MOTOROLA

TECHNICAL UPDATE

MC68HC05K0
MC68HC05K1
MC68HCL05K0
MC68HSC05K0

Technical Update contains updates to documented information appearing in other Motorola technical documents as well as new information not covered elsewhere.

We are confident that your Motorola product will satisfy your design needs. This Technical Update and the accompanying manuals and reference documentation are designed to be helpful, informative, and easy to use.

Should your application generate a question or a problem not covered in the current documentation, please call your local Motorola distributor or sales office. Technical experts at these locations are eager to help you make the best use of your Motorola product. As appropriate, these experts will coordinate with their counterparts in the factory to answer your questions or solve your problems. To obtain the latest document, call your local Motorola sales office.

TABLE OF CONTENTS

Modules

Parts Specific

TECHNICAL UPDATE

Modules

Timer Module

TIM15BIRTICLR

Revision History

Date	Revision	Description
5/1/95	2.00	Original release. Includes tracker HC705K1.004R2.

Real-Time Interrupt Generation

Reference Document: Not applicable

Tracker Number: HC705K1.004 Revision: 2.00

A comment has been added concerning changing memory map for other parts.

```
*****************************************************************
*
*       Filename: RTI1.ASM
*
*       Modified By: David Yoder
*                    Motorola CSIC Applications
*                    Revision: 2.00
*                    Date: May 5, 1993
*       Written By: Mark Johnson
*                   Motorola CSIC Applications
*                   Revision: 1.00
*                   Date: May 5, 1993
*
*       Assembled Under: P&E Microcomputer Systems IASM05K Version 3.02
*
*                        *******************************
*                        *       Revision History      *
*                        *******************************
*
*       Revision 1.00   5/5/93 Original Source
*
*****************************************************************************
*       Program Description:
*
```

```
*       This code was written to work with the MultiFunction Timer
*       TIM15B1RTICLR_A. It was tested on the MC68HC705K1 and has reset vectors
*       and memory map equates for this part. To use this code with other parts
*       that have the MultiFunction Timer TIM15B1RTICLR_A it may be necessary
*       to change the vector locations and the memory map org's.
*
*       This program generates a 49.2ms* square wave on pin PA0 using the real
*       time interrupt feature of the HC705K1. The real time interrupt rate is
*       set to 8.2ms and pin PA0 is toggled after every third interrupt.
*       (((8.2ms x 3) x 2) = 49.2ms period). The program was tested on the
*       HC705KICS board and was then programmed into an HC705K1 device to
*       verify proper operation.
*
*       *Note: internal clock frequency = 2Mhz
*
*******************************************************************************
*
*     Start of main routine
*
*     The K version of the P&E IASM05 assembler contains built-in reserved
*     labels for the HC705K1. If a different assembler is to used, the labels
*     referenced below would have to be defined with equate statements.
*     (ex. "RAM EQU $E0")
*
*
            org     ram         ;start of RAM ($E0)
rticnt      rmb     1           ;counter for number of RTIs
            org     rom         ;start of EPROM/ROM ($200)
*
*     Initialization routine
*
start       clra                ;clear accumulator
            sta     rticnt      ;clear counter
            sta     porta       ;set port a data register to zero
            lda     #$01        ;
            sta     ddra        ;make PA0 an output
            lda     #$10        ;
            sta     tcsr        ;enable real time interrupts
            cli                 ;enable all interrupts
*
*     Toggle PA0 after every third real time interrupt
*
wait        lda     rticnt      ;get counter value
            cmp     #3          ;had three RTIs yet?
            bne     wait        ;no, wait for next interrupt
            lda     porta       ;yes, get porta data register
            eor     #1          ;exclusive or PA0
            sta     porta       ;write new PA0 value to port
            clr     rticnt      ;clear counter
            bra     wait        ;wait for interrupt
*
*     Real time interrupt service routine
*
rtiisr      inc     rticnt      ;increment rti counter
            lda     #$14        ;clear rti flag
            sta     tcsr        ;
            rti                 ;return from interrupt
*
*     User vector setup
*
            org     vectors     ;start of user vectors ($3f8)
            fdb     rtiisr      ;rti vector
            fdb     start       ;no external interrupt vector
            fdb     start       ;no software interrupt vector
            fdb     start       ;reset vector
```

Computer Operating Properly (COP) Module

COP0COPRT2

Revision History

Date	Revision	Description
5/1/95	2.00	Includes tracker HC705K1.005R2.

COP Timeout Test

Reference Document: Not applicable

Tracker Number: HC705K1.005 Revision: 2.00

The program below tests the timeout on the COP module COP0COPRT2. The program can be used on any HC05 as long as the part has the COP0COPRT2 module. To work properly with a particular MCU, memory and reset vectors may need to be changed . The HC705K1 was used to verify operation.

```
************************************************************
*
* Program Name: 7K1_COP.ASM ( COP Test on the HC705K1 )
* Revision: 1.00
* Date: May 25, 1993
*
* Written By: Mark Glenewinkel
*             Motorola CSIC Applications
*
* Assembled Under: P&E Microcomputer Systems IASM05
*
*       *******************************
*       *       Revision History      *
*       *******************************
*
*       Rev     1.00    03/26/93        Mark Glenewinkel
*                       Initial Release
*
************************************************************
*
* Program Description:
*
*       This program is a simple routine that tests the COP
*       timeout on the HC705K1 MCU. The HC705K1 was programmed
*       with the M68HC705KICS board. The part was then tested
*       for COP resets on a protoboard. If the COP is working
*       correctly, PortA will toggle on approximately
*       1/2 sec intervals.*
************************************************************
```

```
PORTA   equ     $00
DDRA    equ     $04
MOR     equ     $17

        ORG     MOR
        DB      $01             ;enable COP

        ORG     $200

START   lda     #$FF            ;make port A all output
        sta     DDRA            ;

        com     $E0             ;complement RAM mem $E0
        lda     $E0             ;ACCA <- ($E0)
        sta     PORTA           ;port A <- (ACCA)


DONE    NOP                     ;branch into an infinite loop
        BRA     DONE            ;waiting for a COP timeout

        ORG     $03FE           ;define reset vector
        DW      START
```

CPU

HC05CPU

Revision History

Date	Revision	Description
5/3/95	1.00	Includes trackers HC05CPU.001, HC705C8.002R2, HC705C8.017, HC705C8.018R2, and HC705C8019.

Correction to SUB in Applications Guide

Reference Documents: M68HC05 Applications Guide MC68HC05AG/AD, page A-62; M68HC05 Applications Guide MC68HC05AG/AD Rev. 1, page A-62

Tracker Number: HC05CPU.001 Revision: 1.00

Replace the C bit description with:

The C bit (carry flag) in the condition code register gets set if the absolute value of the contents of memory is larger than the absolute value of the accumulator, cleared otherwise.

External Interrupt Timing

Reference Documents: MC68HC705C8/D Rev. 1, page 3-5; MC68HC05B6/D, Rev. 3, page 11-11, note 4; MC68HC705C8/D, Rev. 1, page 3-5; MC68HC05C9/D, page 13-7, note 3; MC68HC05C12/D, page 13-9, note 4; MC68HC05D9/D, Rev. 1, page 10-4, note 1; MC68HC05J3/D, page 9-6, note 3; and MC68HC05X16/D, page 12-6, note 4

Tracker Number: HC705C8.002 Revision: 2.00

This time (Tilil) is obtained by adding 19 instruction cycles to the total number of cycles needed to complete the service routine. The return to interrupt (RTI) is included in the 19 cycles.

I Bit in CCR During Stop Mode

Reference Document: M68HC05 Applications Guide, page 3-93

Tracker Number: HC705C8.017 Revision: 1.00

The stop mode flow chart shows that the I bit is set when stop mode is entered. However, this is not true. The I bit actually is cleared when stop mode is entered so that an external IRQ may release the processor from stop mode.

This error is present in the original applications guide as well as the revision.

BSET and BCLR are Read-Modify-Write Instructions

Reference Documents: MC68HC705C8/D Rev. 1, page 7-6; MC68HC05J1/D Rev. 1, page 5-7; MC68HC05J3/D, page 8-4; MC68HC705J2/D, page 4-16; HC05J3/705J3 Technical Databook - MC68HC05J3/D, page 8-6; MC68HC05K1/D, page 10-10; MC68HC705K1/D, page 11-10

Tracker Number: HC705C8.018 Revision: 2.00

In many data books, the read-modify-write instruction table located in the instruction set and addressing mode section does not list the BSET and BCLR instructions. These data books list BSET and BCLR as bit-manipulation instructions only.

While this is correct, it is not complete. These operations use a read-modify-write method to accomplish their task and, therefore, should be included in the table of read-modify-write instructions.

NOTE: These instructions do not use the same addressing modes as the other read-modify-write istructions. Only direct addressing is valid for BSET and BCLR.

Because BSET and BCLR are read-modify-write instructions, they may not be used with write-only registers. These registers will read back undefined data. Therefore, a read-modify-write operation will read undefined data, modify it as appropriate, and then write it back to the register. Because the original data is undefined, the data written back will be undefined also.

I Bit in CCR During Wait Mode

Reference Document: M68HC05 Applications Guide, page 3-93

Tracker Number: HC705C8.019 Revision: 1.00

The wait mode flow chart does not show that the I bit gets cleared upon entering wait mode. The I bit is cleared when wait is entered. An external IRQ or any of the internal interrupts (timer, SCI, SPI) can release the processor from wait mode.

This error is present in the original applications guide as well as the revision.

Low-Voltage Inhibit (LVI) Module

LVI3.5

Revision History

Date	Revision	Description
5/1/95	1.00	Original release. Includes trackers HC05K0.003 and HC05K0.006.

LVI and RESET

Reference document: MC68HC05K1/D, page 5-3

Tracker number: HC05K0.003 Revision: 1.00

When the LVI is enabled, a pullup resistor should be used on $\overline{\text{RESET}}$. This is because the LVI shorts $\overline{\text{RESET}}$ to GND when it detects a low V_{DD}. If there is no pullup to limit current, this will short V_{DD} to GND. It is possible that the chip will never come out of reset because V_{DD} is pulled down by the short. It also may pull great current and permanently damage the chip.

This also applies to the MC68HC05K1 and MC68HC705K1.

LVI Stop I_{DD}

Reference Documents: MC68HC05K1/D, pages 11-3 and 11-4; MC68HC705K1/D, pages 12-3 and 12-4

Tracker Number: HC05K0.006 Revision: 1.00

The stop mode I_{DD} specification does not include the I_{DD} contributed by the LVI circuit when it is enabled. The LVI adds approximately 20 μa to stop I_{DD} when enabled. This isnot a guaranteed specification but aids in the design process only.

Keyboard Interrupt (KBI) Module

KBIELPA4HPD

Revision History

Date	Revision	Description
5/1/95	1.00	Includes trackers HC705K1.008R2 and HC05J1A.004.

Interrupt Hardware Structure

Reference Document: MC68HC705K1/D, pages 4-2 and 4-3

Tracker Number: HC705K1.008 Revision: 2.00

The following information applies to all devices that include the keyboard interrupt (KBI) module KBIELPA4HPD.

The $\overline{IRQ}$ pin on the KBIELPA4HPD has an internal Schmitt trigger. The KBIELPA4HPD also has the ability to configure some of its port A pins as interrupt lines. These pins do not have Schmitt triggers.

Port A Interrupt Generation

Reference Document: MC68HC05J1A/D, page 4-3; MC68HC05K1/D, page4-4; MC68HC705K1/D Rev. 1, page 4-3;

Tracker Number: HC05J1A.004 Revision: 1.00

The external interrupt capability of pins PA3-PA0 is not disabled when those pins are defined as outputs, thus setting any of those pins to a logic one will generate an external interrupt. What follows is a detailed description of the interrupt process for the different mask options: With the mask option for an edge- and level-sensitive external interrupttrigger, a rising edge or a high level on a PA3-PA0 pin latches an external interrupt request. Edge- and level-sensitive triggering allows multiple external interrupt sources to be wire-ORed to any of the PA3-PA0 pins. As long as any source is holding a PA3-PA0 pin high, an external interrupt request is latched, and the CPU continues to execute the interrupt service routine.

With the mask option for an edge-sensitive only external interrupt trigger, a rising edge on a PA3-PA0 pin latches an external interrupt request. A subsequent external interrupt request on the same pin, or any of the other pins (PA3-PA0), can be latched only after the voltage level of the previous interrupt signal returns to logic zero.

Parts Specific

MC68HC05K0

Revision History

Date	Revision	Description
5/5/95	**1.00**	Original release. Includes trackers HC05K0.001, HC05K0.002, HC05K0.004, and 68HC05K0MSE1.

Three-Pin RC Oscillator

Resource Document: MC68HC05K1/D, page 1-9

Tracker Number: HC05K0.001 Revision: 1.00

On page 1-9, replace this statement: “The resistor (0-10 Mohm) connected to the OSC1 input improves accuracy by reducing the effects of clipping by the OSC1 input protection circuit during the RC charge/discharge cycle."

The correct statement is: “The OSC1 pin should be shorted to the side of resistor R2 which is connected to capacitor C (see Figure 1-8)."

Also, on page 1-9, Figure 1-8 resistor R1 should be replaced by a short. In other words, the resistance of R1 should be 0 ohms.

External Clock Oscillator

Resource Document: MC68HC05K1/D, page 1-8

Tracker number: HC05K0.002 Revision: 1.00

On page 1-8, the last sentence of Section 1.4.2.4 now reads: “The OSC2 pin may be left unconnected, or it can be connected to ground through a 10 ohm to 100 ohm resistance to reduce radio frequency interference .”

Correct the sentence to read: “The OSC2 pin should be left unconnected."

Stop Current Graphs

Reference Document: MC68HC05K1/D, page 11-6

Tracker Number: HC05K0.004 Revision: 1.00

The notes are incorrect at the bottom of Figure 11-5 Maximum Supply Current vs Clock Frequency stating the maximum stop I_{DD} = 100 µa when V_{DD} = 5 V and 50 µa when V_{DD} = 3 V.

These notes should be removed and the stop I_{DD} in Tables 11-3 and 11-4 should be used instead.

Mask Set Errata 1
68HC05K0MSE1

68HC05K0 8-Bit Microcontroller Unit

INTRODUCTION

This errata provides timer overflow and real-time interrupt information applicable to the following 68HC05K0 MCU mask set devices:

- E31A

MCU DEVICE MASK SET IDENTIFICATION

The mask set is identified by a four-character code consisting of a letter, two numerical digits, and a letter (e.g., E31A). Slight variations to the mask set identification code may result in an optional numerical digit preceding the standard four-character code (e.g., 1E31A).

MCU DEVICE DATE CODES

Device markings indicate the week of manufacture and the mask set used. The data is coded as four numerical digits where the first two digits indicate the year and the last two digits indicate the work week. The date code "9115" would indicate the 15th week of the year 1991.

MCU DEVICE PART NUMBER PREFIXES

Some MCU samples and devices are marked with an "SC" or "XC" prefix. An "SC" prefix denotes special/custom device. An "XC" prefix denotes device is tested but is not fully characterized or qualified over the full range of normal manufacturing process variations. After full characterization and qualification, devices will be marked with the "MC" prefix.

TIMER OVERFLOW OR REAL-TIME INTERRUPTS

In rare instances, clearing any of the timer control and status register (TCSR) enable bits could result in vectoring to the reset vector rather than the timer interrupt vector if the correct precautions are not followed. Do not clear any of the enable bits (i.e., TOFE, and RTIF) with bit manipulation instructions.

CLEARING TIMER OVERFLOW FLAG ENABLE (TOFE) BIT

```
SEI             SEI NOT REQUIRED IF USED WITHIN TIMER INTERRUPT ROUTINE.
BCLR            5,TCSR
CLI             DO NOT USE CLI IF THIS CODE SEGMENT IS USED WITHIN TIMER
                INTERRUPT ROUTINE.
```

MC68HC05K1

Revision History

Date	Revision	Description
5/5/95	1.00	Original release. Includes trackers HC05K1.001, HC05K1.002, HC05K1.004 and 68HC05K1MSE1.

Three-Pin RC Oscillator

Resource Document: MC68HC05K1/D, page 1-9

Tracker Number: HC05K0.001 Revision: 1.00

On page 1-9, replace this statement: “The resistor (0-10 Mohm) connected to the OSC1 input improves accuracy by reducing the effects of clipping by the OSC1 input protection circuit during the RC charge/discharge cycle."

The correct statement is: “The OSC1 pin should be shorted to the side of resistor R2 which is connected to capacitor C (see Figure 1-8)."

Also, on page 1-9, Figure 1-8 resistor R1 should be replaced by a short. In other words, the resistance of R1 should be 0 ohms.

External Clock Oscillator

Resource Document: MC68HC05K1/D, page 1-8

Tracker number: HC05K0.002 Revision: 1.00

On page 1-8, the last sentence of Section 1.4.2.4 now reads: “The OSC2 pin may be left unconnected, or it can be connected to ground through a 10 ohm to 100 ohm resistance to reduce radio frequency interference .”

Correct the sentence to read: “The OSC2 pin should be left unconnected."

Stop Current Graphs

Reference Document: MC68HC05K1/D, page 11-6

Tracker Number: HC05K0.004 Revision: 1.00

The notes are incorrect at the bottom of Figure 11-5 Maximum Supply Current vs Clock Frequency stating the maximum stop I_{DD} = 100 µa when V_{DD} = 5 V and 50 µa when V_{DD} = 3 V.

These notes should be removed and the stop I_{DD} in Tables 11-3 and 11-4 should be used instead.

Mask Set Errata 1
68HC05K1MSE1

68HC05K1 8-Bit Microcontroller Unit

INTRODUCTION

This errata provides timer overflow and real-time interrupt information applicable to the following 68HC05K1 MCU mask set devices:

- E30A

MCU DEVICE MASK SET IDENTIFICATION

The mask set is identified by a four-character code consisting of a letter, two numerical digits, and a letter (e.g., E30A). Slight variations to the mask set identification code may result in an optional numerical digit preceding the standard four-character code (e.g., 1E30A).

MCU DEVICE DATE CODES

Device markings indicate the week of manufacture and the mask set used. The data is coded as four numerical digits where the first two digits indicate the year and the last two digits indicate the work week. The date code "9115" would indicate the 15th week of the year 1991.

MCU DEVICE PART NUMBER PREFIXES

Some MCU samples and devices are marked with an "SC" or "XC" prefix. An "SC" prefix denotes special/custom device. An "XC" prefix denotes device is tested but is not fully characterized or qualified over the full range of normal manufacturing process variations. After full characterization and qualification, devices will be marked with the "MC" prefix.

TIMER OVERFLOW OR REAL-TIME INTERRUPTS

In rare instances, clearing any of the timer control and status register (TCSR) enable bits could result in vectoring to the reset vector rather than the timer interrupt vector if the correct precautions are not followed. Do not clear any of the enable bits (i.e., TOFE, and RTIF) with bit manipulation instructions.

CLEARING TIMER OVERFLOW FLAG ENABLE (TOFE) BIT

```
SEI             SEI NOT REQUIRED IF USED WITHIN TIMER INTERRUPT ROUTINE.
BCLR            5,TCSR
CLI             DO NOT USE CLI IF THIS CODE SEGMENT IS USED WITHIN TIMER
                INTERRUPT ROUTINE.
```