**APPLICATION OPERATION**

This application note describes a method for implementing a Random Timer Light Switch using the 8-pin PIC12CXXX series of 8 pin, 8 bit microcontrollers. The Random Timer Switch could be used as a regular light switch to turn the light on and off. When the Random Timer Switch is set to the random mode, the Switch’s light sensor kicks in. When it is night, the switch will start the first random timer. At the end of the first random period, the Switch turns the light on. After the light is turned on, a second random timer starts, which will make the light stay on for a random period of time, before turning the light off at the end of the second random period.

When the sw1 (Figure 1) is switched to OFF, the relay is switched off by q1, and the light is off. The LED is also turned off. When the sw1 is switched to ON position, the relay is energized, and both the light and LED are turned on. If the sw1 is switched to AUTO, then the circuit is turned into the Random Timer Mode. The LED will blink 1 second on and 1 second off. The circuit polls the pin 3 to see if the LDR (s1) senses the darkness. If s1 senses darkness, the first random delay starts. The random delay period is between 0 seconds to 1 hour. When the first random delay time is up, q1 is switched on so that the relay is energized and the light comes on. This starts the second random delay. The second random delay period is between 1 to 4 hours. When the second random delay reaches time-up, the relay is switched off. s1 is sensing to see if it is daytime. If it is, then s1 waits for night time to begin again. When evening comes, the circuit recycles the above procedure. The software for the Random Timer Switch uses the Pseudo Random Number Generating algorithm modeled in Figure 1.

**FIGURE 1: PSEUDO RANDOM NUMBER GENERATOR**

The Random Timer Switch will turn the light on once every night at different times, with the light staying on for different lengths of time. For those who want to make their unattended house appear to potential intruders that the house is occupied, the Random Timer Switch is the perfect answer.
GRAPHICAL HARDWARE REPRESENTATION

MICROCHIP TOOLS USED

PICSTART™ Plus Development Kit

Assembler/Compiler Version

MPLAB 3.22, MPASM 1.5
APPENDIX A: SOURCE CODE

;****************************************************************************************
; Random Timer Light Switch --- Control Code
; Rosc = 7.5k, Cosc = 33pf ----> 1.87us instruction cycle
; The config bits addressed at fffh is to be set as: 1111110011
; Copyright (c) All right Reserve. J. Zhu (11/96)
;****************************************************************************************

;DEFINE PORT
#define relay gpio, 0
#define led gpio, 1
#define manu gpio, 2
#define auto gpio, 3
#define ldr gpio, 4

;DEFINE CONSTANT:
carry   equ     0h
c       equ 0h
zero    equ     2h
z       equ     2h
work    equ     0h
f       equ     1h

;DEFINE REGISTER:
 tmr0    equ     0x1
 pcl      equ     0x2
 status  equ     0x3
 fsr     equ     0x4
 osccal  equ     0x5
 gpio   equ     0x6
 temp    equ     0x8
 rand    equ     0x9
 count6  equ     0xa
 count7  equ     0xb

; org 0x00h
goto main

;-------------------------------------------------------------------------
; SUBROUTINE rand8          random number generator
;-------------------------------------------------------------------------
rand8 rlf rand,work
xorwf rand,work
movwf temp
rlf temp,f
swapf rand,work
xorwf rand,work
rlf temp,f
xorwf temp,f
rrf temp,f
rlf rand,f
erand8 retlw 0x0

;-------------------------------------------------------------------------
; SUBROUTINE delay1s       1 sec. delay for 1:64 prescale
;-------------------------------------------------------------------------
delay1s movlw .255
movwf tmr0
dly1slp clrw
xorwf tmr0,work
btfss status,z
goto dly1slp
edly1s retlw 0x0

;-------------------------------------------------------------------------
main   movlw b'11010101' ; enable internal tmr0 with 1:64 prescale
        option
movlw b'11111100'
tris
movlw 0x21
movwf rand ; initializing the seed
bcf relay ; turn off the relay
bcf led ; turn off the led

swchk btfss auto ; is sw1 on the auto position
goto random ; yes, goto random mode
btfsc manu ; no
goto on ; it is on the ON position, goto on
goto off ; it is on the OFF position, goto off

; on   bcf relay ; on position: turn on relay
            led ; turn on led
    goto swchk ; go back to check the switch sw1 positions

; off  bcf relay ; off position: turn off relay
            led ; turn off led
    goto swchk ; go back to check the switch sw1 positions

; random  bcf led ; random mode: turn on led
               ldr ; is it night yet
    goto dark ; yes, go to dark routine
btfss auto ; no, is sw1 still on auto position
goto ranend ; no, goto ranend
call delays1 ; yes, call 1 sec. delay
bcf led ; turn off led
btfss ldr ; is it night
    goto dark ; yes goto dark
btfss auto ; no, is sw1 still on auto position
goto swchk ; no, goto swchk
call delays1 ; yes call 1 sec. delay
    goto random

; ranend  bcf led
            relay
    goto swchk ; random mode end with led and relay turned off

; dark  call rand8 ; call for 8 bit random number generator
    movf rand,work
movwf count7 ; move rand(random number) to count7
darklp2 movlw 0x7
movwf count6
darklp1 bcf led ; blink led 1 sec. On and 1 sec. Off while check
            ldr ; the darkness and sw1 position
btfsc ldr
    goto random
btfss auto
goto ranend
bcf led
call delays1
btfsc ldr
goto random
btfss auto
goto swchk
decfsz count6
goto darklp1
decfsz count7 ; first random delay is done?
goto darklp2

; fix   bcf relay ; yes, turn on the relay
        movlw 0x0 ; set for fixed one hour delay and blink the led
fixlp2  movlw 0x7
  movwf count6
fixlp1  bsf   led
  call  delay1s
  btfsc  ldr
  goto  random
  btfss  auto
  goto  ranend
  bcf   led
  call  delay1s
  btfsc  ldr
  goto  random
  btfss  auto
  goto  ranend
  decfsz  count6
  goto  fixlp1
  decfsz  count7
    ;fixed delay is done?
  goto  fixlp2

; ran  call  rand8
    ;generate another random number
  movf  rand.work
    ;set up for the second random delay
  movwf  count7
    ;relay stays on while led blink in 1 second interval
ranlp2  movlw 0x15
  movwf  count6
ranlp1  bsf   led
  call  delay1s
  btfsc  ldr
  goto  random
  btfss  auto
  goto  ranend
  bcf   led
  call  delay1s
  btfsc  ldr
  goto  random
  btfss  auto
  goto  ranend
  decfsz  count6
  goto  ranlp1
  decfsz  count7
    ;second random delay is done?
  goto  ranlp2

;  bcf    relay
    ;turn off the relay
again  bsf   led
    ;blink led and chk if sw1 is still on auto position
  btfsc  auto
    ;if it is, check to see if it is day time
  goto  ranend
    ;if it is, go to random routine and wait for another
night  btfsc  ldr
    
  goto  random
  call  delay1s
  bcf  led
  btfsc  auto
  goto  ranend
  btfsc  ldr
  goto  random
  call  delay1s
  goto  again

; end