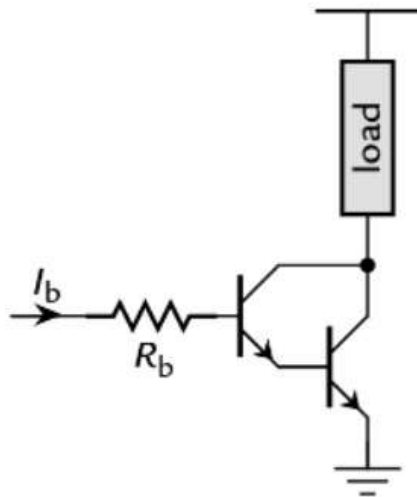


1. Projete o hardware necessário para o MSP430 controlar um motor DC de 12V e 4A. Utilize transistores bipolares de junção (TBJ) com $V_{be} = 0,7 \text{ V}$, $\beta = 100$ e $V_{ce}(\text{saturação}) = 0,2 \text{ V}$. Além disso, considere que $V_{cc} = 3 \text{ V}$ para o MSP430, e que este não pode fornecer mais do que 10 mA por porta digital.

2. Projete o hardware necessário para o MSP430 controlar um motor DC de 10V e 1A. Utilize transistores bipolares de junção (TBJ) com $V_{be} = 0,7 \text{ V}$ e $\beta = 120$. Além disso, considere que $V_{cc} = 3,5 \text{ V}$ para o MSP430, e que este não pode fornecer mais do que 10 mA por porta digital.



Questão 1-

$$I_c = 4 \text{ A}$$

$$I_b = 4 / 100 = 0.04 \text{ mA}$$

$$R_b = (3 - 0.7) / 0.04 \text{ mA} = 57.5 \text{ k}\Omega$$

Questão 2-

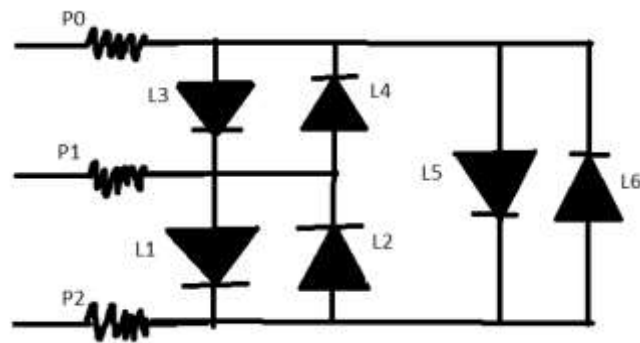
$$I_c = 1 \text{ A}$$

$$I_b = 1 / 120 = 0.0083 \text{ mA}$$

$$R_b = (3.5 - 0.7) / 0.0083 \text{ mA} = 337.5 \text{ }\Omega$$

3. Projete o hardware utilizado para controlar 6 LEDs utilizando charlieplexing. Apresente os pinos utilizados no MSP430 e os LEDs, nomeados L1-L6

Numero de leds = Pinos(pinos - 1)



4. Defina a função void main(void){ } para controlar 6 LEDs de uma árvore de natal usando o hardware da questão anterior. Acenda os LEDs de forma que um ser humano veja todos acesos ao mesmo tempo.

```
#include <msp430g2553.h>
#define BTN BIT2
int main(void) {
    WDTCTL = WDTPW | WDTHOLD;
    P1DIR = 0x0F;
    P1OUT = 0;
    for (;;) {

        P1OUT = 0x01;
        P1OUT = 0x02;
        P1OUT = 0x03;
        P1OUT = 0x04;
        P1OUT = 0x05;
        P1OUT = 0x06;
    }
}
```

5. Defina a função void main(void){ } para controlar 6 LEDs de uma árvore de natal usando o hardware da questão 3. Acenda os LEDs de forma que um ser humano veja os LEDs L1 e L2 acesos juntos por um tempo, depois os LEDs L3 e L4 juntos, e depois os LEDs L5 e L6 juntos.

```
#include <msp430g2553.h>
#define BTN BIT2
int main(void) {
    WDTCTL = WDTPW | WDTHOLD;
    int i;
    P1DIR = 0x0F;
    P1OUT = 0;
    for (;;) {
        for(i=0;i<0xFFFF;i++){
            P1OUT = 0x01;
            P1OUT = 0x02;
        }
        for(i=0;i<0xFFFF;i++){
            P1OUT = 0x03;
            P1OUT = 0x04;
        }
        for(i=0;i<0xFFFF;i++){
            P1OUT = 0x05;
            P1OUT = 0x06;
        }
    }
}
```

6. Defina a função void EscreveDigito(volatile char dig); que escreve um dos dígitos 0x0-0xF em um único display de 7 segmentos via porta P1, baseado na figura abaixo. Considere que em outra parte do código os pinos P1.0-P1.6 já foram configurados para corresponderem aos LEDs A-G, e que estes LEDs possuem resistores externos para limitar a corrente

```
void EscreveDigito(volatile char dig){
```

```
    if (dig == '0') {
        P1OUT = 0x7E;
    }
    if (dig == '1') {
        P1OUT = 0x30;
    }
    if (dig == '2') {
        P1OUT = 0x6D;
    }
    if (dig == '3') {
        P1OUT = 0x79;
    }
    if (dig == '4') {
        P1OUT = 0x33;
    }
    if (dig == '5') {
        P1OUT = 0x5B;
    }
    if (dig == '6') {
        P1OUT = 0x5F;
    }
    if (dig == '7') {
        P1OUT = 0x70;
    }
    if (dig == '8') {
        P1OUT = 0x7F;
    }
    if (dig == '9') {
        P1OUT = 0x73;
    }
    if (dig == 'A') {
        P1OUT = 0x77;
    }
    if (dig == 'B') {
        P1OUT = 0x1F;
    }
    if (dig == 'C') {
        P1OUT = 0x4E;
    }
    if (dig == 'D') {
        P1OUT = 0x3D;
    }
    if (dig == 'E') {
        P1OUT = 0x4F;
    }
    if (dig == 'F') {
        P1OUT = 0x47;
    }
}
```

7. Multiplexe 2 displays de 7 segmentos para apresentar a seguinte sequência em loop
00 - 11 - 22 - 33 - 44 - 55 - 66 - 77 - 88 - 99 - AA - BB - CC - DD - EE - FF

Utilizando a função do exercício anterior
int main(void) {

```
    WDTCTL = WDTPW | WDTHOLD;
    char j;
    int i;
    P1DIR = 0xFF;
    for (;;) {
        j = '0';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '1';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '2';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '3';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '4';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '5';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '6';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '7';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '8';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = '9';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = 'A';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = 'B';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = 'C';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = 'D';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = 'E';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
        j = 'F';
        EscreveDigito(j);
        for(i=0;i<0x1000;i++);
    }
}
```

