESCRIPTION:

If the REV or REVW instructions are interrupted while processing a rules list, the results from the instructions may be incorrect. The Condition Code Register and Index Register Y (weight pointer for REVW) may be incorrect when the stacking occurs for the interrupt. The REV and REVW instructions produce the wrong result after returning from the interrupt because the V-bit in the CC register and IY registers (RE VW) may not hold the same state as prior to the interrupt.

WORKAROUND:

Disable the interrupts prior to using the REV or REVW instruction (which could increase the interrupt latency). If a non-maskable interrupt has been enabled, there is no workaround.

HC12_AR_217

Customer Information

EBI_AV4.1.1

DESCRIPTION:

The multiplexed IPIPE(1:0) signals, which are available on PORTE(6:5) pins, do not convey the correct information for reconstruction of the internal instruction queue during a BDMFRZ. BDMFRZ occurs when a BDM access is requested and either no free cycles are found within 128 cycles or the requested access cannot be completed in one cycle (i.e.: mis-aligned word access).

WORKAROUND:

This is very minor. When using a logic analyzer, be aware that a BDMFRZ operation will not be properly identified by the pipe status bits even though the operation will function properly.

HC12_AR_253**Customer Information****EBI_AV4.1.1****DESCRIPTION:**

When the part is in the delay phase of coming out of STOP, the E clock will be running external. Since the oscillator is just starting up, the E clock is likely to be “dirty” during the beginning of this time.

WORKAROUND:

None needed.

HC12_AR_508**Customer Information****EBI_AV4.1.1****DESCRIPTION:**

The multiplexed IPIPE(1:0) signals, which are available on PORTE(6:5) pins, do not convey the correct information for reconstruction of the internal instruction queue during a BDMFRZ. BDMFRZ occurs when a BDM access is requested and either no free cycles are found within 128 cycles or the requested access cannot be completed in one cycle (i.e.: mis-aligned word access).

WORKAROUND:

This is very minor. When using a logic analyzer, be aware that a BDMFRZ operation will not be properly identified by the pipe status bits even though the operation will function properly.

HC12_AR_193**Customer Information****EE4K.2.1****DESCRIPTION:**

Read from array after latches are written could re-open the address latches.

WORKAROUND:

Customer software should not read the array once eelat has been set and the write to the array has been performed. This is not part of the normal programming sequence anyway.

HC12_AR_235

Customer Erratum

INT.1.3

DESCRIPTION:

If IRQEN bit is cleared, the IRQ pin is still able to wake up the part from stop mode. Once the part is out of STOP, it continues with the instruction following the STOP (IRQEN = 0 prevents the IRQ interrupt from being taken).

WORKAROUND:

Make certain IRQEN is 1 before going into STOP or make certain the level on the IRQ pin is always 1 during STOP. Third choice: set the I mask before going into STOP, but this only leaves XIRQ and reset to get out of STOP.

HC12_AR_528

Customer Erratum

INT.1.3

DESCRIPTION:

When using an edge sensitive IRQ signal to trigger an interrupt service routine and the IRQ is not released during servicing, the part will not enter stop mode if the following executed instruction is STOP.

WORKAROUND:

When IRQ is set to be edge sensitive, release pin before executing STOP instruction.

HC12_AR_441

Customer Erratum

INT.1.3

DESCRIPTION:

There is a cycle at the beginning of executing a BDM instruction that is susceptible to being interrupted directly after the background has executed. Since the interrupt source is masked, the interrupt vector that is requested is \$FFF6. The BDM firmware at \$FFF6 points to the routine at \$FF24. The interrupt was stacked when it was taken, but the BDM routines do not un-stack (no RTI). When you return from BDM the stack pointer is pointing to the wrong place. Avoid using TRACE during cycles that allow interrupts to become unmasked (i.e. TRACE of a CLI if interrupts are pending). Caution should a

WORKAROUND:

None.

HC12_AR_527

Customer Information

INT.1.3

DESCRIPTION:

If the source of an interrupt is taken away by disabling the interrupt without setting the I mask bit in the CCR, an SWI interrupt may be fetched instead of the vector for the interrupt source that was disabled.

WORKAROUND:

Before disabling an interrupt using a local interrupt control bit, set the I mask bit in the CCR.

HC12_AR_510

Customer Erratum

IOB.2.2

DESCRIPTION:

Expanded mode operation causes an increase in bus driver shoot through current pin leakage which can cause inaccurate A/D conversions.

WORKAROUND:

Enabling reduced drive on ports A, B, and E while the HC12 is in expanded mode reduces the amount of VDD/VSS noise caused by bus driver shoot through current. Therefore, the A/D accuracy meets specified limits.

HC12_AR_289

Customer Information

MSI2C1P.2.0

DESCRIPTION:

Not a functional problem, but operation of general purpose I/O may be confusing when used in conjunction with SPI.

WORKAROUND:

Avoid getting a mode fault: Do not set SPE and MSTR bits with active low on SS pin when DDRS[7] is cleared. Initialize DDRS[7] to be an output before enabling SPI.

HC12_AR_161

Customer Erratum

ROC.2.0

DESCRIPTION:

If the power is dropped to approximately 1 Volt the internal clocks collapse. The part will not respond to the reset pin until VDD is brought back to 5 Volts.

WORKAROUND:

After VDD is stabilized at nominal VDDs (2.7-5 Volts), pulling the reset pin low will initiate the reset. Alternatively, if the watchdog is enabled the part will recover when it times out.

HC12_AR_196

Customer Erratum

ROC.2.0

DESCRIPTION:

Coming out of STOP with DLY = 0 and a free-running external clock source (i.e. oscillator pack) could cause the interrupt vector to be missed.

WORKAROUND:

Always use DLY = 1 when using STOP.

HC12_AR_472

Customer Erratum

ROC.2.0

DESCRIPTION:

When the device exits WAIT mode, it does not return to the correct location within the routine if the stack is positioned in external memory and if the stretch bits have been enabled to lengthen the clock.

WORKAROUND:

To overcome this problem, locate the stack in internal RAM resources and/or clear the stretch bits to prevent clock stretching.

HC12_AR_504

Customer Erratum

ROC.2.0

DESCRIPTION:

When a clock monitor reset occurs, the crystal may start-up with no delay period. As a result, the MCU attempts to fetch reset vectors before the crystal has fully stabilized.

WORKAROUND:

When clock monitor failure has occurred, hold the reset signal until the crystal has reached stable oscillation.

HC12_AR_305

Customer Erratum

ROC.2.0

DESCRIPTION:

When the clock source (i.e. crystal) has been stopped for greater than 300us there is high probability that an unstable clock will disrupt the internal tclocks. With a free running clock source (i.e. oscillator pack), asserting the wake-up signal (i.e. XIRQ, IRQ) just prior to the falling edge of external clock source can cause the part to not exit stop.

WORKAROUND:

Crystal: No workaround. Oscillator Pack: Synchronize wake-up signal to the rising edge of external clock input.

HC12_AR_400**Customer Information****ROC.2.0****DESCRIPTION:**

When coming out of STOP by using reset, the crystal start-up delay time-out does not occur. This means the MCU may be attempting to run before the crystal has become stable.

WORKAROUND:

When coming out of STOP with reset hold the reset signal until the crystal has reached stable oscillation.

HC12_AR_285**Customer Erratum****TIM16B8C.1.3****DESCRIPTION:**

The output compare flags cannot be cleared until the timer increments past the matching count. This will cause a problem when using a timer prescaler value larger than divide-by-8. The result is the output compare interrupt is asserted continuously until the timer increments past the matching count. The same result will occur when an external clock source is used to drive the timer through the pulse accumulator pin.

WORKAROUND:

It takes at least 11 cycles to get into an interrupt service routine. If the effective prescale value is greater than divide-by-8, then the user must account for the number of cycles it takes to increment the timer counter past the match point before clearing the associated output compare flag and exiting the interrupt service routine.

HC12_AR_257**Customer Erratum****VPP.3.0**

DESCRIPTION:

Vpp pin does not meet ESD expectations.

WORKAROUND:

Use extra care when handling parts.

HC12_AR_322**Customer Erratum****WCR.1.1****DESCRIPTION:**

When an I-type interrupt occurs at the same time as an RTI interrupt, the program address at \$FFC0 will be fetched.

WORKAROUND:

Point the \$FFC0 vector to a return from interrupt (RTI) instruction. This will get the program returned to start processing the proper interrupt as quickly as possible.

HC12_AR_187**Customer Information****WCR.1.1****DESCRIPTION:**

The COP reset rate was changed from the slowest to the fastest rate on the 68HC912B32. The 68HC812A4 was not affected.

WORKAROUND:

Not applicable to the 68HC812A4.

Last updated: Wednesday, 25-Aug-1999 12:59:08 CDT