# The Real Time Clock

## Built-in RTC with the HMS81C7332

## INTRODUCTION

The HMS81C7332 has dual oscillator, main and sub-oscillator. The sub oscillator used to run 32.768 KHz crystal source, and we can make watch time interval interrupt then it is normally used as real time clock source, furthermore this device has power saving mode and one minute timer interrupt function, then we are able to lengthen the battery life with one minute interrupt and sub oscillator. The HMS81C7332 features three characteristics for RTC application as follows, power saving mode, wide operating voltage and one minute watch timer. Under sub oscillator active mode, its current is very low, 100uA roughly. And this device operates in wide voltage range, 1.8V ~ 5.5V. Therefore it can be able to be biased by normal battery.: The watch timer function can make one minute interval interrupt. Under main power saving mode, the watch timer generates interrupt to wakeup main oscillator and counts up the clock hour and minute. After calculating clock data, MCU switch on power saving mode again. Then we can minimise power consumption, because main oscillator works on several 10us long at every minute.

This RTC solution is an effective method which compose of one minute watch timer, power saving mode, sub colck and external interrupt. Compare to standalone RTC device, this HMS81C7332 built-in RTC solution is a cost effective RTC method.

## DESCRIPTION

Figure 1 is a typical block diagram to apply the RTC application. The HMS81C7332 microcontroller is used and the oscillator are 4MHz as main clock and 32.768KHz as sub one. Figure 2. explains power consumption timing.

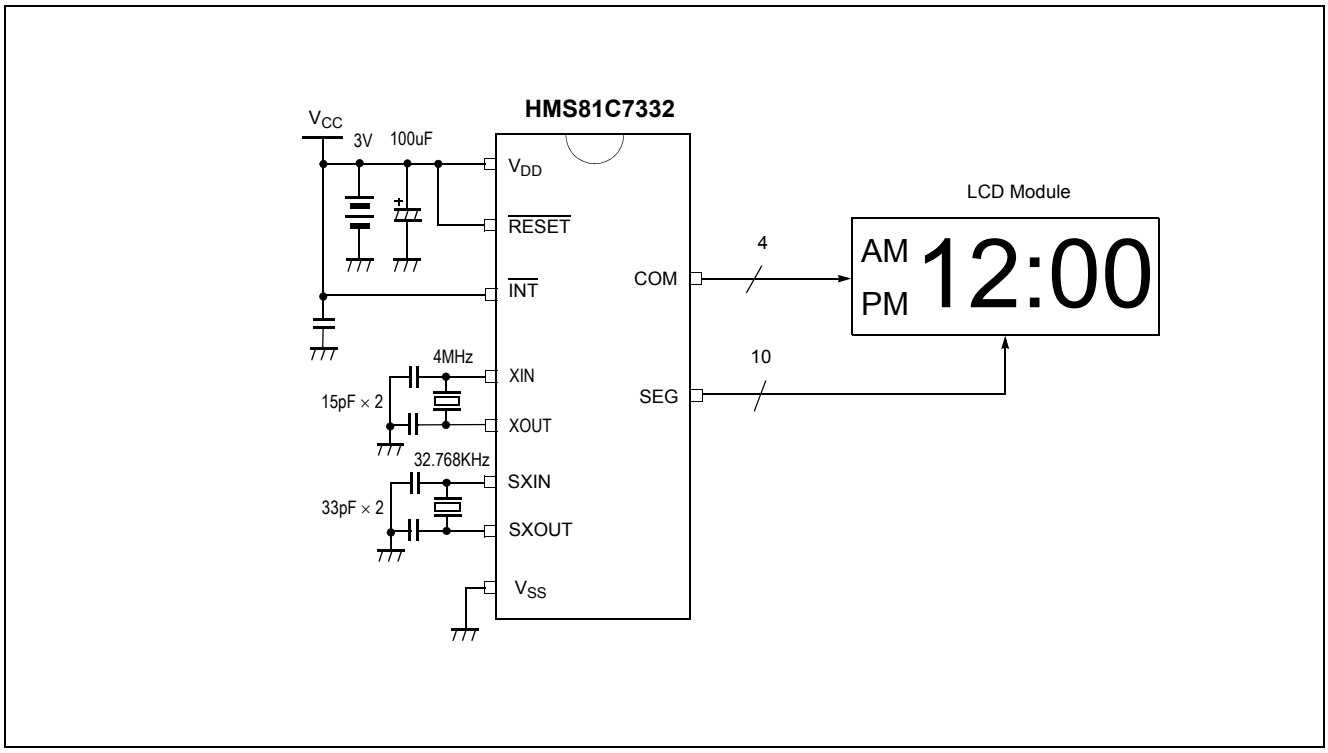


**Figure 1. Circuit Block Diagram**

## HARDWARE

The operating voltage of HMS81C7332 is 1.8V to 5.5V, the lower level of $V_{DD}$, the better power saving effect. As an example, two AAA type battery voltage is enough to support $V_{DD.}$ And $\overline{RESET}$ pin is connected to $V_{DD}$. The INT2 pin is used to detect main power off and on, which is triggered by falling edge or rising edge and is able to detect input level too. The watch timer generates one minute interval interrupt by sub oscillator source. And it has a function that wake up STOP mode. In this condition, the STOP mode means main oscillator stop, sub oscillator operates for watch timer. After STOP mode release, MCU run normal operating. The current consumption is 100uA at STOP mode and 1.5mA at normal operating. There are several power saving mode, STOP mode, sub active mode, sleep mode and other STOP mode which stops all oscillator. In this application, the power saving mode means STOP mode which run sub oscillator and watch timer. When main power is off, MCU runs STOP mode, and every watch timer interrupt wakeup STOP mode for real time clock updating for short terms only. Sometimes low cost RTC application uses capacitor backup insted of battery backup. This style RTC requires large capacitor, but the backup time is limited. So we have to trade off cost and backup time.

In figure 2, $T_{RTC}$ is the execution time of watch timer interrupt subroutine and $T_{ISR}$ is the execution time of INT2 subroutine. Both $T_{ISR}$ and $T_{RTC}$ are very short term, 20 microseconds roughly.

## SOFTWARE

The RTC application program composes of three parts, Initialization, Power fail detection and power save mode switching, and watch timer interrupt subroutine.

The RTC initialization block is a part of firmware initial program by power-on reset, which starts watch timer working one minute interval interrupt, and enable external interrupt pin and interrupt pending at falling and rising edge.

The power fail detecting interrupt subroutine is conposed of detecting main power fail or main power-on state, when MCU detects power fail, it switch on power saving mode, main oscillator stop, but sub oscillator is still alive because sub oscillator is the frequency source of RTC and LCD display. In this example, we use INT2 as power fail detection pin and this port is also used for level detection..

The watch timer interrupt subroutine updates real time clock data, hour and minute. When watch timer interrupt is pended, the MCU is waked up automatically and main oscillator restart working. As soon as finishing clock data update, MCU has to switch on power saving mode again, then main oscillator stop and power consumption is low. The main oscillator runs on short term $T_{RTC}$ in Figure 2. therefore the average power consumption on the term of main power off is very low. The watch timer function is used to wake up power saving mode, and update clock hour and minute, and go back to power saving mode again.

An example program is attached to Appendix A.

*Author:* *Beong-jin Lim*
*MCU application team*
*e-mail: beongjin.lim@magnachip.com*

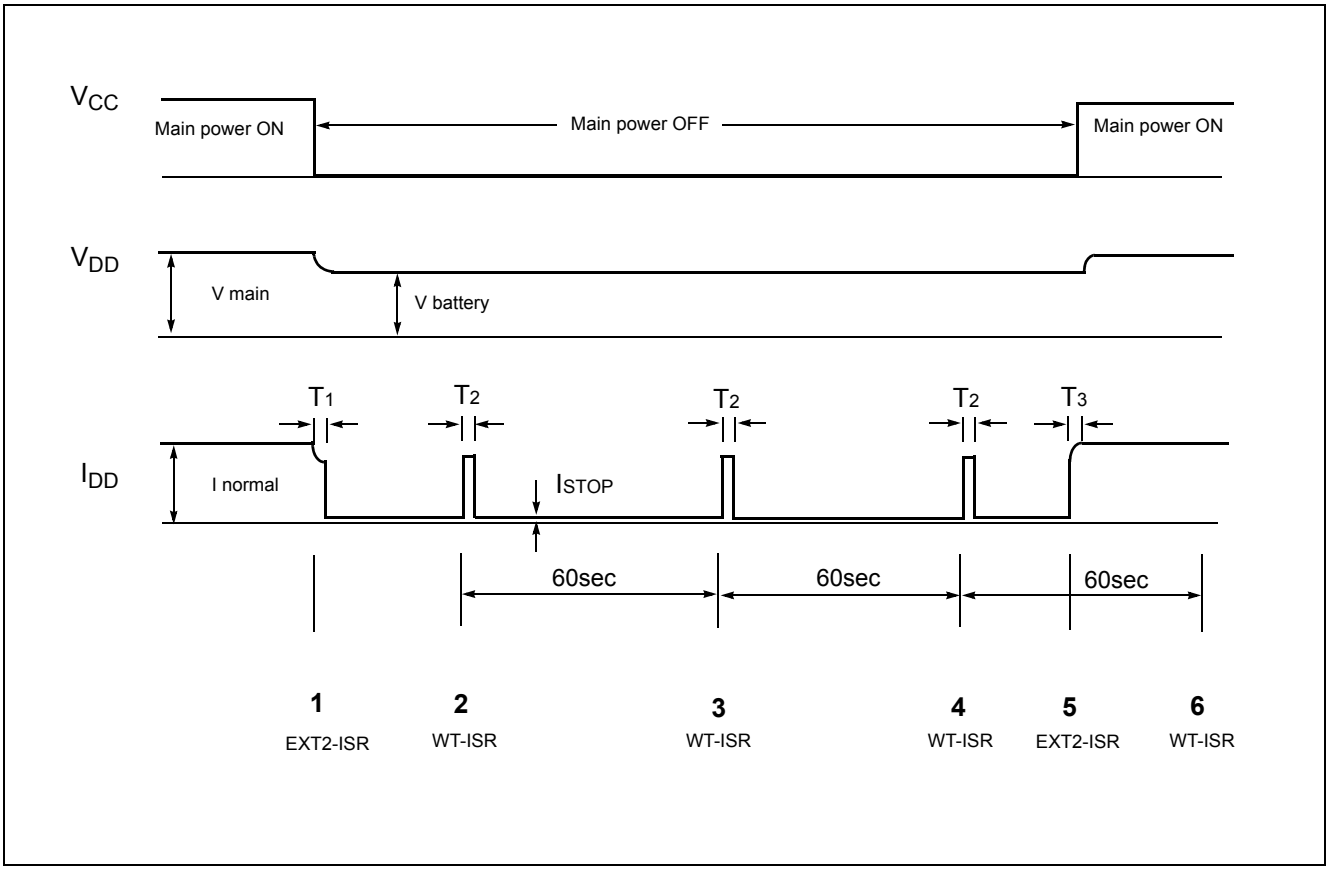


**Figure 2. The RTC wake up Timing Diagram**

## Appendix A:

HMS800 series MICOM ASSEMBLER

```
                ;*******************************************************************************
                ;Company        : MagnaChip Semiconductor Ltd. MCU Application Team
                ;Programmer     : BeeJay Lim
                ;-------------------------------------------------------------------------------
                ;TITLE          : The Real Time Clock Application
                ;Device         : HMS81C7332
                ;OSC            : Main 4MHz / Sub 32.768KHz
                ;*******************************************************************************

;-------------------------------------------------------------------------------------------------------
                ; Peripheral Register assignment

                include     <HMS81C7332.H>       ;

                p_PowerOn   equ     R11          ; high is power on state? (INT2 pin )

;-------------------------------------------------------------------------------------------------------
                ; User Memory(RAM) assignment

                mStatus     ds      1            ;
                f_PowerFail equ     0, mStatus   ; request power saving mode

                mClk_Hour   ds      1            ; Clock Hour data
                mClk_Minute ds      1            ; Clock Minute data


;-------------------------------------------------------------------------------------------------------
                ; external function proto type assignment

                extrn label Process_Task1        ;
                extrn label Process_Task2        ;
                extrn label Process_Task3        ;
                extrn label Clock_Display        ;

;-------------------------------------------------------------------------------------------------------

                ; Vector Table

                ORG     0FFE0H
                DW      Watch_Timer              ; Watch timer interrupt for clock update

                ORG     0FFF6H
                DW      Ext2_INT                 ; External interrupt 2  for power fail

                ORG     0FFFEH
                DW      Program_Start            ; Power On Reset

;-------------------------------------------------------------------------------------------------------

                ; Main program

                org         8000h

Program_Start:
                DI                               ; Disable global interrupt
                ldx         #0FFh                ;
                txsp                             ; stack point initialize
                clrg                             ; RAM0 page
                  .
                  .
                  .
```

```
                ldm         R1, #00h            ;
                ldm         R1IO, #---- ---0-b  ; R11 input for level detection
                ldm         R1PU, #00h          ;
                ldm         R1OD, #00h          ;
                ldm         R1FUNC, #---- --1-b ; R11 define INT2
                  .
                  .
                  .
                  .
                clr1        f_PowerFail         ; clear request power saving mode
                  .
                ldm         WTMR, #1100_1100b   ; Sub Clock fsub x 2e*14 x (count+1) =
                ldm         WTR,  #1111_0111b   ; 0x80 | (119+1)
                  .
Wakeup_start:
                ldm         IENH, #---1 ----b   ; INT2 enable
                ldm         IENL, #---1 ----b   ; Watch Timer enable
                ldm         IRQH, #0            ; Clear All Interrupt Request
                ldm         IRQM, #0            ;
                ldm         IRQL, #0            ;
                ldm         IEDS, #--11 ----b   ; Int2 both falling and rising Edge triggered
                  .
                  .
                ldx         #0FFh               ;
                txsp                            ; stack point initialize
                EI                              ; Enable global interrupt
;-------------------------------------------------------------------------------------------------

main:
                clrg                            ; set Zero page
                  .
                  .
                call        !Process_Task1      ; execute task1
                call        !Process_Task2      ; execute task2
                call        !Process_Task3      ; execute task3
                  .
                  .
                  .
PowerSaveMode:                                  ;
                bbc         f_PowerFail, main   ;

                clr1        f_PowerFail         ; clear request power saving mode in

                ldm         IENH, #---1 ----b   ; INT2 enable
                ldm         IENL, #---1 ----b   ; Watch Timer enable
                ldm         SSCR, #0Fh          ; sleep mode enable (sub clock only enable)
                nop                             ;
                stop                            ; stop !!!

                nop                             ;
                nop                             ;
                jmp         !Wakeup_start       ;
;-------------------------------------------------------------------------------------------------

Ext2_Int:                                       ; Power fail detect and Power saving mode in !!!
                bbc         p_PowerOn, Detect_fail ; R11 == high or low ?
Detect_On:
                  .                             ; now detect power on state
                  .                             ;
                jmp         !Wakeup_start       ; restart watch timer and interrupt

Detect_fail:      .                             ;
```

```
                .                           ; now detect power fail state
                .                           ; stop all power consumption path
                .                           ;
              set1      f_PowerFail         ; request power saving mode
                .                           ;
              reti                          ;
;-----------------------------------------------------------------------------------------------

Watch_Timer:
              push      A                   ;

              inc       mClk_Minute         ;
              lda       mClk_Minute         ;
              cmp       #60                 ; one hour?
              bcc       Clock_Updated       ;
              ldm       mClk_Minute, #0     ;
              inc       mClk_Hour           ;
              lda       mClk_Hour           ;
              cmp       #24                 ; one day ?
              bcc       Clock_Updated       ;
              ldm       mClk_Hour, #0       ;
Clock_Updated:
              call      !Clock_Display      ; clock display

              bbs       p_PowerOn, WatchTimer_end ; R11 == high or low ?

              set1      f_PowerFail         ; request power saving mode at power fail
WatchTimer_end:
              pop       A                   ;
              reti                          ;

;-----------------------------------------------------------------------------------------------
;
;             end of file
```