



Universidade Estadual do Maranhão
Centro de Ciências Tecnológicas
Curso de Engenharia da Computação e Sistemas
Disciplina: Sistemas de Tempo Real
Professor: David Silva e Silva
Aluno: wesleson souza silva

Atividade Avaliativa Módulo II

Instruções:

Esta atividade consiste em implementar em código o algoritmo de executivo cíclico utilizando a plataforma (Thinkercad¹). Esta plataforma disponibiliza um módulo simulador para a construção de projetos utilizando Arduino.

- 1) Monte uma solução de escalonamento do tipo executivo cíclico composto pelas tarefas abaixo. O circuito deve conter 1 LED para cada tarefa, a atividade executada pela tarefa deverá ser um Blink (acender e apagar o LED). Estas tarefas devem ser escalonadas com base nos tempos descritos abaixo (em milissegundos):

t_1 : C1=4 P1=10 D1=10

t_2 : C2=4 P2=15 D2=15

t_3 : C3=5 P3=30 D3=30

```
void setup()
{
    //Cada pino associado a uma tarefa
    pinMode(13, OUTPUT); //Tarefa 1
    pinMode(12, OUTPUT); //Tarefa 2
    pinMode(11, OUTPUT); //Tarefa 3
}
//Associando pinos as respectivas tarefas
int tarefa1=13, tarefa2=12, tarefa3=11;

//Definindo tempos de execução de cada tarefa
int C1=4000, C2=4000, C3=5000;
//Definindo periodo de cada ciclo menor
int p_cicloMenor=5000;
//Todos os tempos estão em segundos para melhor visualização

//Tarefas acionam o seu led durante o seu tempo de execução
//no ciclo menor
void cicloMenor(int pino, int p_cicloMenor, int C){
    digitalWrite(pino, HIGH);
    delay(C);
    digitalWrite(pino, LOW);
    delay(p_cicloMenor-C);
}

void loop()
{
    cicloMenor(tarefa1, p_cicloMenor, C1);
```

```
//Ciclo menor de 5 ms
```

```
cicloMenor(tarefa2, p_cicloMenor, C2);
```

```
cicloMenor(tarefa1, p_cicloMenor, C1);
```

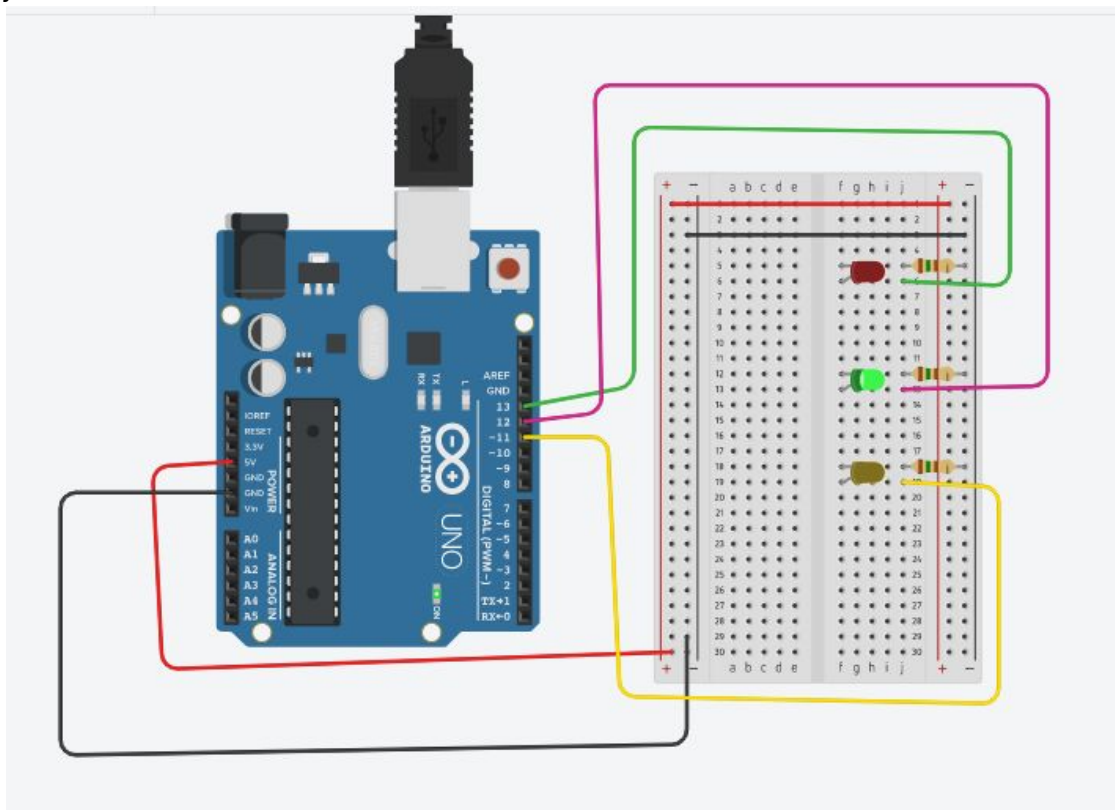
```
cicloMenor(tarefa3, p_cicloMenor, C3);
```

```
cicloMenor(tarefa1, p_cicloMenor, C1);
```

```
cicloMenor(tarefa2, p_cicloMenor, C2);
```

```
//Ciclo maior de 30 ms
```

```
}
```



2) Faça o mesmo para o seguinte sistema com quatro tarefas:

t_1 : $C_1=1$ $P_1=5$ $D_1=5$

t_2 : $C_2=2$ $P_2=5$ $D_2=5$

t_3 : $C_3=2$ $P_3=10$ $D_3=10$

t_4 : $C_4=2$ $P_4=15$ $D_4=15$

```
void setup()
```

```
{
```

```
//Cada pino associado a uma tarefa
```

```
pinMode(13, OUTPUT); //Tarefa 1
```

```
pinMode(12, OUTPUT); //Tarefa 2
```

```
pinMode(11, OUTPUT); //Tarefa 3
```

```
pinMode(10, OUTPUT); //Tarefa 4
```

```

}
//Associando pinos as respectivas tarefas
int tarefa1=13,tarefa2=12,tarefa3=11, tarefa4=10;

//Definindo tempos de execução de cada tarefa
int C1=1000,C2=2000, C3=2000, C4=2000;

//Definindo periodo de cada ciclo menor
int p_cicloMenor=5000;
//Todos os tempos estão em segundos para melhor visualização

//Tarefas acionam o seu led durante o seu tempo de execução
//no ciclo menor
void cicloMenor(int *pino, int p_cicloMenor, int *C, int n){
    int C_Total=0;
    for(int i=0; i<n;i++){
        digitalWrite(pino[i], HIGH);
        C_Total = C_Total+C[i];
        delay(C[i]);
        digitalWrite(pino[i], LOW);
    }
    delay(p_cicloMenor-C_Total);
}

//Vetores que armazenam as tarefas e seus respectivos
//tempos de execução em cada ciclo menor
int tarefas[3], C[3];

void loop()
{
    tarefas[0]=tarefa1, tarefas[1]=tarefa2, tarefas[2]=tarefa3;
    C[0]=C1, C[1]=C2,C[2]=C3;
    cicloMenor(tarefas, p_cicloMenor, C, 3);
    //Ciclo menor de 5 ms

    tarefas[0]=tarefa1, tarefas[1]=tarefa2, tarefas[2]=tarefa4;
    C[0]=C1, C[1]=C2, C[2]=C4;
    cicloMenor(tarefas, p_cicloMenor, C, 3);

    tarefas[0]=tarefa1, tarefas[1]=tarefa2, tarefas[2]=tarefa3;
    C[0]=C1, C[1]=C2, C[2]=C3;
    cicloMenor(tarefas, p_cicloMenor, C, 3);

    tarefas[0]=tarefa1, tarefas[1]=tarefa2;
    C[0]=C1, C[1]=C2;
    cicloMenor(tarefas, p_cicloMenor, C, 2);

    tarefas[0]=tarefa1, tarefas[1]=tarefa2, tarefas[2]=tarefa3;
    C[0]=C1, C[1]=C2, C[2]=C3;
    cicloMenor(tarefas, p_cicloMenor, C, 3);
}

```

```

tarefas[0]=tarefa1, tarefas[1]=tarefa2, tarefas[2]=tarefa4;
C[0]=C1, C[1]=C2, C[2]=C4;
cicloMenor(tarefas, p_cicloMenor, C, 3);

```

```

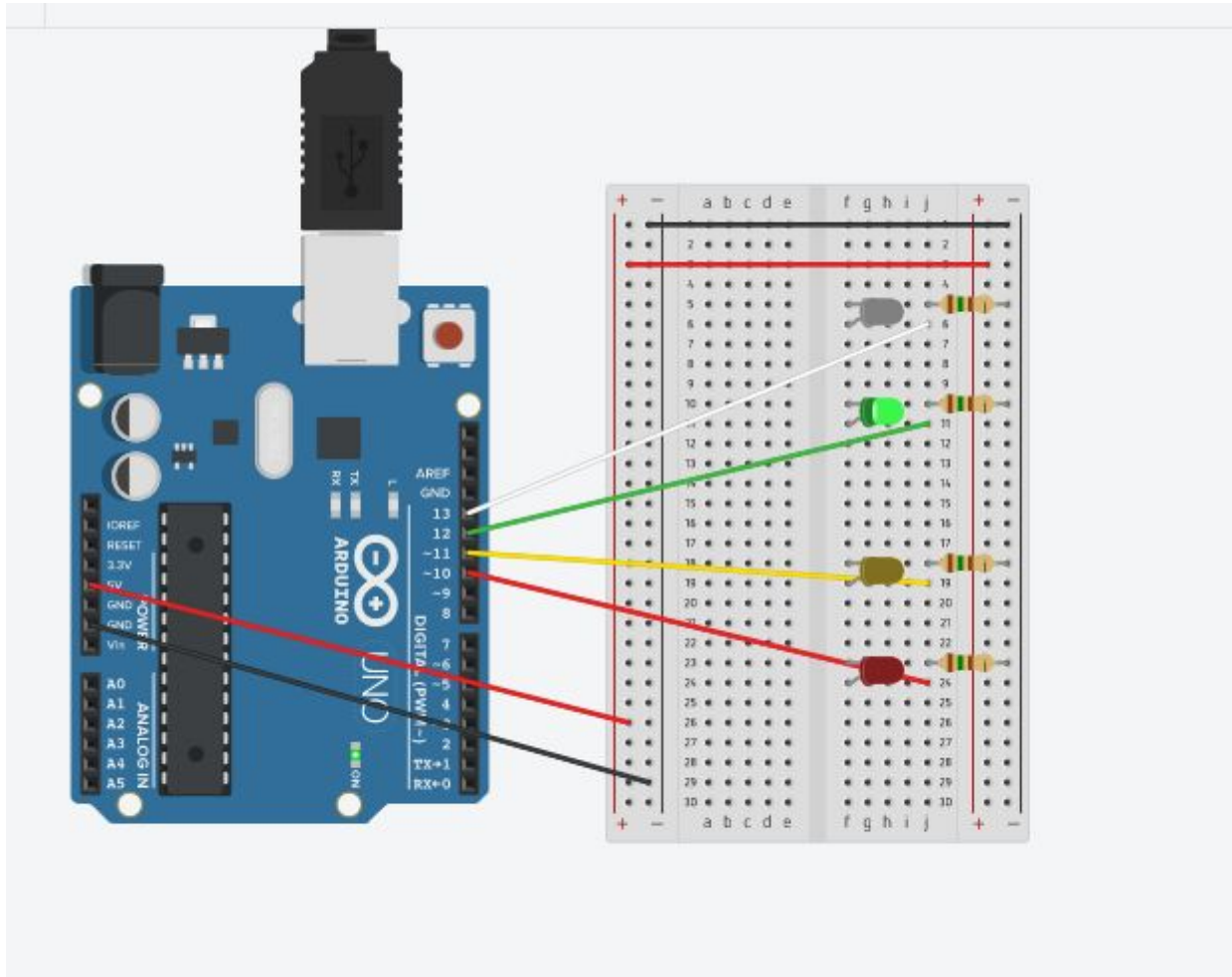
//Ciclo maior de 30 ms

```

```

}

```



3) Faça o mesmo para o seguinte Sistema com quatro tarefas:

$t1$: $C1=2$ $P1=8$ $D1=8$

$t2$: $C2=4$ $P2=16$ $D2=16$

$t3$: $C3=4$ $P3=24$ $D3=24$

$t4$: $C4=6$ $P4=48$ $D4=48$

```

void setup()

```

```

{

```

```

    //Cada pino associado a uma tarefa

```

```

    pinMode(13, OUTPUT); //Tarefa 1

```

```

    pinMode(12, OUTPUT); //Tarefa 2

```

```

    pinMode(11, OUTPUT); //Tarefa 3

```

```

    pinMode(10, OUTPUT); //Tarefa 4

```

```

}

```

```

//Associando pinos as respectivas tarefas

```

```

int tarefa1=13,tarefa2=12,tarefa3=11, tarefa4=10;

```

```
//Definindo tempos de execução de cada tarefa  
int C1=2000,C2=4000, C3=4000, C4=6000;
```

```
//Definindo periodo de cada ciclo menor  
int p_cicloMenor=8000;
```

```
//Todos os tempos estão em segundos para melhor visualização
```

```
//Tarefas acionam o seu led durante o seu tempo de execução
```

```
//no ciclo menor
```

```
void cicloMenor(int *pino, int p_cicloMenor, int *C, int n){  
    int C_Total=0;  
    for(int i=0; i<n;i++){  
        digitalWrite(pino[i], HIGH);  
        C_Total = C_Total+C[i];  
        delay(C[i]);  
        digitalWrite(pino[i], LOW);  
    }  
    delay(p_cicloMenor-C_Total);  
}
```

```
//Vetores que armazenam as tarefas e seus respectivos
```

```
//tempos de execução em cada ciclo menor
```

```
int tarefas[2], C[2];
```

```
void loop()
```

```
{  
    tarefas[0]=tarefa1, tarefas[1]=tarefa2;  
    C[0]=C1, C[1]=C2;  
    cicloMenor(tarefas, p_cicloMenor, C, 2);  
    //Ciclo menor de 8 ms
```

```
    tarefas[0]=tarefa1, tarefas[1]=tarefa3;  
    C[0]=C1, C[1]=C3;  
    cicloMenor(tarefas, p_cicloMenor, C, 2);
```

```
    tarefas[0]=tarefa1, tarefas[1]=tarefa2;  
    C[0]=C1, C[1]=C2;  
    cicloMenor(tarefas, p_cicloMenor, C, 2);
```

```
    tarefas[0]=tarefa1, tarefas[1]=tarefa4;  
    C[0]=C1, C[1]=C4;  
    cicloMenor(tarefas, p_cicloMenor, C, 2);
```

```
    tarefas[0]=tarefa1, tarefas[1]=tarefa2;  
    C[0]=C1, C[1]=C2;  
    cicloMenor(tarefas, p_cicloMenor, C, 2);
```

```
    tarefas[0]=tarefa1, tarefas[1]=tarefa3;
```

```
C[0]=C1, C[1]=C3;  
cicloMenor(tarefas, p_cicloMenor, C, 2);
```

```
//Ciclo maior de 48 ms
```

```
}
```

