Binary Outputs: LEDs

ECE 376 Embedded Systems

Jake Glower - Lecture #6

Please visit Bison Academy for corresponding
lecture notes, homework sets, and solutions

Binary Outputs: LEDs

Light Emitting Diodes (LED's)

- Are diodes, allowing current to only flow in one direction,
- They convert current to light.(light is proportional to current flow), and
- They are *very* fast, capable of > 100MHz flashes per second, and
- They are a simple way to output binary data (light on / light off)

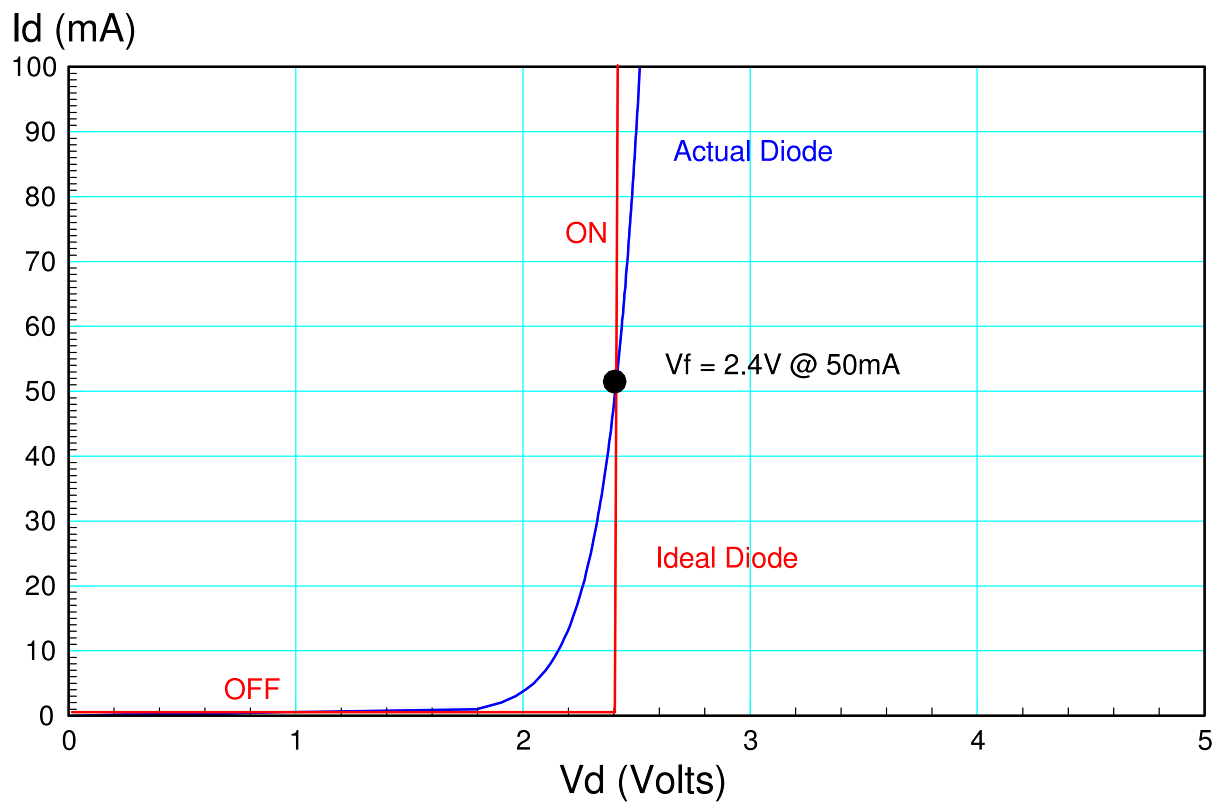
LED VI Characteristics

Exponential in nature

- Makes analysis hard

Ideal Diode

- Vd = constant when Id > 0
- Slightly wrong but often times close enough



Diode Specifications

- Vf: Vd when Id > 0
  - Ideal diode approximation
- mcd: Light output in millicandles
  - One beeswax candle = 1000 mcd
- lux: Light output in lux
  - 100W light bulb = 1000 lux
- Color: Nonscientific term
- Wavelength: More accurate color

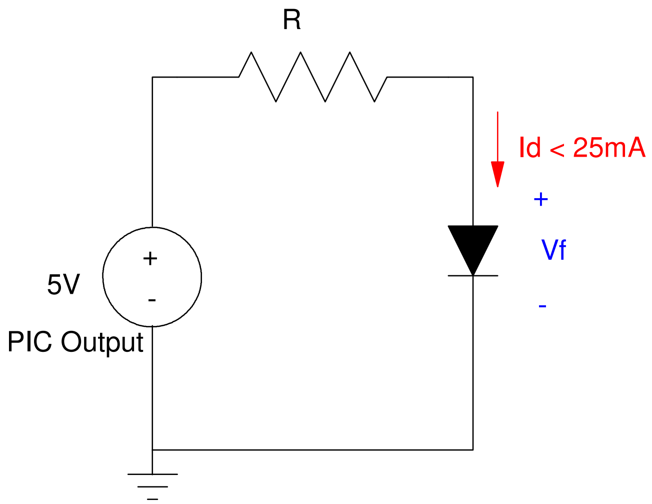
LED	Color	Current	Typical Vf	Typical mcd	Wavelength (nm)	Price ea
1W White Star LED		350mA	3.4V	100 lm	n/a	$1.55
0.5W 10mm White LED		100mA	3.3V	25 lm	n/a	$0.30
Piranah RGB LED	Red	20mA	1.8V	8000 mcd	630 nm	$0.31
	Green	20mA	3.0V	8000 mcd	525 nm	
	Blue	20mA	3.0V	8000 mcd	470 nm	

Diode Circuits

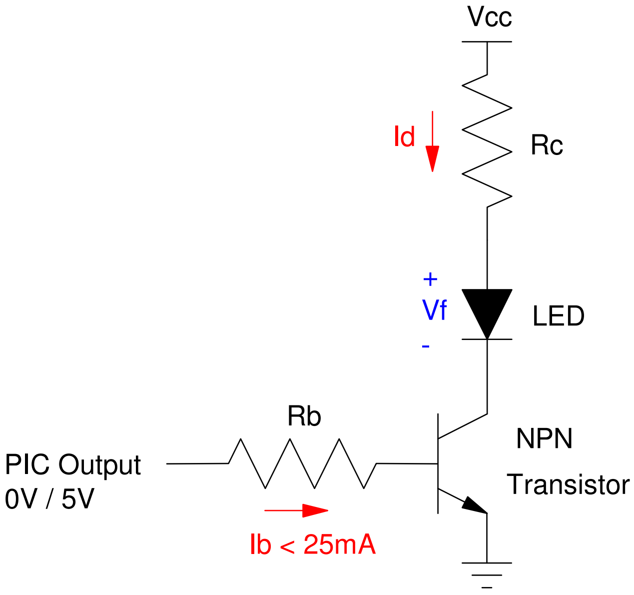
Case 1: Connect with a resistor

- Vf < 5V
- Id < 25mA

Case 2: Connect with a transistor



Case 1: Id < 25mA



Case 2: Id > 25mA

Case 1: Vd < 5V, Id < 25mA

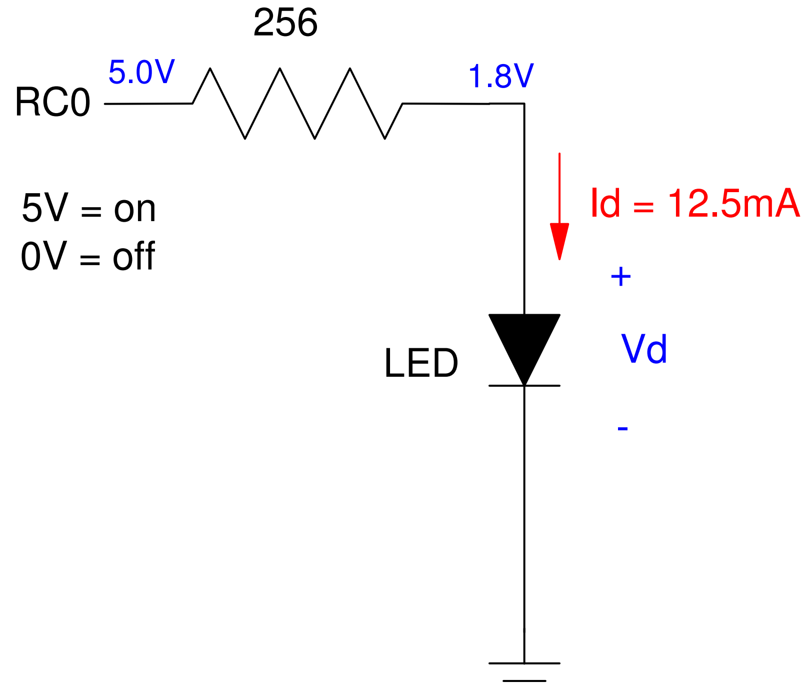
Connect a red LED to a PIC. Set light level to 5000mcd

- Vf = 1.8V @ 20mA, 8,000mcd @ 20mA

Solution: Light is proportional to current

$$I_d = \left(\frac{5000mcd}{8000mcd}\right) \cdot 20mA = 12.5mA$$

$$R_r = \left(\frac{5V - 1.8V}{12.5mA}\right) = 256\Omega$$



Case 2: Id > 25mA

Option #1: Use a solid state relay ($7 solution from www.mpja.com)

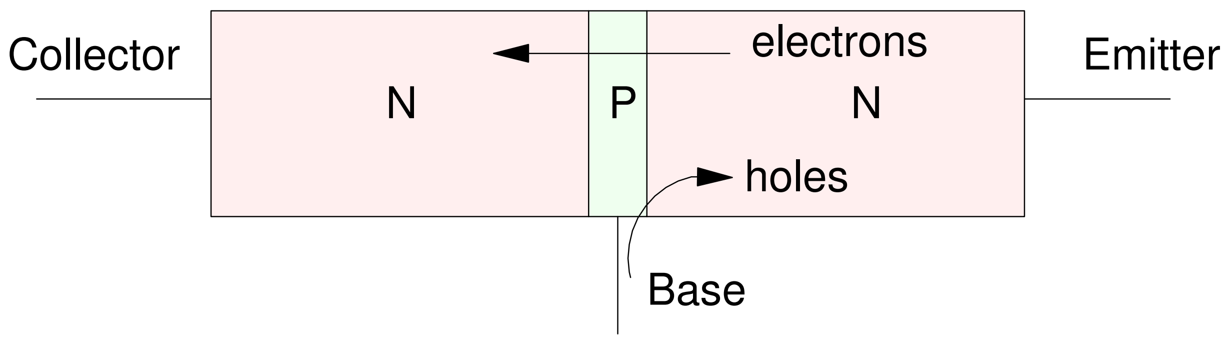
- Input: 3 - 32V DC @ < 10mA
- Output: 5 - 60VDC @ 10A max

Also available for up to 480VAC, 40A

Case 2: Id > 25mA

Option #2: BJT Transistor ($0.53 solution)

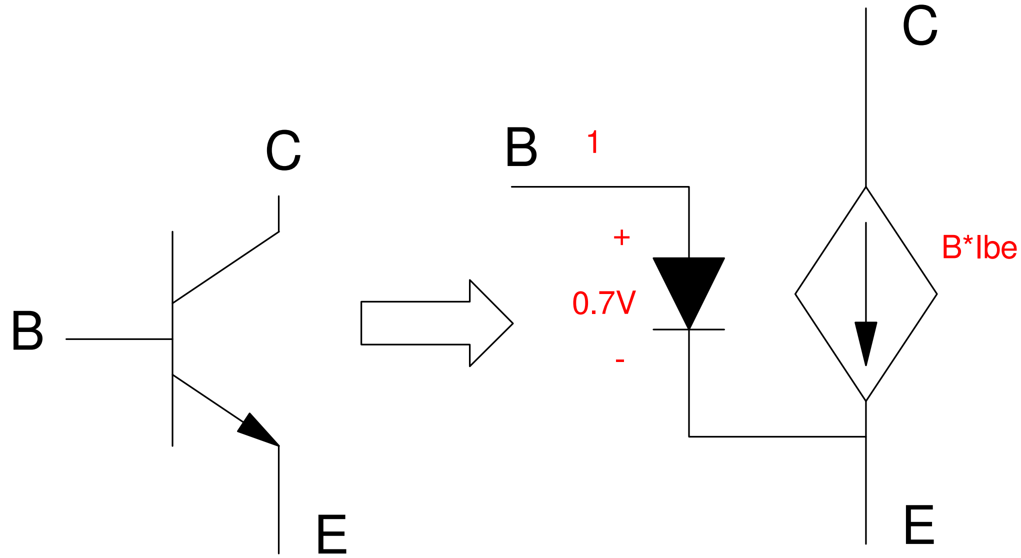
- NPN semiconductor
- Base current can turn on / off the current
  - Saturated (on) when $\beta I_b > I_c$



NPN Transistor: The base current controls the current from collector to emitter

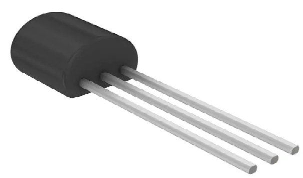
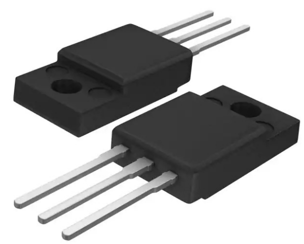
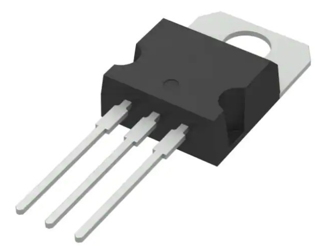
Transistor Model

- Between the base and the emitter is a diode. It takes 0.7V to turn on a Silicon diode.
- Between the collector and the emitter is a current-controlled current source. The current flow amplifies the base to emitter current.



Symbol and Circuit Model for an NPN Transistor: The arrow indicates a diode from base to emitter.

Transistor Specifications

	3904	6144	TIP112
Type	NPN	NPN	NPN
Current Gain	100	200	1,000
max(Ic)	200mA	10A	4A
max(Vce)	40V	50V	40V
Vce(sat)	300mV	360mV	900mV
Vbe	0.7V	0.7V	1.4V
price	$0.037	$0.53	$0.59
Image			

Using a Transistor as a Switch

- Turn on and off a 1W White LED @ 100mA
  - Vf = 3.0V @ 350mA
- Use a 6144 NPN transistor

Pick Ic to set the current

$$R_c = \left(\frac{10V - 3.0V - 0.36V}{100mA}\right) = 66.4\Omega$$

Pick Ib so that $\beta I_b > I_c$

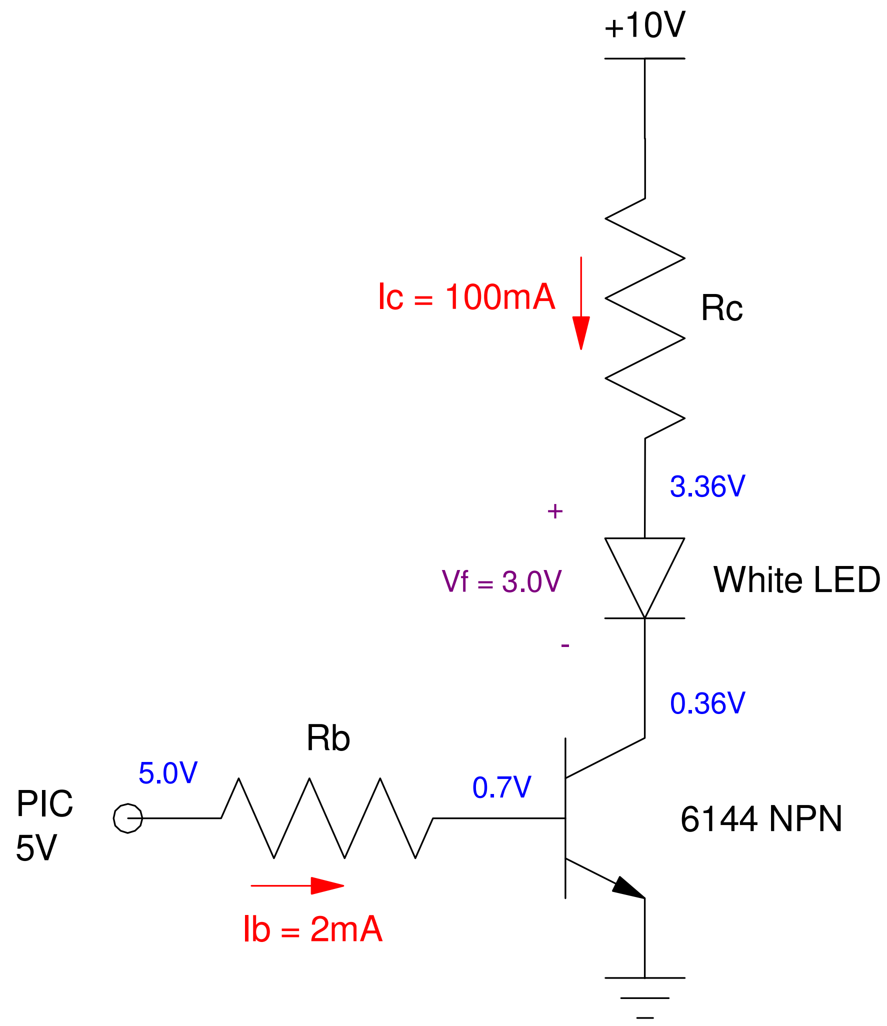
$$I_b > \frac{I_c}{\beta} = \frac{100mA}{200} = 0.5mA$$

Let Ib = 2mA

$$R_b = \left(\frac{5V - 0.7V}{2mA}\right) = 2150\Omega$$

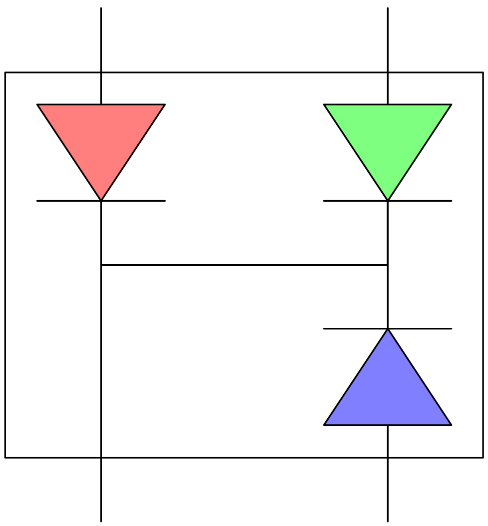
note: Rb is somewhat arbitrary

- $R_b > \left(\frac{5V - 0.7V}{25mA}\right) = 172\Omega$
- $R_b < \left(\frac{5V - 0.7V}{0.5mA}\right) = 8600\Omega$

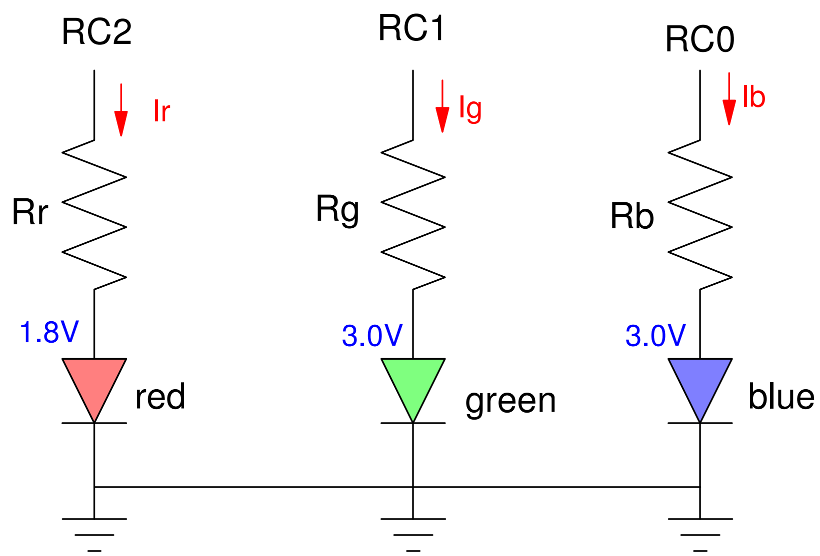


Fun with RGB LEDs

- You can make any color by mixing red + green + blue
- Prianah LED's combine three colors in one LED
- Single pixel in a scoreboard



Piranah Package



Connection to a PIC

LED Flashlight

Build an LED flashlight with the following functions:

- RB0 All lights off
- RB1 Red light on
- RB2 Green light on
- RB3 Blue light on

Define "on" to be 20mA.

Hardware Solution: Previous figure with

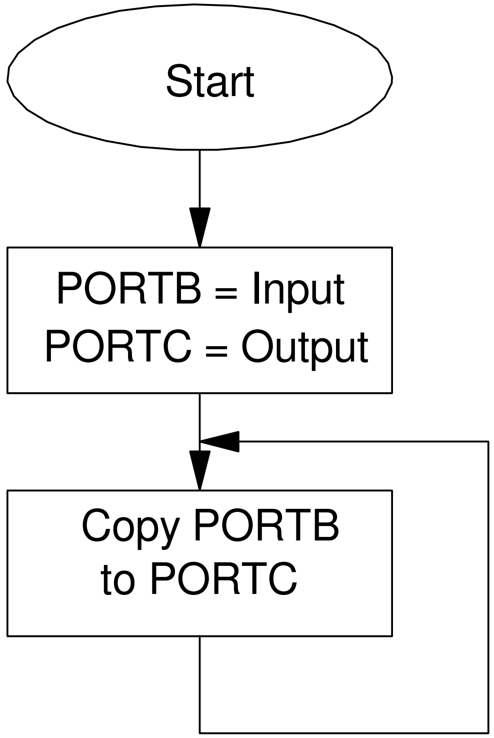
$$R_r = \left(\frac{5V-1.8V}{20mA}\right) = 160\Omega$$

$$R_g = \left(\frac{5V-3.0V}{20mA}\right) = 100\Omega$$

$$R_b = \left(\frac{5V-3.0V}{20mA}\right) = 100\Omega$$

Option 1: Lights on while button is pressed

- No Buttons: LEDs off
- RB0: Turn on Red
- RB1: Turn on Green
- RB2: Turn on Blue



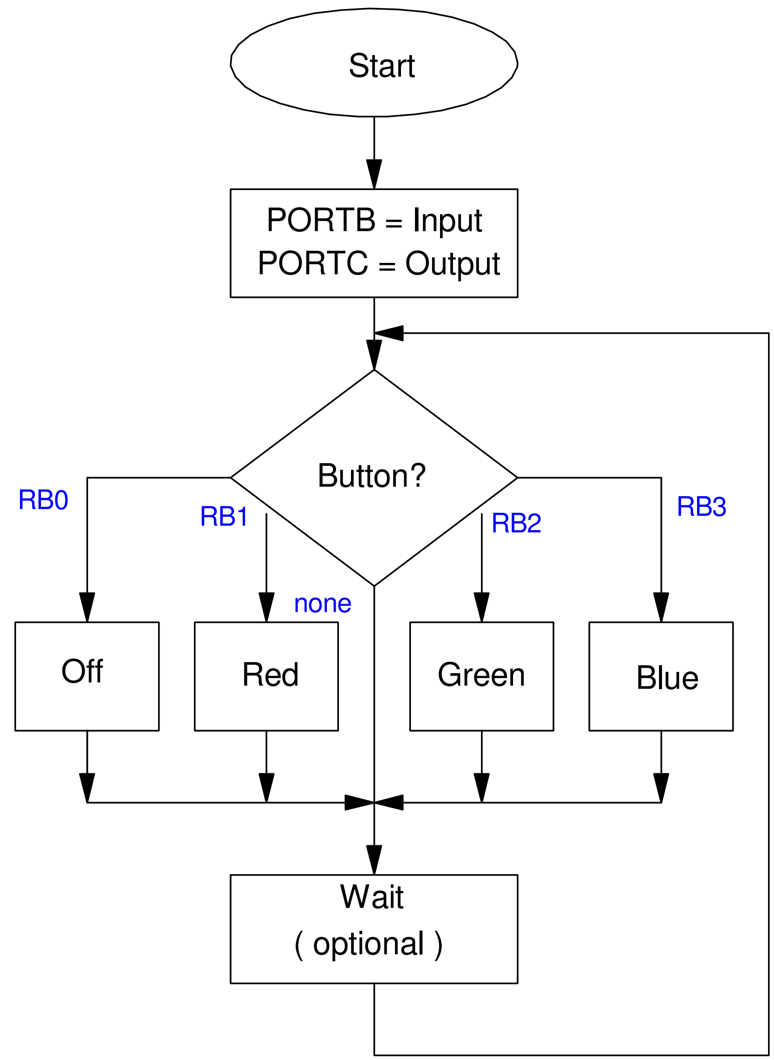
Code:

```
Init:
        movlw 0xFF
        movwf TRISB
        clrf  TRISC
        movlw 0x0F
        movwf ADCON1
Loop:
        movff PORTB, PORTC
        goto  Loop

        end
```

note: Code is a lot more complicated if the hardware shuffles the pins to the LED

Option 2: Lights Remain On

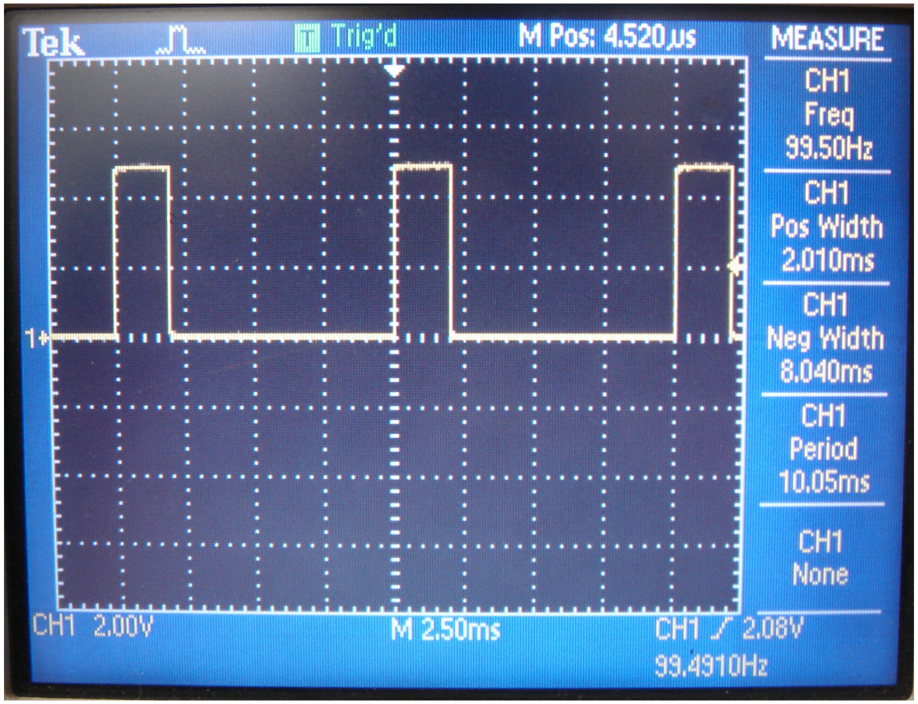
```
COLOR   equ   0

#include <p18f4620.inc>

        org 0x800
        call Init
Loop:
        movf  COLOR,W
        btfsc PORTB,0
        movlw 0
        btfsc PORTB,1
        movlw 1
        btfsc PORTB,2
        movlw 2
        btfsc PORTB,3
        movlw 4
        movwf COLOR
        movff COLOR, PORTC

        call  Wait

        goto  Loop
```



Option 3: Change the brightness to 1600mcd (20%)

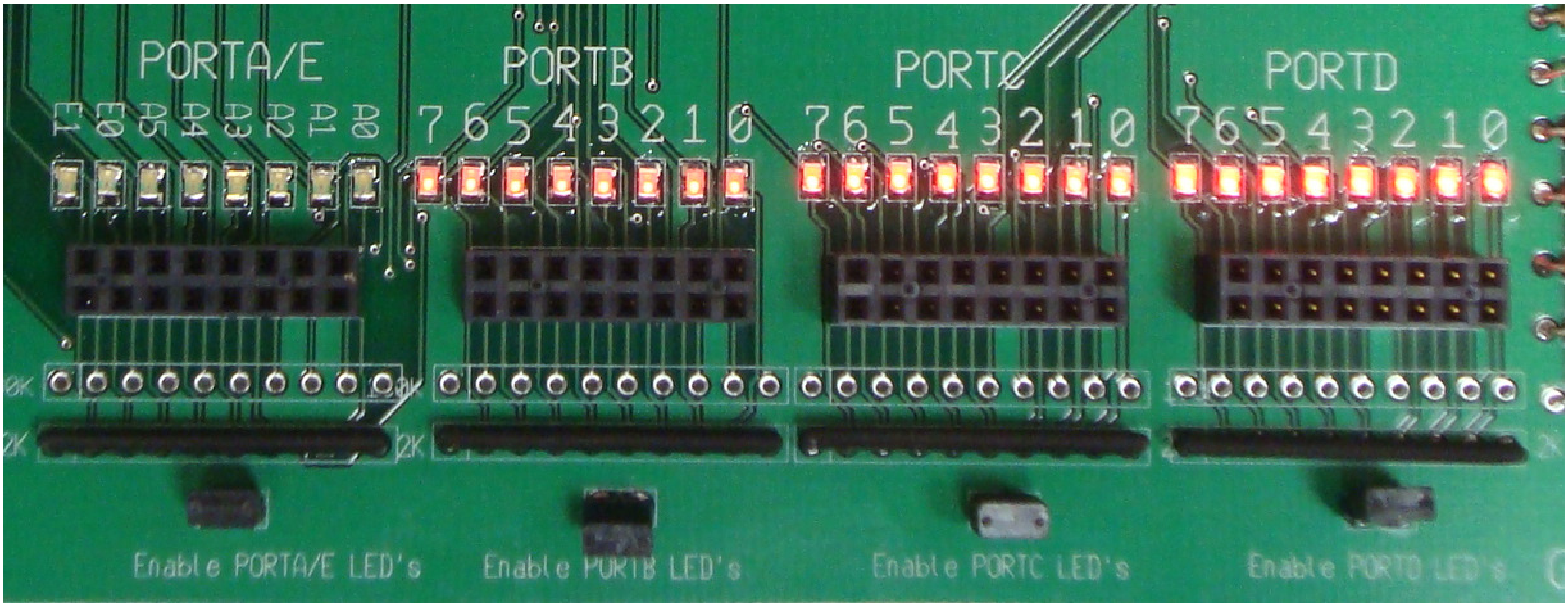
- Hardware: Change R
- Software: Make the duty cycle 20%

```
Loop:
        clrf   PORTC
        call   Wait
        call   Wait
        call   Wait

        movf   COLOR,W
        btfsc  PORTB,0
        movlw  0
        btfsc  PORTB,1
        movlw  1
        btfsc  PORTB,2
        movlw  2
        btfsc  PORTB,3
        movlw  4
        movwf  COLOR
        movff  COLOR, PORTC
        call   Wait

        goto   Loop
```



Pulse Width Modulation

- Changing the duty cycle allows you to adjust how bright the LEDs are via software
- 5% (PORTB), 20% (PORTC), 90% (PORTD)



Summary: LEDs

LEDs are

- Easy to connect to a PIC board
- Easy to turn on and off
- Very fast
  - > 100MHz very common
  - Must faster than a PIC

Hardware:

- If you need less than 5V and 25mA, use a resistor to connect an LED to the PIC board
- If you need more than 5V or more than 25mA, use a transistor

Software:

- Logic 0 (0V) turns off the LED
- Logic 1 (5V) turns on the LED
- PWM allows you to vary the brightness from 0% to 100% in software
- note: you can change this in hardware if you want