

### Atividade de laboratório – Utilização das portas digitais do Arduino

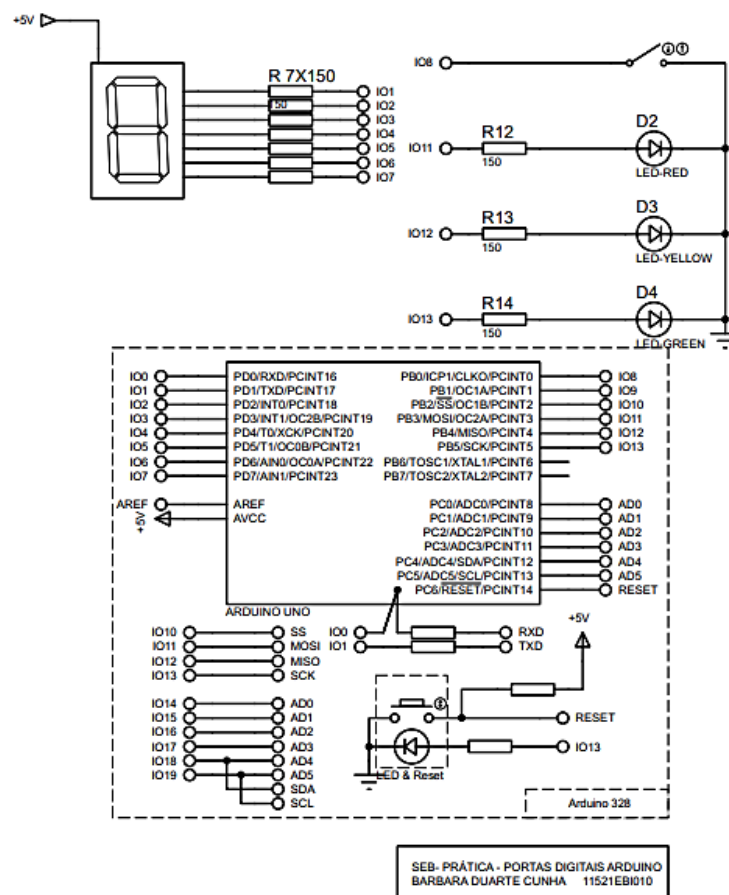
Esta experiência tem a finalidade de mostrar o funcionamento das portas digitais (entrada e saída) do Arduino. O estudo das portas será realizado através da implementação de um semáforo.

#### Material:

- Arduino UNO;
- 10 resistores de 150Ω;
- 3 LEDs sendo: Verde, Vermelho e Amarelo;
- 1 display 7 segmentos catodo-comum;

#### Procedimentos / Questões:

- ✓ Montar o circuito do roteiro no Proteus;



- ✓ Consultar nas referências do Arduino o funcionamento dos comandos *pinMode*, *digitalRead*, *digitalWrite*, *delay*.

***pinMode()*** : Configura o pino especificado para funcionar como uma entrada ou saída.

Sintaxe: *pinMode*(pino, modo)

Parâmetros: pino é o numero do pino do Arduino no qual se quer configurar o modo. Modo é o modo do pino, “estado”, pode ser *INPUT* (entrada), *OUTPUT* (saída) ou *INPUT\_PULLUP* (saída e entrada com *pull-up* ativado).

***digitalRead()***: Lê o valor de um pino digital especificado, *HIGH* (nível alto) ou *LOW* (nível baixo).

Sintaxe: *digitalRead*(pino)

Parâmetros: pino é o número pin do Arduino que você deseja ler.

***digitalWrite()***: Escreva um *HIGH* ou um *LOW* valor em um pino digital. Se o pino tiver sido configurado como *OUTPUT* com *pinMode()*, sua tensão será configurada para o valor correspondente: 5 V (ou 3,3 V em placas de 3,3 V) para *HIGH*, 0 V (terra) para *LOW*. Se o pino estiver configurado como *INPUT*, *digitalWrite()* irá habilitar (*HIGH*) ou desabilitar (*LOW*) o pullup interno no pino de entrada.

Se você não definir *pinMode()* como *OUTPUT* e conectar um *LED* a um pino, ao chamar *digitalWrite* (*HIGH*), o *LED* pode parecer fraco. Sem definir explicitamente *pinMode()*, *digitalWrite()* terá habilitado o resistor *pull-up* interno, que atua como um grande resistor limitador de corrente.

***Delay()***: Pausa o programa pelo período de tempo (em milissegundos) especificado como parâmetro. (Existem 1000 milissegundos em um segundo.)

Sintaxe: *delay*(ms)

Parâmetros: ms é o número de milissegundos para pausar.

- ✓ Faça um programa no Arduino para acionar repetidamente os LEDs, seguindo a sequência de um semáforo de trânsito e obedecendo os seguintes tempos: a. Vermelho: 3 segs. b. Amarelo: 1 seg. c. Verde: 2 segs.
- ✓ Monitore a chave Amarelo Intermitente de modo que quando ela estiver fechada, o semáforo piscará de forma intermitente o LED amarelo com tempos de 0,5 segundo apagado e 0,5 segundo aceso. Quando a chave estiver aberta o semáforo deverá funcionar normalmente. 6. Desenvolva outro software para acionar o display, de modo que no mesmo seja mostrado os números de 0 a 9 repetidamente, com intervalo de 300ms.

```
const int LED_VERMELHO = 11;
```

```
const int LED_AMARELO = 12; // atribuindo variável do tipo inteira e inicializando nas portas digitais do arduino.
```

```
const int LED_VERDE = 13;
```

```
const int chave = 8;
```

```
void setup() { //A void setup() é uma função que é executada apenas quando se inicia o programa e serve para configurar os pinos da placa e estabelecer a comunicação serial com um computador.
```

```
  pinMode(chave, INPUT_PULLUP); //configurar um pino como sendo saída ou entrada //INPUT_PULLUP: saída e entrada com pull-up ativado.
```

```
  pinMode(LED_VERMELHO, OUTPUT);
```

```
  pinMode(LED_AMARELO, OUTPUT);
```

```
  pinMode(LED_VERDE, OUTPUT);
```

```
}
```

```
void loop() { //é a função que executa os comandos.
```

```
  if(digitalRead(chave)){ // serve para ler em um pino digital 0 ou 1(0V ou 5V).
```

```
    digitalWrite(LED_VERMELHO, HIGH); //serve para escrever em um pino digital 0 ou 1(0V ou 5V).
```

```
    delay(3000); // aguarda 3s
```

```
    digitalWrite(LED_VERMELHO, LOW);
```

```
    digitalWrite(LED_AMARELO, HIGH);
```

```
    delay(1000); aguarda 1s
```

```
    digitalWrite(LED_AMARELO, LOW);
```

```
    digitalWrite(LED_VERDE, HIGH);
```

```
    delay(2000);aguarda 2s
```

```
    digitalWrite(LED_VERDE, LOW);
```

```
}
```

```
  else{
```

```
digitalWrite(LED_VERMELHO, LOW);  
digitalWrite(LED_VERDE, LOW);
```

```
digitalWrite(LED_AMARELO, HIGH);  
delay(500);  
digitalWrite(LED_AMARELO, LOW);  
delay(500);
```

```
}
```

```
}
```

✓ Padrão de resposta esperado:

- a) Gerar um arquivo pdf com o desenho do esquema eletrônico;
- b) Gerar um arquivo pdf com o software desenvolvido na IDE do Arduino;
- c) Preparar um vídeo de até 5min explicando o esquema eletrônico, o software desenvolvido e o funcionamento de cada função utilizada na implementação do programa.