HT48 & HT46 Traffic Light Controller

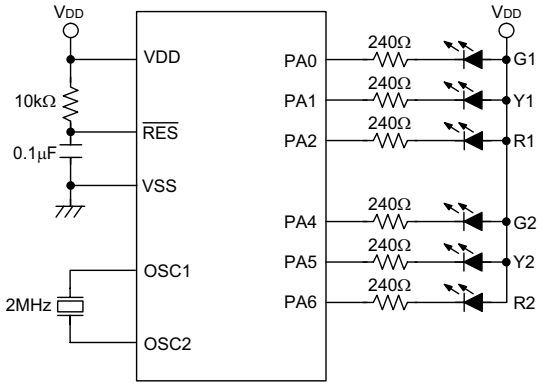
D/N：HA0010E

Introduction

This application model uses 3 LEDs, red, green and yellow to simulate a traffic light controller at an intersection between two streets. Upon starting, R1 and G2 lights up, after a period of time delay, the green light flashes and changes to yellow light, then another period of time delay, after which R2 and G1 lights up. This kind of cycle repeats continuously in this manner simulating a road intersection traffic light. The time interval by which the traffic light lights up can be adjusted according to one′s requirements.

Application Circuits

This circuit uses port PA′s PA0~PA2 and PA4~PA6 pin connections to simulate a road intersection traffic light control signals. Its functions are specifically explained in the subsequent program. The following is the circuit diagram.



HT48R10A-1

Program Example

```
include ht48r10a-1.inc
;------------------------------
data .section 'data'             ; data section
count1         db          ?     ; time delay variable 1
count2         db          ?     ; time delay variable 2
count3         db          ?     ; time delay variable 3
flash          db          ?     ; LED flash control register
rglight        db          ?     ; LED control register
;------------------------------
code .section at 0 'code'        ; program section
       org     00h               ; program entrance address
       jmp     start
start:                           ; start of program
       mov     a, 0        ;(1)  ;
       mov     pac, a            ; set PA as output
       mov     pa, a             ; clear PA
;------------------------------
loop:                            ; LED loop
       mov     a, 0              ;
       mov     tblp, a           ; initial table flag
       tabrdl  rglight     ;(2)  ; read table last page contents to
                                 ; LED control register
       mov     a,rglight   ;(3)  ;
       mov     pa,a              ; output LED register contents
                                 ; to PA
       call    delayl      ;(4)  ; call delayl subroutine
       inc     tblp        ;(5)  ; increment the table pointer
       mov     a,6               ;
       mov     flash,a           ; set LED flash control register
                                 ; to blink 6 times
flashlamp:                       ; LED flash loop
       tabrdl  rglight           ; read table last page contents to
                                 ; LED control register
       mov     a,rglight         ;
  mov  pa,a                      ; output LED register contents to PA
       call    delays      ;(6)  ; call delays subroutine
       inc     tblp              ; increment the table pointer
       sdz     flash             ; check LED flash
       jmp     flashlamp         ; still flashing, jump to LED loop
       tabrdl  rglight           ; read table last page contents to
                                 ; LED control register
       mov     a,rglight         ;
       mov     pa,a              ; output LED register contents to PA
       call    delaym      ;(7)  ; call delaym subroutine
       inc     tblp              ; increment the table pointer
;------------------------------;
```

```
        tabrdl  rglight          ; read the table last page contents
                                 ; to LED control register
        mov     a,rglight        ;
        mov     pa,a             ; output the LED register contents
                                 ; to PA
        call    delayl           ; call delayl subroutine
        inc     tblp             ; increment the table pointer
        mov     a,6              ; set the number of times the LED
                                 ; blinks
        mov     flash,a          ; store the number of times the LED
                                 ; blinks at the LED flash control
                                 ; register
flash2: ;                        ; LED flash2 loop
        tabrdl  rglight          ; read the table last page contents
                                 ; to LED control register
        mov     a,rglight        ;
        mov     pa,a             ; output LED register contents to PA
        call    delays           ; call delays subroutine
        inc     tblp             ; increment the table pointer
        sdz     flash            ; check LED flash
        jmp     flash2           ; still flashing, jump to LED flash2
                                 ; loop
  tabrdl        rglight          ; read table last page contents to
                                 ; LED control register
        mov     a,rglight        ;
        mov     pa,a             ; output LED register contents to PA
        call    delaym           ; call delaym subroutine
        jmp     loop             ;
;--------------------------------
delayl proc                      ; call delayl subroutine
        mov     a,0fh
        mov     count1,a
        mov     count2,a
        mov     count3,a
d1:     sdz     count3
        jmp     d1
        sdz     count2
        jmp     d1
        sdz     count1
        jmp     d1
        ret
delayl endp
;--------------------------------
delaym proc                      ; call delaym subroutine
        mov     a,07h
        mov     count1,a
        mov     a,0ffh
        mov     count2,a
```

```
        mov     count3,a
d2 :    sdz     count3
        jmp     d2
        sdz     count2
        jmp     d2
        sdz     count1
        jmp     d2
        ret
delaym endp
;------------------------------
delays proc                     ; call delays subroutine
        mov     a,0ffh
        mov     count1,a
        mov     count2,a
d3 :    sdz     count2
        jmp     d3
        sdz     count1
        jmp     d3
        ret
delays endp
;------------------------------
        org     300h            ; LED table
        dc      0ebh, 0fbh, 0ebh, 0fbh, 0ebh, 0fbh, 0ebh, 0dbh
        dc      0beh, 0bfh, 0beh, 0bfh, 0beh, 0bfh, 0beh, 0bdh
end
```

Program Explanation

During program initialization (1) first setup the input/output control registers and specify which pins are inputs and which are outputs. In this example, set PAC to ″0″ and PA as an output port. Since the traffic light cycle always run in a fixed pattern, we can use the look-up table in determining the PA value, (2) but to use this table we have to setup the table pointer first. After setting up the table pointer, the table data can be retrieved from the last page using the ″TABRDL″ instruction. The maximum value is ″03FFH″, hence the start address of the last page within the Program ROM is ″0300H″. In (5), the table pointer is incremented. In (2), data from the table last page is read to the LED control register and to the output port to execute LED display. Since different LEDs have different timing parameters that determine the length of each light interval, hence the need for different time delays as shown in the program′s three time delay subroutines; (4) delayl, (7) delaym and (6) delays.