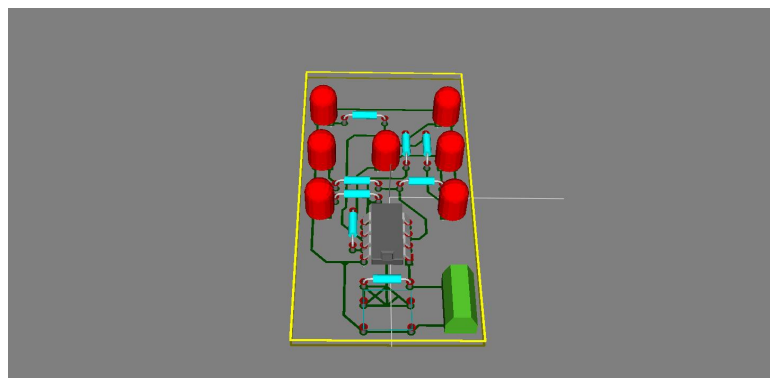
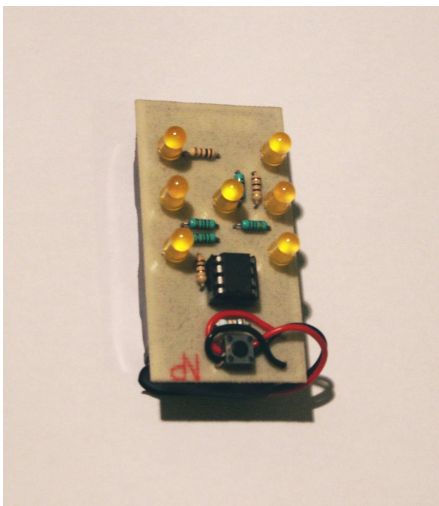


- 1 R 47 kΩ
- 7 R 100 Ω
- 7 Led
- 1 PIC12F508
- 1 bouton pousoir
- 1 coupleur 2 pile R6



```

list p=12f508
#include P12F508.INC
config _MCLRE_OFF & _CP_OFF & _WDT_OFF & _IntRC_OSC
val de equ H'0008'
var1 equ H'0009'
var2 equ H'000A'
var3 equ H'000B'
var4 equ H'000C'
tmp equ H'000D'

;*****
;* Programme principal *
;*****
org 0x00
movwf OSCCAL
btfsc STATUS,GPWUF ; si reveil aller à debut:
goto debut

movlw B'00000001' ; si power-on, définir la valeur 1
movwf val_de ; pour la variable val_de

debut: movlw B'01011111' ; wake-up enabled, pull-ups disabled, clock source internal osc:4:256
option

movlw B'00101000' ; GP0,1,2,4 en sortie et GP3,5 en entrée
tris GPIO
call conv ; convertire la valeur en leds allumées
movwf GPIO ; et afficher

bt0: btfss GPIO,0x03 ; attendre si bouton appuyé
goto bt0
call t10ms
movlw D'120' ; initialisation des variable de la boucle de 30s (120x250ms=30s)
movwf var1

b30s0: decfsz var1,F ; 30s passées ?
goto b30s1 ; non -> continuer la boucle 30s
movlw B'00000000' ; oui -> eteindre les leds
movwf GPIO
call t10ms ;
movf GPIO,W ; lecture du port GP avant la l'instruction sleep
sleep ; mise en veille
nop

b30s1: movlw D'249' ; debut de la boucle de 250ms
movwf var2
b30s2: decfsz var2,F ; 250ms passées ?
goto b30s3 ; non -> continuer la boucle 250ms
goto b30s0 ; oui -> revenir au debut de la boucle 30s

b30s3: call tmp1ms
btfss GPIO,0x03 ; bouton appuyé ?
goto tirg ; oui -> sortir de la boucle et commencer le tirage
nop ; non -> continuer dans la boucle 250ms
goto b30s2

tirg: call t10ms
movlw B'00111111' ; led eteinte (GPIO en entrée)
tris GPIO

trg1: incf val_de,F ; augmenter la valeur du dé
movlw 0x07
xorwf val_de,W
btfss STATUS,Z ; val_de=7 ?
goto trg2 ; non -> continuer
movlw 0x01
movwf val_de ; oui -> val_de=1
btfss GPIO,0x03 ; bouton relâché ?
goto trg1 ; non -> continuer le tirage

call t10ms
call conv ; oui -> convertire la valeur en leds allumées
movwf GPIO ; et afficher
movlw B'00101000' ; led allumée (GPIO 0,1,2,4 en sortie)
tris GPIO
goto debut ; retour au début

;*****
;* Convertir la valeur en leds allumées *
;*****
conv: movf val_de,W
movwf tmp
decf tmp,F
btfsc STATUS,Z
retlw B'00000001' ; val_de=1 w=00000001, 1 led allumée
decf tmp,F
btfsc STATUS,Z
retlw B'00000010' ; val_de=2 w=00000010, 2 leds allumées
decf tmp,F
btfsc STATUS,Z
retlw B'00000011' ; val_de=3 w=00000011, 3 leds allumées
decf tmp,F
btfsc STATUS,Z
retlw B'00000110' ; val_de=4 w=00000110, 4 leds allumées
decf tmp,F
btfsc STATUS,Z
retlw B'00000111' ; val_de=5 w=00000111, 5 leds allumées
decf tmp,F
btfsc STATUS,Z
retlw B'00010110' ; val_de=6 w=00010110, 6 leds allumées
retlw B'00010111' ; 7 leds allumées si valeur non prévue

;*****
;* Tempo de 1ms *
;*****
tmp1ms: movlw d'199' ;debut tempo 1ms
movwf var3

tmpa: decfsz var3,F
goto tmpb
goto tmpc

tmpb: goto tmpa

tmpc: retlw 0x00 ; fin tempo 1ms

;*****
;* Tempo de 10ms *
;*****
t10ms: movlw d'10'+1
movwf var4

t10a: decfsz var4,F
goto t10b
retlw 0x00

t10b: call tmp1ms
goto t10a

;*****
end

```

Le fichier .HEX servant à la programmation du pic doit contenir la suite hexadécimale suivante.

```
:0200000040000FA  
:100000002500E306050A010C28005F0C0200280CFD  
:1000100006003409260066070B0A5009780C2900EF  
:10002000E902180A000C260050090602030000002D  
:10003000F90C2A00EA021D0A100A49096607220A79  
:1000400000001A0A50093F0C0600A802070C88019C  
:1000500043072C0A010C28006607250A50093409B9  
:100060002600280C0600050A08022D00ED004306B4  
:100070000108ED0043060208ED0043060308ED0009  
:1000800043060608ED0043060708ED004306160880  
:100090001708C70C2B00EB024E0A4F0A4B0A000848  
:0E00A0000B0C2C00EC02550A00084909520A0C  
:021FFE00EA0FE8  
:00000001FF
```