



Application Note 23

Title: **Programming an EM6540**

Product Family: **4 bits Microcontroller**

Part Number: EM66xx, EM65xx

Keywords: 4-bits micro-controller

Date: January 07, 2002 REV. B/416

When a MFP is programmed, the programming box pulls the pin "test" high. At the end of the programmation, this pin returns low for 1s (figure 2). After that, the pin "test" is pulled high again during the read of the CRC.

On MFP with an internal RC oscillator, the program starts when the pin "test" is 0. If the circuit goes into sleep mode during the 1s test=0 period the programming box will not be able to read the CRC value anymore. In this situation, the CRC of the MFP will read "8F59" (figure 1).

To resolve this problem, you have two solutions:

- On start up, you read an input terminal. With one value (0 or 1), you wait in an active loop. Reading the other value, you start running the program.
- On start up, you wait more than 1s before running the program (use internal counter).

When you are finished debugging your program on MFP, you should remove the additional routine before sending the program into ROM production.

When a MFP is programmed, you must remove it out of the programming box.

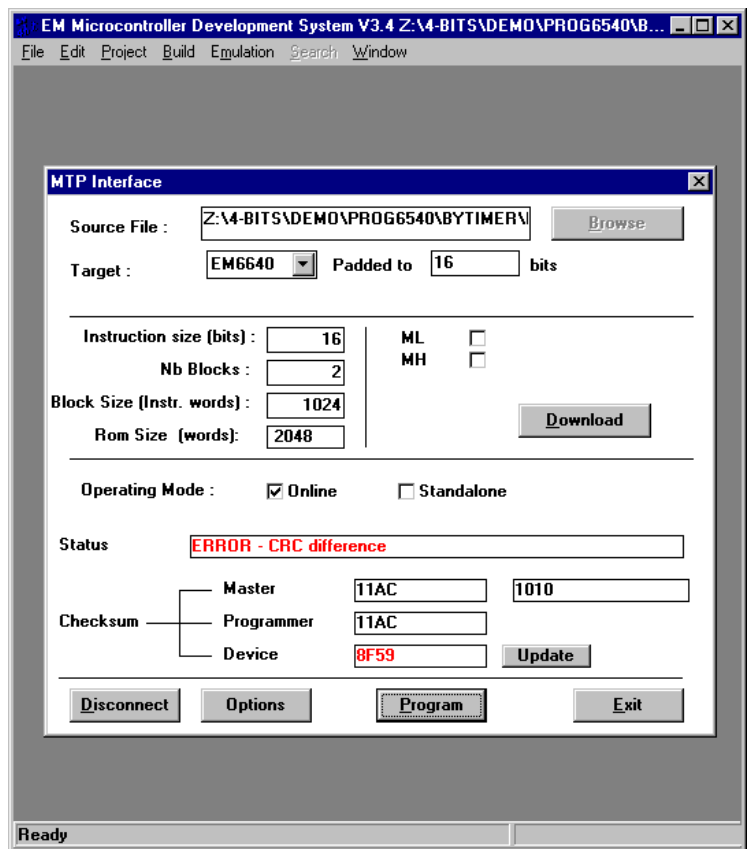


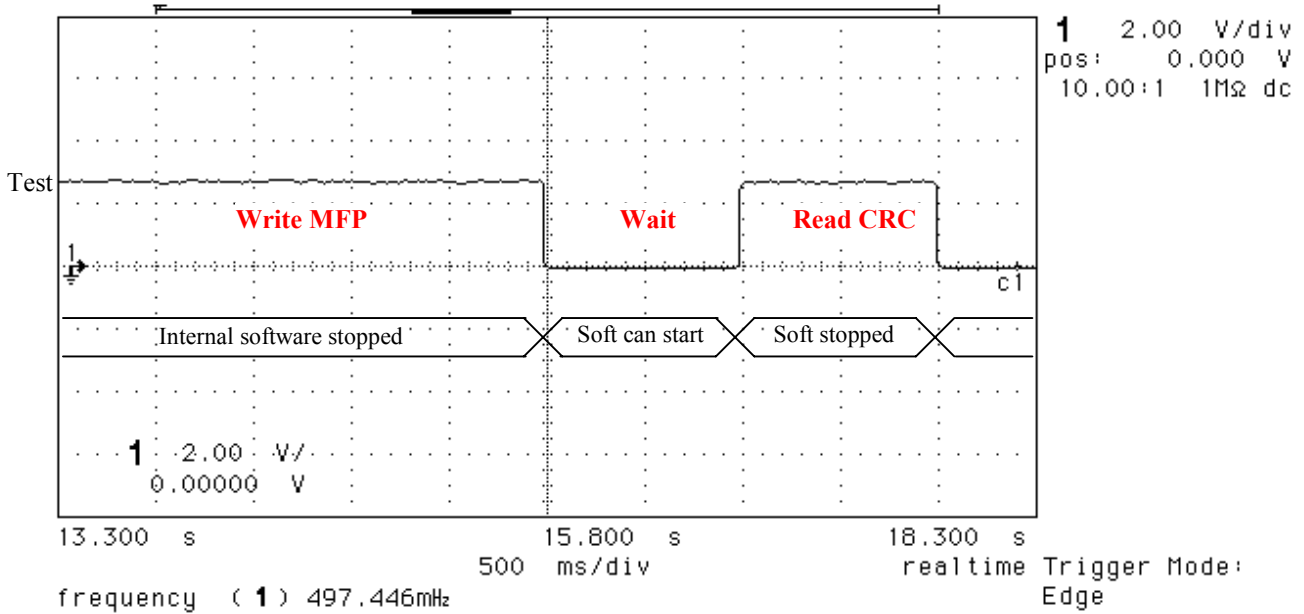
Figure 1.

- Appendix A: ByPort example's program
- Appendix B: ByTimer example's program

You can download this example's program, the latest version of specifications and some other example's programs on our web site. (<http://www.emmarin.com/>)

- End of the programming sequence:

hp stopped

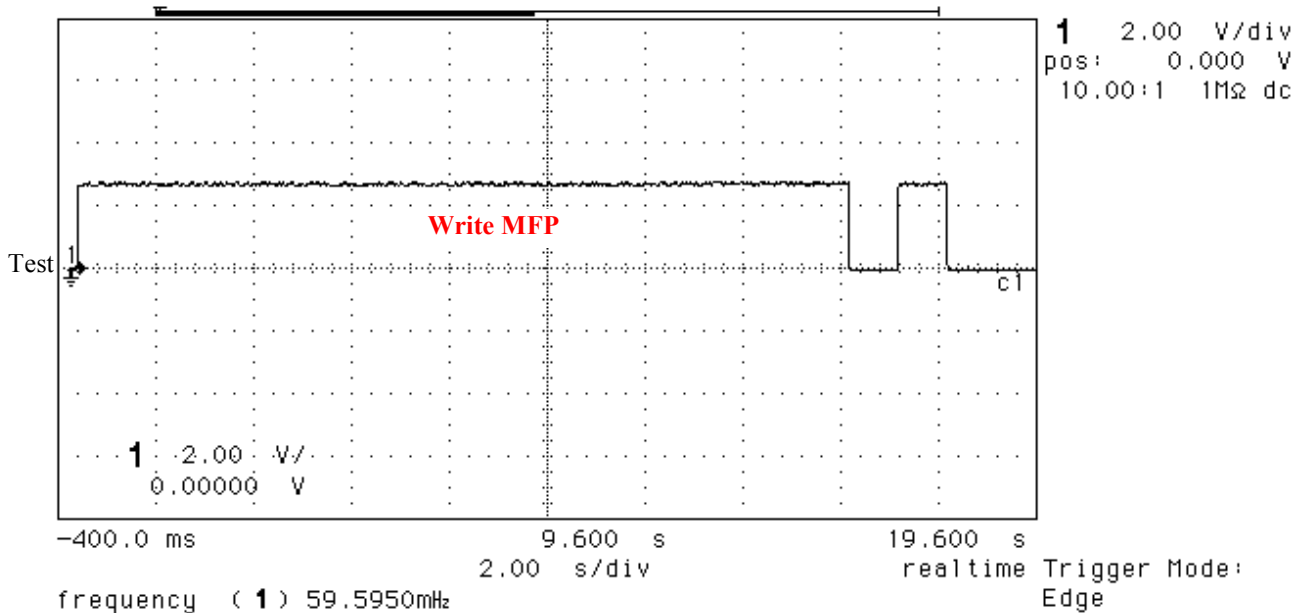


1 1.500 V

Figure 2.

- Complete programming sequence:

hp stopped



1 1.500 V

Figure 3.



Appendix A: ByPort example's program

```
-----
;
; PROGRAM      :      PRP40D01.ASM
; Date last mod :      10/01/2000      Ch.Mayer
; Demo of programming MFP with sleep mode to power up for the EM6540
; Uses the 6640REG.ASM Include file
;
-----
;
; 6640 registers
;
-----
; INCLUDE 6640REG.ASM
;
-----
; Variables
;
-----
STACK0          EQU  030H ; variable used to store the ACCU value during interrupts
;
-----
; Code Start
;
-----

ORG 0

Reset: JMP Main

;
; Interrupt Handler
;
-----
Handler:
    STA  STACK0          ; save the accu value

    LDR  STACK0          ; restore the accu value
    RTI                    ; end of IRQ
;
-----
; Main
;
-----
Main:
Prog:  STI  RegSys3, 0011b ; Disable the watchdog logic and oscillator
      LDI  01H            ; \
      AND  RegPBData     ; => If PB0 = 0 then don't start
      JPZ  Prog          ; /

Start: LDI  04H            ; \
      AND  RegSys2       ; => Test if start from the sleep mode.
      JPNZ Next          ; /
      STI  OPTPaRST, 03H ; No Reset in active mode, and "OR mode" in reset
      STI  OPTInpRSel1, 00H ; \
      STI  OPTInpRSel2, 01H ; => Reset on PA0
      STI  RegSys2, 04H   ; Sleep enable
      STI  RegSys1, 04H   ; Sleep mode

Next:  Nop                ; Next program
      Nop
      JMP  Next
;
-----
; END
;
-----
```



Appendix B: ByTimer example's program

```
-----
; PROGRAM      :      PRT40D01.ASM
; Date last mod :      10/01/2000      Ch.Mayer
; Demo of programming MFP with sleep mode to power up for the EM6540
; Uses the 6640REG.ASM Include file
;-----
;
; 6640 registers
;-----
; INCLUDE 6640REG.ASM
;-----
; Variables
;-----
STACK0          EQU  030H  ; variable used to store the ACCU value during interrupts
;-----
; Code Start
;-----
; ORG 0
Reset: JMP Main
;-----
; Interrupt Handler
;-----
Handler:
; STA  STACK0          ; save the accu value
; LDR  IRQ3            ; Valid the interrupt counter
;
; LDR  STACK0          ; restore the accu value
; RTI                  ; end of IRQ
;-----
; Main
;-----
Main: STI  RegSys3, 0011b  ; Disable the watchdog logic and oscillator
Prog: STI  IRQM3, 02H     ; Enable the interrupt counter
      STI  RegCCnt1, 05H  ; 8 Hz frequency
      STI  RegCDataL, 0FH ; \
      STI  RegCDataM, 00H ; => Load the value 0FH
      STI  RegCDataH, 00H ; /
      STI  RegCCnt2, 09H  ; Load the value and start the counter
      HALT
;
Start: LDI  04H           ; \
      AND  RegSys2       ; => Test if start from the sleep mode.
      JPNZ Next         ; /
      STI  OPTPaRST, 03H ; No Reset in active mode, and "OR mode" in reset
      STI  OPTInpRSel1, 00H ; \
      STI  OPTInpRSel2, 01H ; => Reset on PA0
      STI  RegSys2, 04H  ; Sleep enable
      STI  RegSys1, 04H  ; Sleep mode
;
Next:  Nop               ; Next program
      Nop
      JMP  Next
;-----
; END
;-----
```