



Workshop Arduino

Prof. Simão

O que é "Arduino"?



- É uma plataforma para elaboração de protótipos
 - Hardware e software livres
- Foi criado em 2005 por 5 pesquisadores italianos
 - Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis

https://www.arduino.cc/





Learn more about Arduino

Discover the official Arduino boards

Download the Arduino Software



Combinação de *hardware* + *software*

Arquivo Editar Sketch Ferramentas Ajuda



Bli	ink
11/	*
2	Blink
	Turns an LED on for one second, then off for one second, repeatedly.
	······································
	Most Arduinos have an on-board LED vou can control. On the UNO. MEGA and ZERO
	it is attached to digital pin 13, on MKR1000 on pin 6. LED BUILTIN is set to
3	the correct LED pin independent of which board is used.
9	If you want to know what pin the on-board LED is connected to on your Arduino
0	model, check the Technical Specs of your board at:
L	https://www.arduino.cc/en/Main/Products
3	modified 8 May 2014
	by Scott Fitzgerald
5	modified 2 Sep 2016
5	by Arturo Guadalupi
7	modified 8 Sep 2016
3	by Colby Newman
3	
0	This example code is in the public domain.
2	http://www.arduino.cc/en/Tutorial/Blink
3 *	
4	
2/	/ the setup function runs once when you press reset or power the board
v	old setup() {
<u>_</u>	// initialize digitat pin LED_BUILIIN as an output.
5,	pinMode(LED_BUILTIN, OUTPOT);
93	
2	(the loss function must such and such series formula
1/	/ the loop function runs over and over again forever
2 1	digitalWrite/LED_PUTLTIN_UTCU). // turn the LED on (UTCU is the valtere level)
5	digitatwrite(LED_BOILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
1	<pre>decay(1000); // wait for a second digitalWrite(LED_RUTITIN_LOW). // turn the LED off by making the voltage LOW.</pre>
6	delay(1000), // wait for a second
۲'۲	detay (1000), // wait ibi a second
1 A L L L	





Muitas variantes



Programação *online* ou Desktop





Programação *online* ou Desktop



controlePWM Arduino 1.8.5	• •
rquivo <u>E</u> ditar <u>S</u> ketch Ferramen <u>t</u> as Ajuda	
	ø
controlePWM	
1 Designe en ocida De Versida Della contralaz	
∠#define pwmsalda 3 //salda Pwm a controlar	
A void setun()	
Sf	
6 Serial.begin(9600);	
7 Serial.print("Iniciando\n\n");	
8 pinMode(pvmSaida, OUTPUT); //saída PWM a controlar	
9 }	
L void Loop()	
2 int rivelCMM On (/vei de Chie 100%)	
S int nivetram = 0; //vai de 0% a 100%	
$4 = \frac{1}{2} $	
analogWrite(nymSaida_nivelPWM))	
delay (espera):	
9 }	
Serial print("Diminuindo\n"); //Vai de 100% a 0%	
1 for (nivelPWM = 255; nivelPWM >= 50; nivelPWM)	
2 {	
<pre>3 analogWrite(pwmSaida, nivelPWM);</pre>	
4 delay(espera);	
5 }	
ompilação terminada.	
sketch usa 2224 bytes (6%) de espaco de armazenamento para programas. O máximo são 32256 bytes	
sriávej globaj usam 230 hytes (11%) de memória dinâmica, deixando 1818 hytes para variávejs locais. O máximo são 204	8 bytes.
	0 0) 0001



Simulações online





Design online das ligações





0100000

simao@ufpr.br - 2022







Tl Aplicada



Hands on

Ambiente



- Instale o IDE do Arduino
 - Você terá funcionamento off line e liberdade para instalar bibliotecas e placas
- Ou crie uma conta online
 - Não precisa instalar nada, porém haverá algumas limitações de placas a serem resolvidas com versão paga

Em qualquer caso...



- Haverá necessidade de comunicação entre a placa e o computador
 - A conexão é por meio de USB, porém haverá um mapeamento interno para conexão serial
 - Poderá ser necessário ajustar drivers

Como ficará





O ambiente de desenvolvimento estará instalado / configurado no computador.

A placa escolhida estará disponível com cabo de comunicação USB.

O computador deverá reconhecer a comunicação com a placa!



Led aceso = energia



Led aceso na placa é sinal de que a energia está chegando à placa, a conexão USB está OK, mas não garante que a <u>comunicação</u> esteja OK!

Testando



- Abra o IDE do arduino
- Conecte o cabo USB na placa
- Conecte o cabo USB no computador
- Verifique a identificação da placa no IDE
 - Ferramentas / Portas / Portas seriais

0000010 1001011 00

Identificação da placa no IDE



Arquivo <u>E</u> ditar <u>S</u> ketch	Ferramen <u>t</u> as Ajuda Autoformatação	Ctrl+T	
<pre>sketch_sep27a 1 void setup() { 2 // put your setup 3 4 } 5 6 void loop() { 7 // put your main c 8 9 }</pre>	Arquivar Sketch Corrigir codificação e recarregar Monitor serial Plotter serial ArduBlock WiFi101 Firmware Updater Placa: "Arduino/Genuino Uno"	Ctrl+Shift+M Ctrl+Shift+L	
21	Porta: "/dev/ttyUSB0" Obter informações da Placa Programador: "AVRISP mkII" Gravar Bootloader	•	Portas seriais /dev/ttyS4 /dev/ttyS0 ✓ /dev/ttyUSB0

Confirme a placa

Ctrl+T



Ferramentas Ajuda Autoformatação Arguivar Sketch Corrigir codificação e recarregan Monitor serial Ctrl+Shift+M Plotter serial Ctrl+Shift+L ArduBlock WiFi101 Firmware Updater Placa: "Arduino/Genuino Uno" Porta: "/dev/ttyUSB0" Obter informações da Placa Programador: "AVRISP mkll" Gravar Bootloader

Gerenciador de Placas...

Placas Arduino AVR Arduino Yún

 Arduino/Genuino Uno Arduino Duemilanove or Diecimila Arduino Nano Arduino/Genuino Mega or Mega 2560 Arduino Mega ADK Arduino Leonardo Arduino Leonardo ETH Arduino/Genuino Micro

Carregue um exemplo

Arquivo/ Exemplos/ 01.Basics/ Blink

Idnino Edical Skercu i c	Ctrl+N	Exemplos embutidos		
Abrir	Ctrl+O	01.Basics	•	AnalogReadSerial
bricRecente		02.Digital		BareMinimum
ketchbook		03.Analog		Blink
Svemplos		04.Communication		DigitalReadSerial
Exemptos	Ctrl+W	05.Control		Fade
-echar	Ctd+S	06.Sensors		ReadAnalogVoltage
Salvar	CELLShift	07.Display		
Salvar como		08.Strings		
Configuração da página	Ctrl+Shift+P	09.USB		
Imprimir	Ctrl+P	10.StarterKit BasicKit		
Preferências	Ctrl+Vírgula	11.ArduinoISP		
Sair	Ctrl+Q	Exemples para qualquer placa		
		Ridge		
		Ecolora		
		Esplora		
		Ellenie		
		Filliaca		
		Robot Control		
		Robot Motor		



Aolicada

Código do exemplo '*blink*'

Blink Arduino 1.8.5	● 0 😣
Arquivo <u>E</u> ditar <u>S</u> ketch Ferramen <u>t</u> as Ajuda	
	₽
Blink	
<pre>1 /* 2 Blink 3 4 Turns an LED on for one second, then off for one second, repeatedly. 5 6 Most Arduinos have an on-board LED you can control. On the UNO, MEGA and ZERO 7 it is attached to digital pin 13, on MKR1000 on pin 6. LED_BUILTIN is set to 8 the correct LED pin independent of which board is used. 9 If you want to know what pin the on-board LED is connected to on your Arduino 10 model, check the Technical Specs of your board at: 11 https://www.arduino.cc/en/Main/Products 12 13 modified 8 May 2014 14 by Scott Fitzgerald 15 modified 2 Sep 2016 16 by Arturo Guadalupi 17 Mit is a second sec</pre>	<u>×</u>
<pre>by Colby Newman l9 l9 l9 l9 l9 l1 http://www.arduino.cc/en/Tutorial/Blink l3 */ l4 l5 // the setup function runs once when you press reset or power the board l5 // the setup () {</pre>	لے بد بد



Arduino/Genuino Uno em /dev/ttyUSB0

Compilar e transferir o código



- Sem alterar a configuração de porta e placa já testadas:
 - Clique no ícone de compilação para testar o código:

Arquivo	<u>E</u> ditar	<u>S</u> ketch	Ferramen <u>t</u> as	Ajuda
-► <> <>		👱 Ve	erificar	
Blink				
1/*				



Compilar e transferir o código



- Sem alterar a configuração de porta e placa já testadas:
 - Clique no ícone de transferência:

Arquivo	<u>E</u> ditar <u>S</u> ketcl	n Ferramen <u>t</u> as	Ajuda	
- -	🗈 🛨 🕙 o	arregar		
Blink				
1 /*				1.

Nota: durante a transferência de dados os leds *Rx* e *Tx* da placa irão piscar.

Carregado.

O sketch usa 928 bytes (2%) de espaço de armazenamento para programas. O máximo são 32256 bytes. Variáveis globais usam 9 bytes (0%) de memória dinâmica, deixando 2039 bytes para variáveis locais. O máximo são 2048 bytes.

Arduino/Genuino Uno em /dev/ttyUSB0

O led da placa deverá piscar











Tl Aplicada



Parabéns!



Compreendendo e alterando...



- Há duas funções principais no código do Arduino:
 - setup()
 - Seta parâmetros e condições inciais de funcionamento
 - *loop(*)
 - Executa 'para sempre' as rotinas descritas na função

Tl Aplicada

Compreendendo e alterando...



setup()

 No código do exemplo, a função setup() inicializa a porta 13 da placa (LED_BUILTIN, padrão) como sendo uma saída (OUTPUT), usando a função pinMode(pino, direção)

```
25 // the setup function runs once when you press reset or power the board
26 void setup() {
27 // initialize digital pin LED_BUILTIN as an output.
28 pinMode(LED_BUILTIN, OUTPUT);
29 }
```

Tl Aplicada

Compreendendo e alterando...



- *loop(*)
 - No código do exemplo, a função loop() utiliza a função digitalWrite (porta, valor) e a função delay(tempo em milisegundos) para alternar a energia ao LED, que acenderá ou não

```
31 // the loop function runs over and over again forever
32 void loop() {
33 digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
34 delay(1000); // wait for a second
35 digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
36 delay(1000); // wait for a second
37 }
```

Nota: os números de linhas podem ser diferentes.

Compreendendo e alterando...



- Mude o valor do delay()
 - Altere o valor do delay(): tempos menores farão o led piscar mais rápido e vice-versa; troque o intervalo em que o led fica aceso e apagado

```
32 void loop() {
33  digitalWrite(13, HIGH); // liga o led (veja o número da porta...)
34  delay(2000); //espera 2 segundos
35  digitalWrite(LED_BUILTIN, LOW); // desliga o led
36  delay(500); // espera 0,5 segundos
37 }
```

Nota: os números de linhas podem ser diferentes.

simao@ufpr.br - 2022

Atenção



- A linguagem é "case sensitive"
 - Maiúsculas e minúsculas não são a mesma coisa!
- Comentários
 - Servem para documentação e auxílio
 - No código, um "//" transforma em comentário até o final da linha
 - Para um bloco de comentários iniciamos com "/*" e terminamos com "*/"



O que você aprendeu, que será necessário para os próximos passos:

1. Usar o IDE para registrar a conexão serial

- 2. Usar o IDE para registrar a placa Arduino
- 3. Compilar um código fornecido
- 4. Enviar o código compilado para o Arduino