

ADA

PROJETOS EM ENGENHARIA
DE COMPUTAÇÃO

Arduino Intermediário





EEPROM

- Memória não volátil
- Tempo de vida útil reduzido
- Acesso mais lento

Aplicações?



EEPROM

- Como utilizar?

```
#include <EEPROM.h>

void setup() {
  Serial.begin(9600) ;
  int val = 1234 ;
  int addr = 0 ;
  EEPROM.put(addr, val) ;
  Serial.println("Dados gravados!") ;
}

void loop() {
}
```

```
#include <EEPROM.h>

void setup() {
  Serial.begin(9600) ;
  int val = 0 ;
  int addr = 0 ;
  EEPROM.get(addr, val) ;
  Serial.println(val) ;
}

void loop() {
}
```

Exercício

- Gravem um inteiro na EEPROM em um endereço qualquer
- Troquem de arduino entre as duplas e passem o endereço que foi gravado
- Façam um programa que leia o endereço e confirme com a dupla da troca se o número era este



Bibliotecas

- A comunidade do Arduino já criou muitas e muitas bibliotecas
- Instalar uma biblioteca é bem fácil
 - Sketch > Include Library > Manage Libraries para bibliotecas oficiais
 - Basta colocar o .zip da biblioteca na pasta libraries dentro do diretório do Arduino e ir em Sketch > Include library > Add .ZIP library para bibliotecas não oficiais

Bibliotecas

- Como é muito fácil instalar um biblioteca vamos aprender a fazer a nossa
 - É necessário escrevermos um arquivo .h e um .cpp
 - Também podemos escrever um arquivo chamado keywords.txt que diz para a IDE do arduino quais são as palavras reservadas da nossa biblioteca



Bibliotecas

Teste.cpp

```
#include "Teste.h"  
// include core Wiring API  
#include "Arduino.h"  
// include description files for other libraries  
#include "HardwareSerial.h"
```

```
Teste::Teste(int a) {  
    val = a ;  
    Serial.begin(9600) ;  
}
```

```
void Teste::hello() {  
    Serial.print("Hello World ") ;  
    Serial.println(val) ;  
    secret() ;  
}
```

```
void Teste::secret() {  
    pinMode(13, OUTPUT) ;  
    digitalWrite(13, HIGH) ;  
}
```

Teste.h

```
#include "Arduino.h"
```

```
class Teste {  
    public:  
        Teste(int a) ;  
        void hello() ;  
  
    private:  
        int val ;  
        void secret() ;  
};
```

keywords.txt

```
#####  
# Syntax Coloring Map For Test  
#####  
  
#####  
# Datatypes (KEYWORD1)  
#####  
  
Teste    KEYWORD1  
  
#####  
# Methods and Functions (KEYWORD2)  
#####  
  
hello    KEYWORD2  
  
#####  
# Instances (KEYWORD2)  
#####  
  
#####  
# Constants (LITERAL1)  
#####
```

Bibliotecas

- Crie um .zip com estes três arquivos
- Sketch > Include library > Add .ZIP library
- Selecione o seu .zip
- Sketch > Include library
- Selecione a sua biblioteca da lista



Bibliotecas

- Já pode usar a sua biblioteca!!

```
#include <Teste.h>

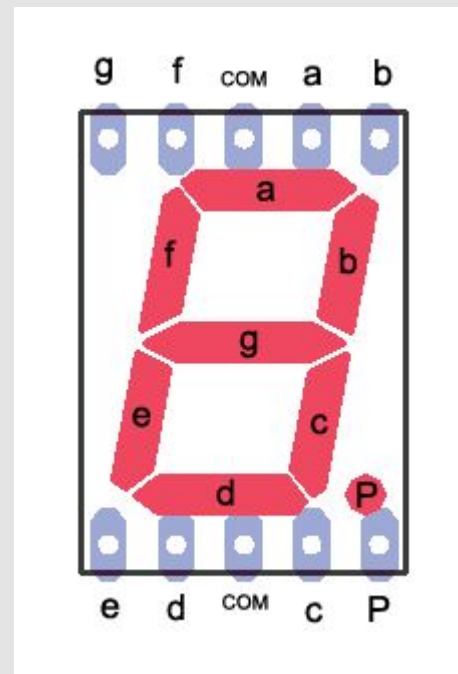
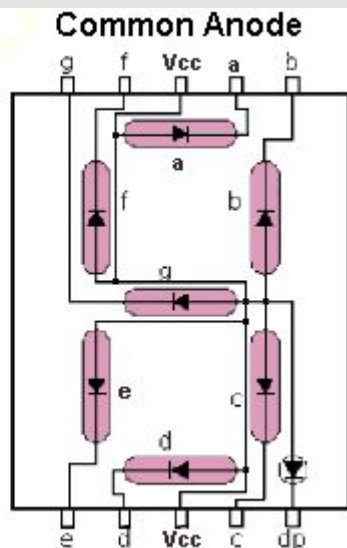
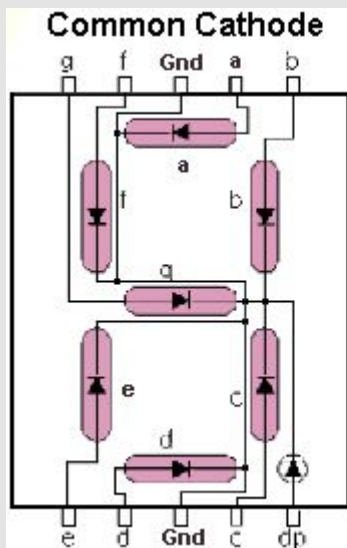
Teste t = Teste(5);
void setup() {
    t.hello() ;
}

void loop() {
}
```



Display de 7segmentos

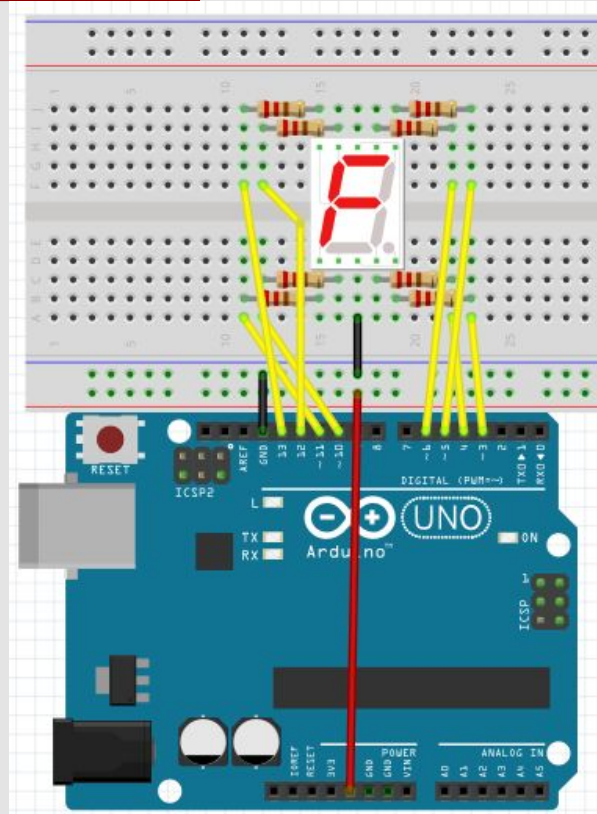
- Mais uma forma de mostrar informações
- Composto por 7 LEDs





Display de 7segmentos

- O oitavo LED é o ponto
- Liguem o terra no pino do central, tanto em cima, quanto em baixo





Display de 7segmentos

Segments (✓ = ON)							Display	Segments (✓ = ON)							Display
a	b	c	d	e	f	g		a	b	c	d	e	f	g	
✓	✓	✓	✓	✓	✓		0	✓	✓	✓	✓	✓	✓	✓	8
	✓	✓					1	✓	✓	✓			✓	✓	9
✓	✓		✓	✓		✓	2	✓	✓	✓		✓	✓	✓	A
✓	✓	✓	✓			✓	3			✓	✓	✓	✓	✓	b
	✓	✓			✓	✓	4	✓			✓	✓	✓		c
✓		✓	✓		✓	✓	5		✓	✓	✓	✓		✓	d
✓		✓	✓	✓	✓	✓	6	✓			✓	✓	✓	✓	E
✓	✓	✓					7	✓				✓	✓	✓	F

Exercício 5

- Escrevam no display os número entre 0 e 9
- Mudando o número a cada um segundo

Desafio 5

- Façam uma biblioteca para controlar o display de 7 segmentos
- O construtor da classe deve receber os pinos em que estão conectados os segmentos
- A classe deve ter uma função que dado um número inteiro entre 0 e 9 imprima este número no display



Random

- `randomSeed(int)`
 - Inicializa o gerador de números pseudo aleatórios
 - Pode-se passar por parâmetro a entrada de uma porta analógica que não está sendo utilizada
- `random(max)` ou `random(min, max)`
 - Retorna um número aleatório entre min e max-1
 - Caso não passe min por parâmetro, min = 0

Projeto final

Dado eletrônico

- Ligue o display de 7 segmentos e um botão no Arduino
- Quando o botão for apertado, gere um número aleatório entre 1 e 6
- Apresente o número no display