



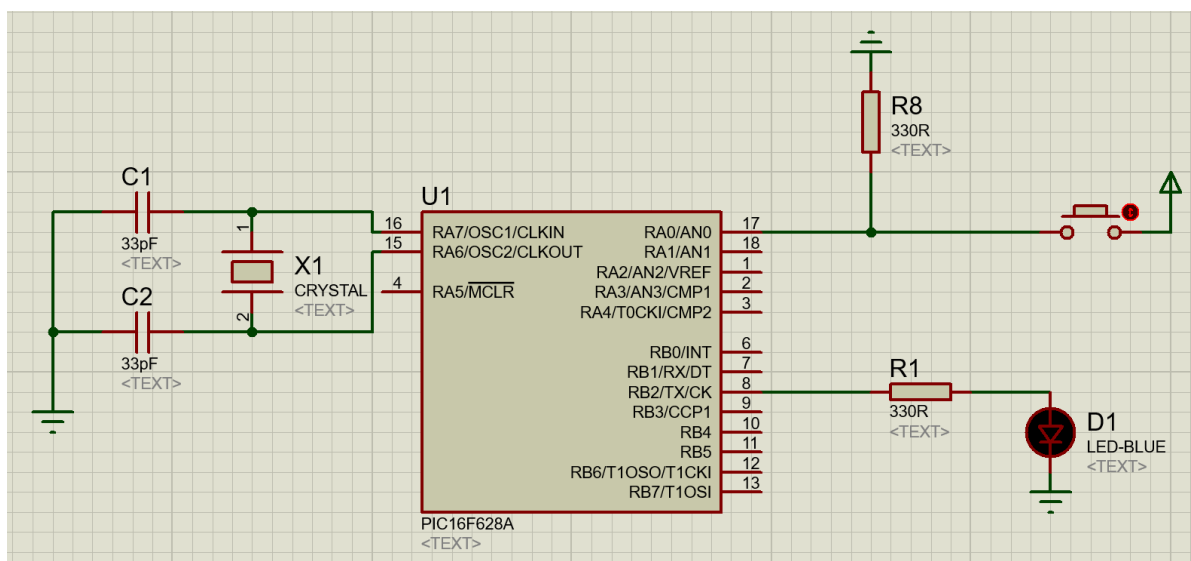
Microcontroladores e DSPs – AP2

Questão 1. Projete um circuito no Proteus e o código em C utilizando um Microcontrolador Pic de sua escolha que apresente:

- Um contador que incremente sua contagem utilizando um botão.
- Apresente esta contagem em piscadas do LED, por exemplo, se o contador for 4, pisque um LED 4 vezes.

Materiais utilizados:

- PIC16F628A
- LED
- Botão
- 2 Resistores 330R
- VCC 5V
- Crystal, capacitores.



- Código realizado no CCS C Compiler:

```
#include <16f628a.h> // Declaração de qual microcontrolador iremos utilizar no projeto
```

```
#use delay(clock=400000) // Configura o atraso de tempo padrão para 400000 ciclos de clock, o que geralmente corresponde a 1 milissegundo.
```

```
#fuses INTRC_IO,NOWDT,PUT,BROWNOUT,NOLVP,NOMCLR // Define as configurações dos fusíveis do microcontrolador, como o uso de clock interno, desativação do timer de watchdog, etc.
```

```
void main(){
```



// Inicialização

```
int contador = 0;

while(1){ // Define o loop infinito, para o programa continuar rodando

if (input(pin_a0) == 1) { // Verifica se o botão conectado ao pino A0 está pressionado.

if (input(pin_a0) == 0); // Espera o botão ser solto (nível baixo)

contador++; // Incrementa o contador

delay_ms(5000); // Cria um delay de 5 segundos

for (unsigned char i = 0; i < contador; i++) { // for que executa a parte de piscar o LED de acordo com o contador

    output_high(pin_b2); // Manda um nível alto no pino B2 que é o local que o LED está conectado

    delay_ms(1000); // Cada "piscada" no LED dura 1 segundo

    output_low(pin_b2); // Nível baixo no pino B2 (apaga o LED)

    delay_ms(1000); // Intervalo de 1 segundo

}
```




// Dois loops para contar de 0 a 99. A cada iteração, os segmentos correspondentes aos dígitos das dezenas e unidades são enviados para os pinos de saída C e D

```
for(i=0;i<10;i++){
```

```
    for(j=0;j<10;j++){
```

```
        output_c(segmentos[numero/10]); // Esta operação de divisão inteira por 10 extrai o
        dígito das dezenas do número armazenado na variável numero. Por exemplo, se numero for 32,
        numero/10 resultará em 3, pois estamos interessados no dígito das dezenas.
```

```
        output_d(segmentos[numero%10]); // O operador "%" retorna o resto da divisão inteira
        de numero por 10. Isso resulta no dígito das unidades do número. Por exemplo, se numero for
        32, numero%10 resultará em 2, pois estamos interessados no dígito das unidades.
```

```
        numero++; // a variável "numero" é incrementada a cada iteração e representa o número
        que será exibido
```

```
        //reinicia a contagem quando chega a 99
```

```
        if(numero==100){
```

```
            numero=0; // Variável número volta a valer 0
```

```
        } delay_ms(100); // Delay para recomeçar a contagem
```

```
    }
```

```
}
```

```
}
```

```
}
```