

تكنولوجيا مولود قاسم

PIC 16F84A  
HAMZA - Ain Turck

الميكروكنترولر

**(MICROCONTROLLER)**

# PIC 16F84

## تطبيق

التحكم في ثنائي مضيء موضوع  
في منفذ RB1 بواسطة  
زر ضاغط موجود علي  
منفذ PORTA(RA0)  
لمدة 0.5 ثانية

## AFFECTATION DES ENTREES

تعيين

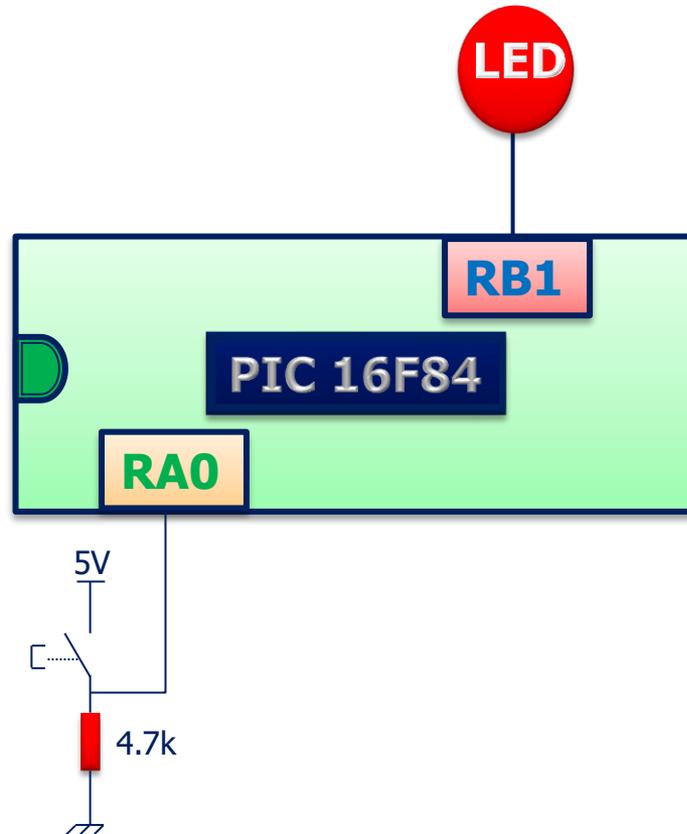
المدخل

Entrée	Broche PIC
Bp	RA0

## AFFECTATION DES SORTIES

تعيين المخرج

Sorties	Broche PIC
LED	RB1



# PROGRAMME EN ASSEMBLEUR AVEC MPLAB

برنامج في لغة التجميع ببرنامج MPLAB

```
LIST    p=16F84
#include <p16F84.inc>
```

```
__CONFIG  _CP_OFF & _WDT_OFF & _PWRTE_ON & _XT_OSC
```

```
#define LED  PORTB,1
```

```
#define BP  PORTA,0
```

```
org 0x000 ; Adresse de départ après reset
```

```
goto  init; Adresse 0: initialiser
```

init

```
bsf    STATUS,RP0  وضع بيت (RPO) للسجل (STATUS) في 1 للذهاب إلى بنك 1
bcf    TRISB,1     وضع البيت 1 لسجل (TRISB) في 0 لكي يكون المنفذ كمرج؛
movlw  1           كتابة 1 في سجل العمل؛
movwf  TRISA       أخذ معلومات السجل w وضعها في بيت 0 لسجل TRISA
bcf    STATUS,RP0  رجوع إلى بنك 0 لكي يكون RA0 كمدخل
goto   debut      اذهب إلى برنامج debut
```



# برنامج

debut البداية

bcf LED

وضع الخروج RB1 في 0

btfs BP

تجاوز إذا كان القيمة يساوي 1 أي زر ضاغط في 1

goto debut

bsf LED

وضع RB1 في 1 أي LED يضيء

call tempo

نداء للبرنامج tempo

bcf LED

وضع RB1 في 0 أي LED ينطفئ

goto debut

ارجع إلى

end

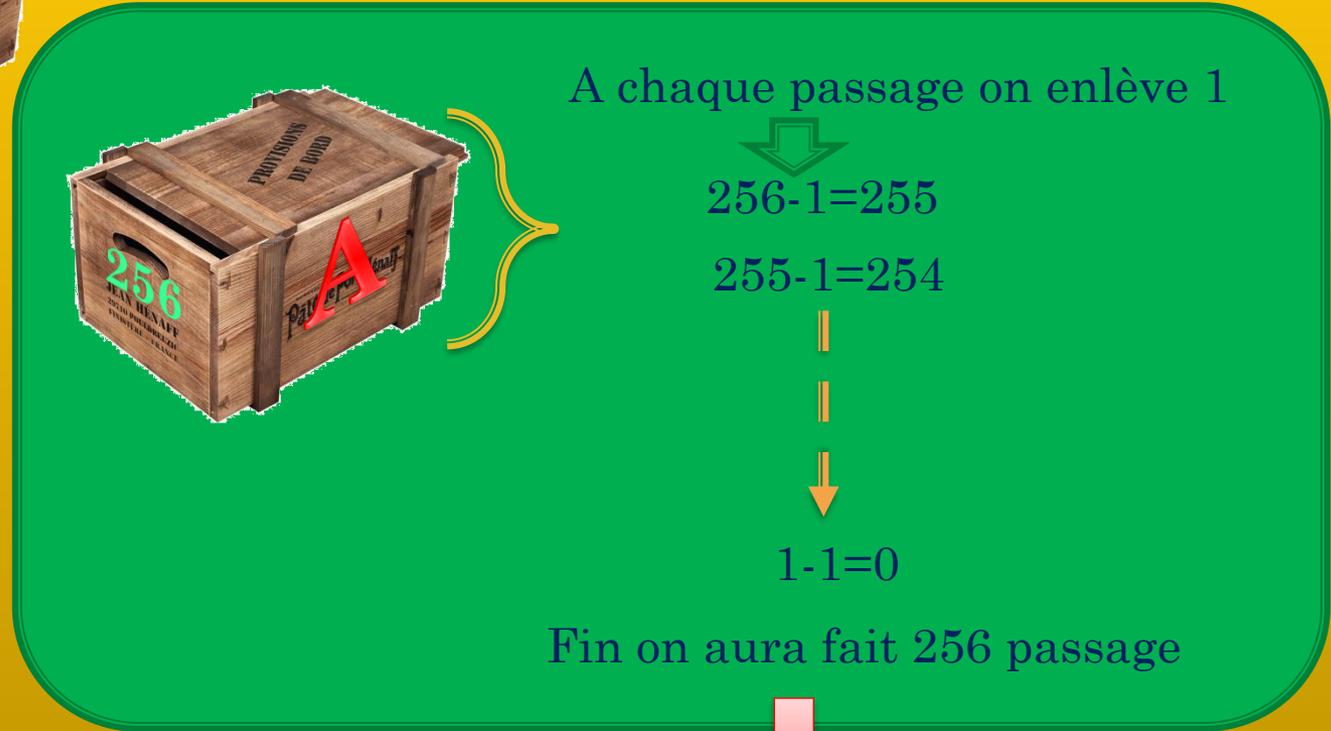
النهاية



A la fin on aura vidé la caisse A 256 fois et B 256 fois  
on aura  $256 \times 256 = 65536$  passages  
Un passage c'est  $1\mu s$  alors  $65536\mu s$



$256 - 1 = 255$



On donne l'ordre pour enlever 1 d'une autre boîte

السجل له 8bit إذا  $2^8 = 256$

الساعة المتكونة من Quartz لها تردد 4Mhz  
أي الزمن هو  $T=1/F$  إذا مع ال-pic يعمل بربع الساعة  
أي  $T/4$  نتحل على  $T= 4/4 = 1\mu s$

نأخذ سجل و ننقص من 256 إلى 0 (0...254..255..256) نتحل على  $256 \mu s$   
إذا أخذنا سجلين  $65536\mu s = 256 \times 256$   
للحصول على 0.5 s اي  $500000 \mu s$

يجب إضافة سجل  $7.6 = 500000/65536$  نستعمل سجل يوجد فيه قيمة 7 فقط

فتتصل على  $(458752=256 \times 256 \times 7)$  مع NOD نصل إلى 0.5 s



# برنامج مؤقتة

tempo

movlw 7      أكتب 7 في سجل العمل W

movwf cmpt3      أكتب معلومات السجل العمل في سجل cmpt3

boucle3      برنامج

clrf cmpt2      امسح عداد 2

boucle2

clrf cmpt1      امسح عداد 1

boucle1

nop      ضياع دورة

decfsz cmpt1, f      انقص 1 من السجل ، إذا كان محتوى السجل يساوي 0 تجاوز التعليمة

goto boucle1      إذا كان محتوى السجل يختلف عن 0 ارجع إلى

decfsz cmpt2, f      انقص 1 من السجل ، إذا كان محتوى السجل يساوي 0 تجاوز التعليمة

goto boucle2      إذا كان محتوى السجل يختلف عن 0 ارجع إلى

decfsz cmpt3, f      انقص 1 من السجل ، إذا كان محتوى السجل يساوي 0 تجاوز التعليمة

goto boucle3      إذا كان محتوى السجل يختلف عن 0 ارجع إلى

return      الرجوع

# PROGRAMME EN ASSEMBLEUR AVEC MPLAB

برنامج في لغة التجميع ببرنامج MPLAB

```
LIST      p=16F84
#include <p16F84.inc>
```

```
__CONFIG  _CP_OFF & _WDT_OFF & _PWRTE_ON & _XT_OSC
```

```
CBLOCK 0x00C ..... بداية حلقة المتغيرات
```

```
    cmpt1:1
```

```
    cmpt2:1
```

```
    cmpt3:1
```

```
ENDC ..... نهاية الحلقة
```

```
#define LED PORTB,1
```

```
#define BP PORTA,0
```

```
org 0x000 ; Adresse de départ après reset
```

```
goto init; Adresse 0: initialiser
```

```
init
```

```
bsf     STATUS,RP0  وضع بيت (RPO) للسجل (STATUS) في 1 للذهاب إلى بنك 1
```

```
bcf     TRISB,1     وضع البيت 1 لسجل (TRISB) في 0 لكي يكون المنفذ كمرج؛
```

```
movlw   1           كتابة 1 في سجل العمل؛
```

```
movwf   TRISA       أخذ معلومات السجل w وضعها في بيت 0 لسجل TRISA
```

```
bcf     STATUS,RP0 رجوع إلى بنك 0 لكي يكون RA0 كمدخل
```

```
goto    debut      اذهب إلى برنامج debut
```

