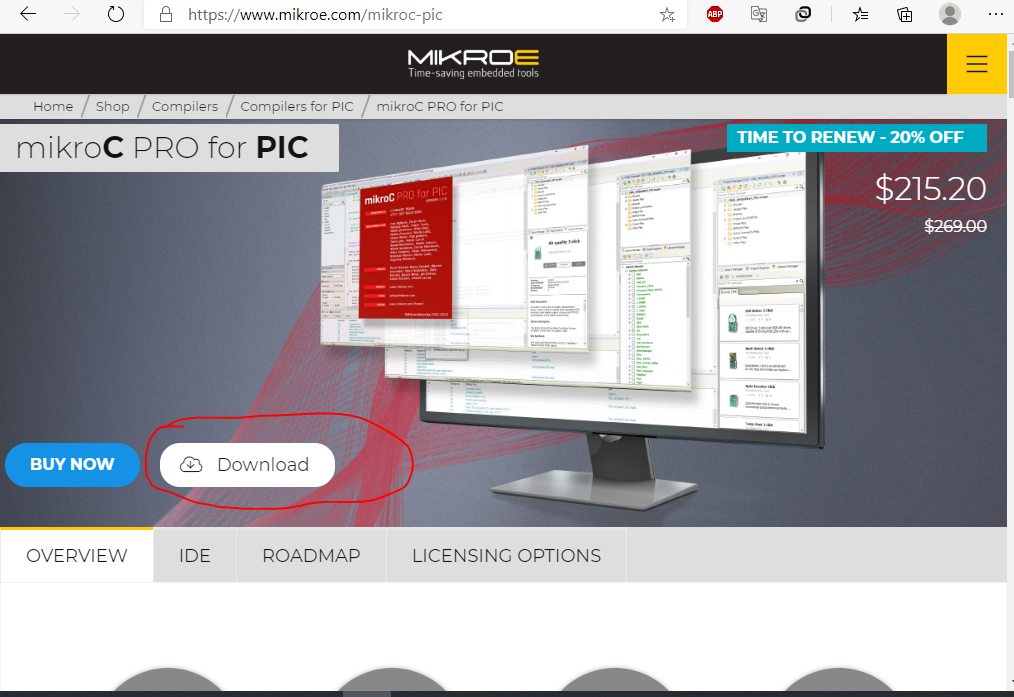
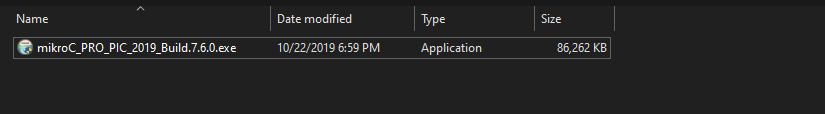
1. **Tải và cài đặt MikroC**

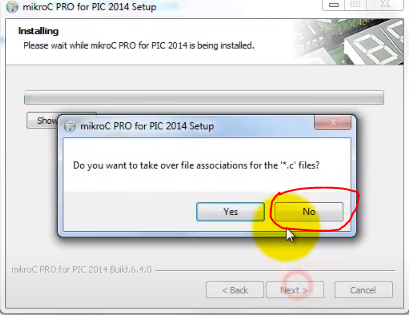
* Truy cập link: <https://www.mikroe.com/mikroc-pic>, chọn download

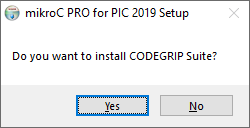


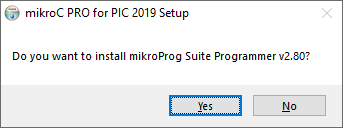
* Giải nén thư mục vừa tải về, “Run as administrator” với file cài đặt

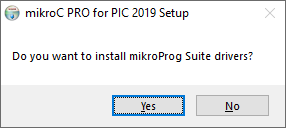


* Chọn “No” ở các mục tiếp theo:



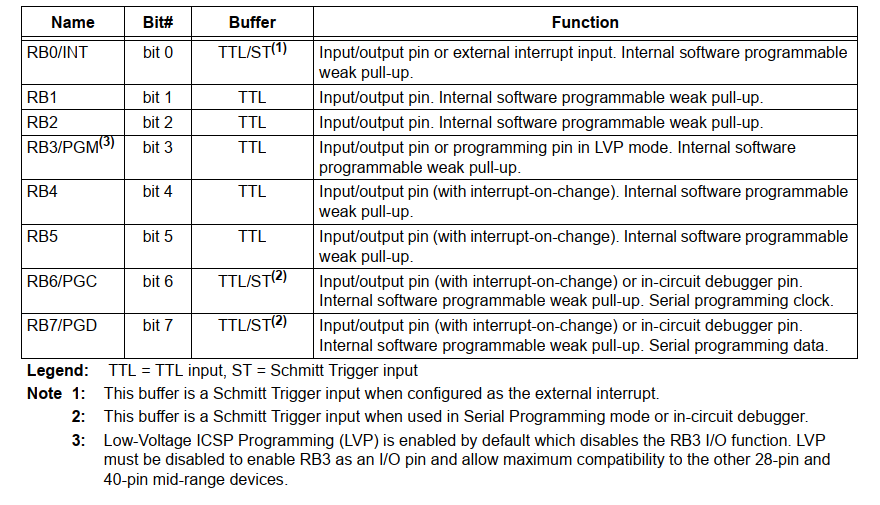






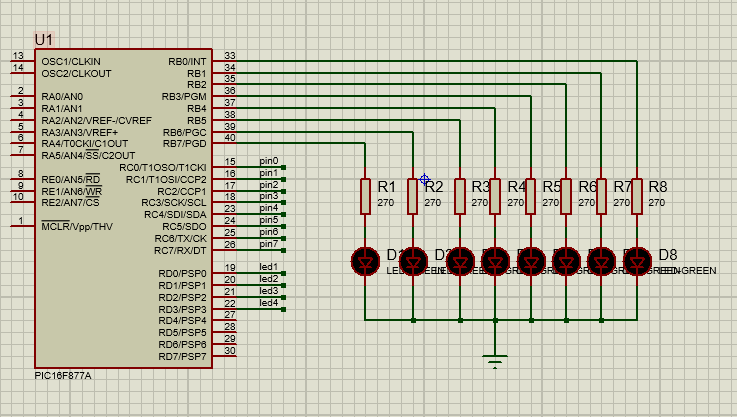
1. **Blynk Led** 
   1. **Port B và thanh ghi TRISB**

* Port B có độ rộng 8bit, là 1 port 2 chiều (vào/ ra)
* Nếu setting 1bit bất kỳ của thanh ghi TRISB = 1 thì chân tương ứng ở PORTB sẽ là input. Ngược lại (set bit = 0) thì chân tương ứng là output



*Chức năng của các pin trong PORTB*

* 1. Sơ đồ nguyên lý



* 1. **Code blynk led**

void main**()** **{**

TRISB **=** 0x00**;** //port B set as output

PORTB **=** 0x00**;** //set port B as Low level

**while(**1**)**

**{**

//blynk 1 port

PORTB **=** **~**PORTB**;**// đảo bit cả port B

Delay\_ms**(**500**);** //delay 500ms

//blynk 1 pin

PORTB**.**B0 **=** **!**PORTB**.**B0**;** // set pin PB0 as Low level

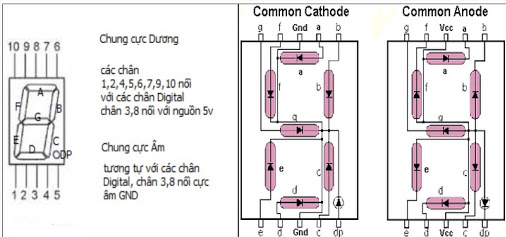
Delay\_ms**(**500**);**

**}**

**}**

1. **Quét led 7 đoạn**
   1. **Sơ lược về led 7 đoạn**

Có 2 loại là Anode chung và Cathode chung

****

* 1. **Cách hoạt động của Led 7 đọan**
* Với loại Anode chung:

Nếu muốn 1 trong 7 led sáng, thì ta setting cho chân tương ứng ở mức Low. Ngược lại nếu muốn tắt thì setting ở mức High

VD: muốn led a sáng, ta sẽ setting cho chân A ở mức Low

* Với loại Cathode chung: muốn led sáng/ tắt thì ta setting ngược lại với loại Anode chung
  1. **Code (Anode chung)**

**A, Led 7 đoạn đơn**

Mảng các chữ số, ở dạng hexa:

char arr\_number**[]** **=** **{**0xC0**,** 0xF9**,** 0xA4**,** 0xB0**,** 0x99**,** 0x92**,** 0x82**,** 0xF8**,** 0x80**,** 0x90**};**

Giải thích:

VD: 0xC0 = 1100 0000 ⬄ DP/G/f/e/d/c/b/a (chân DP, G ở mức High, các chân còn lại của led 7 thanh ở mức Low)

* Lúc này led 7 thanh sẽ hiển thị ra số 0

**Code**

char arr\_number**[]** **=** **{**0xC0**,** 0xF9**,** 0xA4**,** 0xB0**,** 0x99**,** 0x92**,** 0x82**,** 0xF8**,** 0x80**,** 0x90**};**

void Hien\_Thi\_ChuSo**(**int number**)//hàm hiển thị chữ số, sử dụng PortC**

**{**

PORTC **=** arr\_number**[**number**];**

**}**

void main**()**

**{**

**TRISD = 0x00;**

Hien\_Thi\_ChuSo**(**0**);**

Delay\_ms**(**500**);**

**}**

**B, Led 7 đoạn đôi**

Hàm hiển thị số trên led 7 đoạn đôi

void Led7\_Double**(**int number**)**

**{**

int hangChuc **=** number **/** 10**;**

int hangDonVi **=** number **%** 10**;**

Hien\_Thi\_ChuSo**(**hangChuc**);**

led1 **=** 1**;**

Delay\_ms**(**10**);**

led1 **=** 0**;**

Hien\_Thi\_ChuSo**(**hangDonVi**);**

led2 **=** 1**;**

Delay\_ms**(**10**);**

led2 **=** 0**;**

**}**

Ý tưởng: hiển thị chữ số hàng chục và hàng đơn vị lần lượt sử dụng PortD. Khi hiển thị số nào thì on/off led tương ứng đó trong khoảng thời gian vài ms (đây là khoảng thời gian mà mắt người khó phân biệt được led bị tắt, bật)

Code:

sbit led1 at PORTD**.**B0**;**

sbit led2 at PORTD**.**B1**;**

const unsigned char arr\_number**[]** **=** **{**0xC0**,** 0xF9**,** 0xA4**,** 0xB0**,** 0x99**,** 0x92**,** 0x82**,** 0xF8**,** 0x80**,** 0x90**};**

void Hien\_Thi\_ChuSo**(**int number**);**

void Led7\_Double**(**int number**);**

void main**()**

**{**

TRISD **=** 0x00**;** //set as output

TRISC **=** 0x00**;** //set as output

PORTC **=** 0xFF**;**

**while(**1**)**

**{**

**for(**count **=** 0**;** count **<=** 99**;** count**++)//đếm từ - đến 99**

**{**

**for(**i **=** 0**;** i **<=** 50**;** i**++) //delay 1s = (50 x (10+10))ms**

**{**

Led7\_Double**(**count**);**

**}**

**}**

**}**

**}**