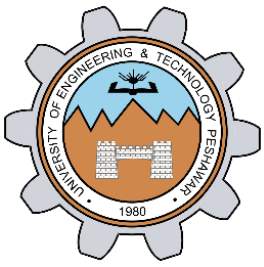


Interfacing button with MSP430 microcontroller and polling based Night rider

Engr. Shahzada Fahim Jan



Testing a bit

C language

```
#define BIT7 0x80
```

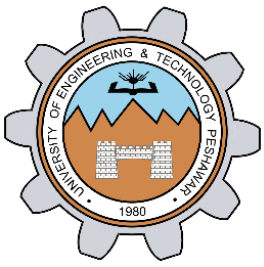
```
unsigned char result;
```

```
if (result & BIT7)
```

```
..... ;
```

```
else
```

```
..... ;
```



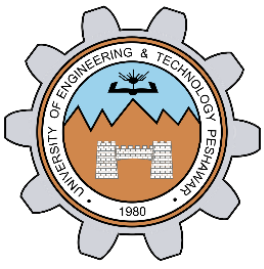
Waiting for a button-press

```
//Input pins are active LOW
```

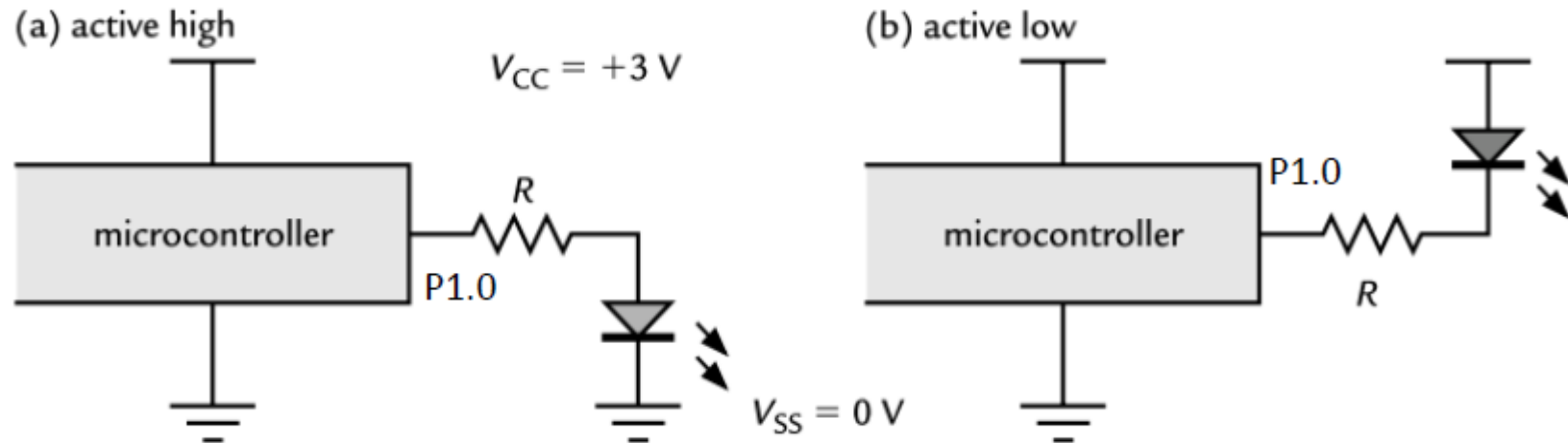
```
#define BIT7 0x80
```

```
while (!(P1IN & BIT7)) //True when button is pressed
```

```
{ //Do Something }
```

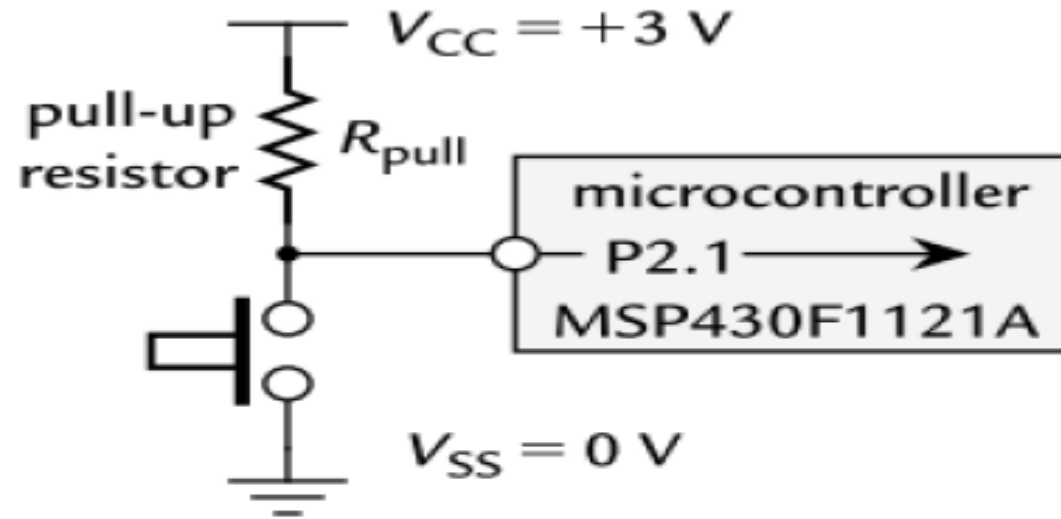
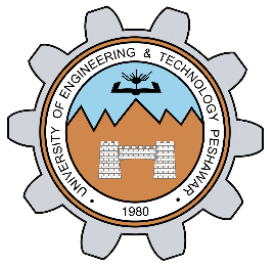


Writing: Active High vs Active Low

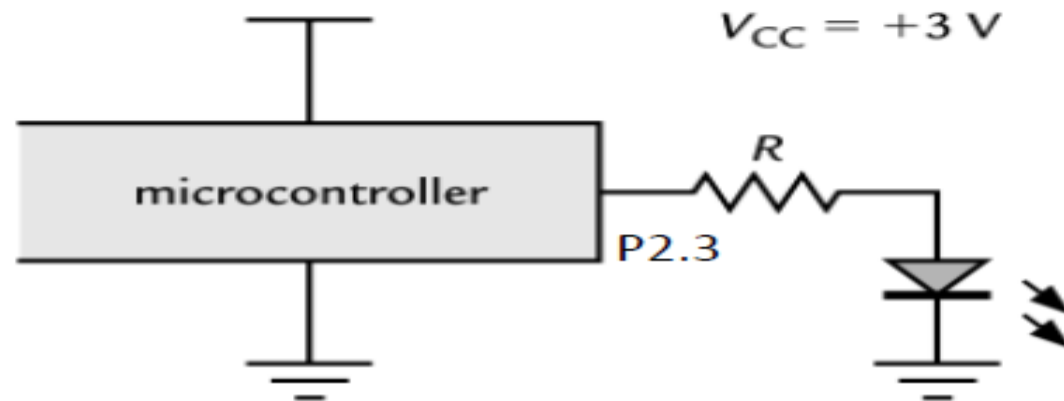


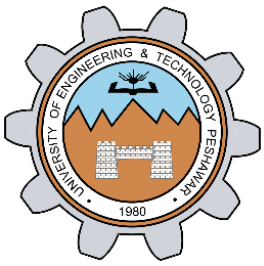
`P1OUT |= BIT0;`

`P1OUT &= ~BIT0;`



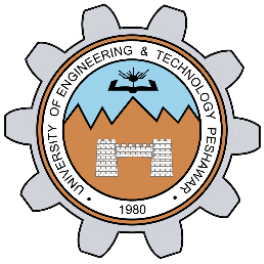
LOW)





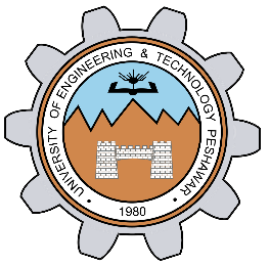
```
#include <msp430fr4133.h>
```

```
int main(void) {  
    WDTCTL = WDTPW | WDTHOLD;  
    PM5CTL0 &= ~LOCKLPM5;  
    P1DIR |= 0x01;  
    P1DIR &= ~0x04;  
    P1OUT |= 0x04;  
    P1REN |= 0x04;  
  
    while (1) {  
        if (!(P1IN & 0x04)) {  
            P1OUT ^= 0x01;  
            __delay_cycles(100000);  
        }  
    }  
}
```

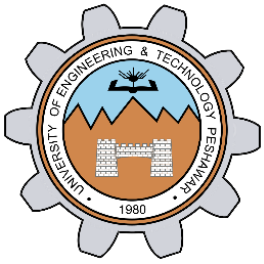


```
#include <msp430fr4133.h>
```

```
int main(void) {  
    WDTCTL = WDTPW | WDTHOLD;  
    PM5CTL0 &= ~LOCKLPM5;  
    P1DIR |= 0x01;  
    P1DIR &= ~0x04;  
    P1OUT |= 0x04;  
    P1REN |= 0x04;  
  
    while (1) {  
        if ((P1IN & 0x04)) {  
            P1OUT ^= 0x01;  
            __delay_cycles(100000);  
        }  
        if (!(P1IN & 0x04)) {  
            P1OUT = 0x00;  
            __delay_cycles(100000);  
        }  
    }  
}
```



```
#include <msp430fr4133.h>
int main(void) {
    WDTCTL = WDTPW | WDTHOLD;
    PM5CTL0 &= ~LOCKLPM5;
    P8DIR |= 0x0F;
    P1DIR &= ~0x04;
    P8OUT |= 0x00;
    P1REN |= 0x04;
    unsigned char pattern = 0x01;
    while (1) {
        if (!(P1IN & 0x04)) {
            __delay_cycles(100000);
            P8OUT = pattern;
            __delay_cycles(100000);
            pattern <<=1;
            if(pattern == 0x10)
            {
                pattern = 0x01;
            }
        }
    }
}
```

TASKS:

Write a program which monitor a switch if it is pressed then toggle LED attached with P1.0.

Write a program which monitor a switch if it is not pressed then toggle LED attached with P1.0 if It is pressed then stopped toggling.

Write a program which monitor a switch, If the switch is pressed the LED if ON should turn OFF if OFF it should turn ON.

Write a program which monitor a switch if it is not pressed then toggle LED attached with P1.0 if It is pressed then stopped toggling.