

Forum:[Hack Forums](#)

Thread: [Coding C for Assignment - Microchips, Switches, LED's, and Pickits](#)

Underworldproblem - January 6, 2016, 2:53 pm

Hi, I have an Culminating Task to do. 15% of my mark.

I finished creating my circuit on the breadboard.

Image of Breadboard: Spoiler [\(Click to View\)](#)

Everything is powered by the Pickit which connects to the 5pin thing, above the microchip. There is a resistor and capacitor on each switch. Switches are used to change the LED pattern. Each switch = 1 pattern. I grounded each LED correctly and powered them according to the diagram below. Switches connect to the microchip, pin 4 and pin 11.

We use this diagram to know where to placed what into the PIC 16F684 Microchip: Spoiler [\(Click to View\)](#)

```
PORTA = 0b0000001;  
PORTC = 0b0000010;
```

Microchip Pickit 2: Spoiler [\(Click to View\)](#)

We use this to input the code into the microchip, which carries out the pattern to the LED's.

Code:

Spoiler [\(Click to View\)](#)

Spoiler [\(Click to View\)](#)

I am wondering if anyone has any idea about this and knows how to make a pattern on C using this base code, and also turning on a specific LED, and even dimming an LED?

sqsgkrldwg - January 6, 2016, 4:54 pm

(Jan 6, 2016 03:53 PM)? HYFR ? Wrote: [\[CITE\]](#) I am wondering if anyone has any idea about this and knows how to make a pattern on C using this base code, and also turning on a specific LED, and even dimming an LED?

Hi,

The answer is in your own question.

Look at the sample codes.

Underworldproblem - January 6, 2016, 4:55 pm

(Jan 6, 2016 05:54 PM)Lef fe Wrote: [\[CITE\]](#) Hi,

The answer is in your own question.

Look at the sample codes.

Like I know im suppose to change the Port numbers. But, like how do I make a pattern. I dont know if I am explaining it right.

sqsgkrldwg - January 6, 2016, 5:15 pm

(Jan 6, 2016 05:55 PM)? HYFR ? Wrote: [\[CITE\]](#) Like I know im suppose to change the Port numbers. But, like how do I make a pattern. I dont know if I am explaining it right. What do you mean by pattern ?

Lighting some LEDs ? Doing it rythmically ?

I never used to program on that type of chip.

Maybe this documentation will help you (heavy file) :

<http://ww1.microchip.com/downloads/en/DeviceDoc/70157F.pdf>

However, I bet you can light some leds with that for instance, where the hexadecimal values represents the pin I guess :

PHP Code

```
<?php
```

```
#define GREEN_LED 0x1
```

```
#define RED_LED 0x10
```

```
PORTC = GREEN_LED & RED_LED; // Light 2 LEDs
```

If you want to do it in rythm, use the waiting code in the sample (The asm("nop") loop)

0xDEAD10CC - January 6, 2016, 8:12 pm

(Jan 6, 2016 03:53 PM)? HYFR ? Wrote: [\[CITE\]](#) Hi, I have an Culminating Task to do. 15% of my mark.

I finished creating my circuit on the breadboard.

Image of Breadboard: Spoiler [\(Click to View\)](#)

Everything is powered by the Pickit which connects to the 5pin thing, above the microchip. There is a resistor and capacitor on each switch. Switches are used to change the LED pattern. Each switch = 1 pattern. I grounded each LED correctly and powered them according to the diagram below. Switches connect to the microchip, pin 4 and pin 11.

We use this diagram to know where to placed what into the PIC 16F684 Microchip: Spoiler [\(Click to View\)](#)

```
PORTA = 0b000001;
```

```
PORTC = 0b000010;
```

Microchip Pickit 2: Spoiler [\(Click to View\)](#)

We use this to input the code into the microchip, which carries out the pattern to the LED's.

Code:

Spoiler [\(Click to View\)](#)

Spoiler [\(Click to View\)](#)

I am wondering if anyone has any idea about this and knows how to make a pattern on C using this base code, and also turning on a specific LED, and even dimming an LED?

Lol, I doubt you can dim the LED without a variable resistor which controls the current going to the LED. By your breadboard, I don't see one... The only other way is to have a resistor which is toggled in and out of the circuit by some switch so that you can step up or step down the resistance

immediately. It's not going to be a smooth dimming effect this way however. The other obvious way is to run the LED off of a battery -- fairly constant dimming effect as the battery runs out of juice. :)

Also, Binary literals too are usually only a GCC extension, you don't need them IMO. If you understand how to use bitwise arithmetic to check, set, toggle, and clear certain bits, this is easy.