



# Arduino <sup>Para</sup> leigos

O Arduino pode ajudá-lo a construir robôs ou dispositivos eletrônicos. Porém, há muito a se aprender sobre o Arduino, pois ele contempla os mundos de hardware e software. As seções a seguir apresentam informações a respeito do uso de resistores e ferramentas.

## USANDO RESISTORES NO ARDUINO

Ao criar seus projetos Arduino, você usa os resistores para restringir a quantidade de corrente que percorre determinados componentes no circuito, como LEDs e circuitos integrados. Para calcular a resistência, você deve usar uma versão modificada da Lei de Ohm. Na equação a seguir,  $R$  é a resistência;  $V_{\text{SUPPLY}}$  é a tensão fornecida pela fonte de alimentação (isto é, 5V para um pino digital padrão do Arduino, mas poderia ser mais ou menos isso se o pino  $V_{\text{IN}}$  for usado);  $V_{\text{FORWARD}}$  é a tensão exigida pelo componente; e  $I$  é a corrente requerida pelo componente:

$$R = (V_{\text{SUPPLY}} - V_{\text{FORWARD}}) / I$$

Veja um exemplo que faz um LED piscar:

$$(5V - 2V) / 0,03A = 100\Omega$$

Depois de identificar de qual resistor você precisa, a próxima tarefa é encontrá-lo. Os resistores de valor fixo usam faixas coloridas para indicar seu valor. Para encontrar o valor, você pode utilizar um multímetro definido em ohms ( $\Omega$ ), mas, caso você não tenha um multímetro, use a tabela a seguir para descobri-lo. Por exemplo, um resistor com faixas marrom, preta, marrom e dourada é um resistor de  $100\Omega$  com uma tolerância de 5%. Além do mais, o **Circuit Playground** da Adafruit tem os códigos de cores dos resistores e uma diversidade de ferramentas para a construção de circuitos.

### Tabela de Cores do Resistor

Cor	Valor	Multiplicador	Tolerância
Preto	0	x1	-
Marrom	1	x10	$\pm 1\%$
Vermelho	2	x100	$\pm 2\%$
Laranja	3	x1.000	-
Amarelo	4	x10.000	-
Verde	5	x100.000	$\pm 0,5\%$
Azul	6	x1.000.000	$\pm 0,25\%$
Violeta	7	x10.000.000	$\pm 0,1\%$
Cinza	8	x100.000.000	$\pm 0,05\%$

(continua)



# Arduino <sup>Para</sup> leigos

(continuação)

Cor	Valor	Multiplicador	Tolerância
Branco	9	x1.000.000.000	-
Dourado	-	x0,1	±5%
Prateado	-	x0,01	±10%
Nenhuma	-	-	±20%

## FERRAMENTAS PARA SEU PROJETO ARDUINO

Ao começar a construir seu projeto Arduino, é difícil saber quais ferramentas comprar. Veja a seguir as ferramentas recomendadas para prototipagem com ou sem soldagem. Nota: compre as ferramentas, conforme necessário. Você nunca sabe do que pode precisar e as ferramentas podem ser caras.

### FERRAMENTAS PARA PROTOTIPAGEM SEM SOLDA

- Breadboard (matriz de contato): Possibilita prototipar um circuito sem fixá-lo em um lugar.
- Fios de jumper: Fios pequenos ideais para se construir um circuito em uma placa breadboard.
- Alicates de bico fino: Eles são pontudos, ótimos para segurar componentes minúsculos!
- Multímetro: Mede volts, amperes, resistência e uma série de outras coisas.
- Fonte de alimentação: Semelhante às fontes de alimentação para notebooks!

### FERRAMENTAS PARA SOLDAGEM

- Ferro de solda: Vem com regulação de soldadura e custa por volta de R\$50.
- Solda: Comprar uma solda sem chumbo é ótimo para sua saúde e para o meio ambiente.
- Suporte terceira mão: Ferramenta ideal para segurar placas e componentes.
- Suporte adesivo: Mais barato que um suporte do tipo terceira mão.
- Alicata de corte diagonal: Tem garras pontudas e é preciso.
- Alicata desencapador: Pode ser manual ou automático. Experimente os dois.
- Sugador de solda: Suga o excesso de solda durante a dessoldagem (erros).
- Malha de fita dessoldadora: Outro modo de realizar a dessoldagem.
- Bobinas e carretéis de fios: Comprar bobinas ou carretéis de fio flexíveis é sempre uma boa ideia.

# Arduino

Para

# leigos

Cap. de Amosia

Cap. de Amostra



# Arduino

Para **leigos**

Tradução da 2ª Edição

De John Nussey



ALTA BOOKS  
EDITORA  
Rio de Janeiro, 2019

Cap. de Amostra

## Sobre o Autor

---

John Nussey é um empresário, designer e tecnólogo que mora em Londres.

Seu trabalho envolve o uso da tecnologia de jeitos novos e interessantes, e também abrange muitas áreas, incluindo eletrônica, computação física, prototipagem e design de produtos e a Internet das Coisas.

No decorrer de sua carreira, ele trabalhou em muitos projetos diversificados de clientes como ARUP, BBC e National Maritime Museum. Também fundou empresas, incluindo a ONN Studio, que ajuda as pessoas a concretizarem suas ideias de produtos, e a Ding, que produz belos produtos para casas inteligentes.

John é paladino orgulhoso do Arduino e ensinou o ofício de design de interação e prototipagem para pessoas de todas as idades, competências e habilidades em diversas instituições, incluindo o Goldsmiths College, a Bartlett School of Architecture e o Royal College of Art.

## Dedicatória

---

À Avril O'Neil, o amor da minha vida (e única pessoa em quem confio com um ferro de solda), por compartilhar tantas experiências comigo e dar apoio quando mais preciso; à Roanne O'Neil, por nossas estimulantes conversas literárias; e à Alexandra Deschamps-Sonsino e Massimo Banzi, por me apresentarem ao Arduino, fato que me possibilitou seguir uma carreira tão diversificada e fascinante.

## Agradecimentos do Autor

---

Gostaria de agradecer ao pessoal da Wiley, especialmente à Susan Pink e Katie Mohr, por serem sempre positivas e incentivadoras.

Muito obrigado a Guy Hart-Davis, por sua excelente edição técnica.

Meus sinceros agradecimentos aos meus amigos, à minha família e aos leitores do Arduino Para Leigos que conheço desde 2013, pelo incentivo e entusiasmo. Adoro criar coisas. Espero que este livro o inspire a criar as próprias coisas e a encontrar a mesma paixão que encontrei.

Cap. de Amostra

# Sumário Resumido

---

Prefácio .....	xix
Introdução .....	1
<b>Parte 1: Conhecendo Melhor o Arduino</b> .....	5
CAPÍTULO 1: Descobrimdo o Arduino .....	7
CAPÍTULO 2: Encontre Sua Placa e Familiarize-se com Ela .....	17
CAPÍTULO 3: Piscando um LED .....	37
<b>Parte 2: Pegando ojeito com o Arduino</b> .....	55
CAPÍTULO 4: Ferramentas do Ofício .....	57
CAPÍTULO 5: Manual sobre Eletricidade e Conjunto de Circuitos .....	69
CAPÍTULO 6: Sketches Básicos: Entradas, Saídas e Comunicação .....	85
CAPÍTULO 7: Mais Sketches Básicos: Movimento e Som Neste Capítulo .....	119
<b>Parte 3: Construindo os Alicerces</b> .....	159
CAPÍTULO 8: Aprendendo com Exemplos .....	161
CAPÍTULO 9: Soldagem .....	177
CAPÍTULO 10: Desenvolvendo Suas Habilidades com o Código .....	205
CAPÍTULO 11: Use o Bom Senso com os Sensores .....	239
CAPÍTULO 12: Torne-se um Especialista em Shields e Bibliotecas .....	289
<b>Parte 4: Explorando Melhor o Software</b> .....	307
CAPÍTULO 13: Familiarizando-se com o Processing .....	309
CAPÍTULO 14: Processando o Mundo Físico .....	327
<b>Parte 5: A Parte dos Dez</b> .....	363
CAPÍTULO 15: Dez Lugares para Aprender Mais sobre Arduino .....	365
CAPÍTULO 16: Dez Lojas Excelentes para Conhecer .....	369
<b>Índice</b> .....	373

Cap. de Amostra

# Sumário

PREFÁCIO .....	XIX
INTRODUÇÃO .....	1
Sobre Este Livro.....	1
Penso que.....	2
Ícones Usados Neste Livro .....	2
Além Deste Livro .....	3
De Lá para Cá, Daqui para Lá .....	3
PARTE 1: CONHECENDO MELHOR O ARDUINO .....	5
<b>CAPÍTULO 1: Descobrindo o Arduino</b> .....	7
De Onde Veio o Arduino?.....	8
Aprendizagem pela Prática .....	12
Patching .....	12
Hacking.....	12
Circuit bending .....	13
Eletrônica .....	14
Entradas .....	15
Saídas .....	15
Open Source .....	15
<b>CAPÍTULO 2: Encontre Sua Placa e Familiarize-se com Ela</b> .....	17
Conhecendo Melhor o Arduino Uno .....	18
Os cérebros: Chip microcontrolador ATmega328P .....	19
Soquetes header.....	20
Pinos digitais .....	21
Pinos com entradas analógicas .....	21
E quanto à saída analógica? .....	21
Pinos de alimentação.....	22
USB .....	22
Fonte de alimentação.....	22
LEDs.....	23
Botão de reset .....	24
Descobrir Outras Placas Arduino .....	24
Placas Arduino oficiais .....	24
Arduinos aprovados .....	26

Comprando Placas Arduino .....	27
Loja Arduino .....	27
Adafruit .....	28
SparkFun .....	28
Seeed Studio .....	28
Watterott Electronic .....	28
Distribuidores de eletrônicos .....	29
Amazon .....	29
eBay .....	29
Apetrechos: Kits para Iniciantes .....	30
Preparando um Espaço para Trabalho .....	33
Instalando o Arduino .....	33
Explorando o Ambiente Arduino .....	34
Usando o Arduino Web Editor .....	36
<b>CAPÍTULO 3: Piscando um LED</b> .....	37
Trabalhando com o Seu Primeiro Sketch Arduino .....	38
Encontrando o Blink sketch .....	38
Reconhecendo sua placa .....	39
Configurando o software .....	40
Fazendo o upload do sketch .....	42
Você não merece palmas, mas o Tocantins inteiro! .....	43
O que acabou de acontecer? .....	43
Analisando o Sketch Mais de Perto .....	44
Comentários .....	46
Funções .....	47
A função setup .....	48
Loop .....	50
Um Blink Mais Brilhante .....	52
Ajustando o Sketch .....	53
<b>PARTE 2: PEGANDO OJEITO COM O ARDUINO</b> .....	55
<b>CAPÍTULO 4: Ferramentas do Ofício</b> .....	57
Encontrando as Ferramentas Adequadas para o Trabalho .....	58
Breadboard .....	58
Fios de jumper .....	60
Alicates de bico fino .....	62
Multímetro .....	62
Usando um Multímetro para Medir Tensão, Corrente e Resistência .....	64
Medindo a tensão em um circuito .....	64
Medindo a corrente em um circuito .....	65

	Medindo a resistência de um resistor.....	66
	Medindo a resistência de um resistor variável .....	67
	Verificando a continuidade (em bipes) do seu circuito .....	67
<b>CAPÍTULO 5:</b>	<b>Manual sobre Eletricidade e Conjunto de Circuitos</b> .....	69
	Entendendo a Eletricidade .....	70
	Usando Equações para Construir Seus Circuitos.....	71
	Lei de Ohm.....	72
	Calculando a energia .....	74
	Lei de Joule.....	75
	Trabalhando com Diagramas de Circuito.....	76
	Diagrama de circuito simples .....	76
	Usando um diagrama de circuito com um Arduino .....	79
	Código de Cores .....	80
	Datasheets (Fichas Técnicas).....	81
	Gráficos de Cores de Resistores.....	82
<b>CAPÍTULO 6:</b>	<b>Sketches Básicos: Entradas, Saídas e Comunicação</b> .....	85
	Carregando um Sketch.....	86
	Usando a Modulação por Largura de Pulso (PWM) .....	87
	O Efeito Fader do LED no Sketch .....	87
	Entendendo o sketch Fade .....	91
	Declarações .....	92
	Variáveis .....	92
	Ajustando o sketch Fade .....	94
	O Sketch Button.....	96
	Compreendendo o sketch Button .....	100
	Ajustando o sketch Button .....	101
	O Sketch AnalogInput.....	102
	Entendendo o sketch AnalogInput.....	106
	Ajustando o sketch AnalogInput.....	108
	Um Papo sobre o Monitor Serial .....	109
	O Sketch DigitalReadSerial .....	109
	Compreendendo o sketch DigitalReadSerial .....	112
	O Sketch AnalogInOutSerial .....	113
	Entendendo o sketch AnalogInOutSerial .....	116
<b>CAPÍTULO 7:</b>	<b>Mais Sketches Básicos: Movimento e Som Neste Capítulo</b> .....	119
	Trabalhando com Motores Elétricos .....	119
	Descobrimos os Diodos .....	121

Girando um Motor de Corrente Contínua (CC) .....	122
O sketch motor .....	122
Compreendendo o sketch Motor .....	126
Mudando a Velocidade do Seu Motor .....	127
O sketch MotorSpeed .....	127
Entendendo o sketch MotorSpeed .....	128
Controlando a Velocidade do Seu Motor .....	129
O sketch MotorControl .....	129
Entendendo o Sketch MotorControl .....	132
Ajustando o sketch MotorControl .....	132
Conhecendo Melhor os Servomotores .....	133
Criando Movimentos Longos e Contínuos .....	134
O sketch Sweep .....	134
Compreendendo o sketch Sweep .....	137
Controlando Seu Servo .....	139
O sketch Knob .....	139
Entendendo o sketch Knob .....	142
Fazendo um Som .....	143
Piezo buzzer .....	143
O sketch toneMelody .....	146
Entendendo o sketch .....	152
Fazendo um Instrumento .....	154
O sketch PitchFollower .....	154
Compreendendo o sketch .....	157

## PARTE 3: CONSTRUINDO OS ALICERCES .....

### CAPÍTULO 8: **Aprendendo com Exemplos** .....

Skube .....	161
Leituras complementares .....	163
Chorus .....	163
Como a Chorus funciona .....	164
Leituras complementares .....	165
Push Snowboarding .....	165
Como funciona .....	166
Leituras complementares .....	167
Baker Tweet .....	167
Como o Baker Tweet funciona .....	167
Leituras complementares .....	168
O Compass Lounge e Compass Card do Museu Nacional Marítimo de Londres .....	168
Como o Compass Lounge e o Compass Card funcionam .....	169
Leituras complementares .....	171

A Luminária Good Night . . . . .	171
Como a Good Night Lamp funciona . . . . .	172
Leituras complementares . . . . .	172
A Little Printer . . . . .	172
Como a Little Printer funciona . . . . .	173
Leituras complementares . . . . .	174
Flap to Freedom . . . . .	174
Como funciona o Flap to Freedom . . . . .	174
Leituras complementares . . . . .	175
<b>CAPÍTULO 9: Soldagem . . . . .</b>	<b>177</b>
Noções Básicas sobre Soldagem . . . . .	177
Juntando Tudo que É Necessário para a Soldagem . . . . .	178
Montando um espaço de trabalho . . . . .	178
Escolhendo um ferro de solda . . . . .	179
Solda . . . . .	183
Terceira mão (mão auxiliar) . . . . .	184
Massa adesiva . . . . .	185
Alicates cortadores de fio . . . . .	186
Alicates descascadores de fios . . . . .	186
Alicates de bico reto e liso . . . . .	187
Multímetro . . . . .	187
Sugador de solda . . . . .	187
Malha dessoldadora . . . . .	188
Fios conectores . . . . .	189
Preze pela Sua Segurança ao Soldar . . . . .	190
Manuseando seu ferro de solda . . . . .	190
Mantendo seus olhos protegidos . . . . .	190
Trabalhando em um ambiente ventilado . . . . .	190
Limpando seu ferro de solda . . . . .	190
Não ingira a solda! . . . . .	191
Montando um Shield . . . . .	191
Preparando todas as peças do circuito . . . . .	193
Montagem . . . . .	193
Pinos header . . . . .	194
Aprendendo Sua Técnica de Soldagem . . . . .	195
Construindo Seu Circuito . . . . .	198
Conhecendo seu circuito . . . . .	199
Expondo seu circuito . . . . .	199
Preparando seu fio . . . . .	200
Soldando seu circuito . . . . .	200
Limpando tudo . . . . .	201
Testando seu shield . . . . .	201

Protegendo Seu Projeto . . . . .	202
Gabinetes de proteção . . . . .	202
Fiação . . . . .	203
Protegendo a placa e outros elementos. . . . .	204
<b>CAPÍTULO 10: Desenvolvendo Suas Habilidades com o Código</b> . . . . .	205
Um Blink Melhor . . . . .	205
Configurando o sketch BlinkWithoutDelay . . . . .	207
Entendendo o sketch BlinkWithoutDelay . . . . .	210
Removendo o Bounce do Seu Botão. . . . .	212
Configurando o sketch Debounce . . . . .	213
Entendendo o sketch Debounce . . . . .	216
Criando um Botão Melhor . . . . .	218
Configurando o sketch StateChangeDetection . . . . .	218
Entendendo o sketch StateChangeDetection . . . . .	222
Suavizando Seus Sensores . . . . .	225
Configurando o sketch Smoothing . . . . .	226
Compreendendo o sketch Smoothing . . . . .	229
Calibrando Suas Entradas . . . . .	231
Configurar o sketch Calibration . . . . .	232
Compreendendo o sketch de calibração . . . . .	235
<b>CAPÍTULO 11: Use o Bom Senso com os Sensores</b> . . . . .	239
Construa Botões Facilmente . . . . .	240
Implementando o sketch DigitalInputPullup . . . . .	241
Entendendo o sketch DigitalInputPullup . . . . .	244
Explorando os Sensores Piezo . . . . .	245
Implementando o sketch Knock. . . . .	246
Compreendendo o sketch Knock. . . . .	250
Utilizando Sensores de Pressão, de Força e de Carga . . . . .	251
Implementando o sketch toneKeyboard . . . . .	254
Compreendendo o sketch toneKeyboard . . . . .	257
Sensores com Estilo . . . . .	258
Implementando o sketch CapPinSketch. . . . .	261
Entendendo o sketch CapPinSketch . . . . .	264
Ativando os Lasers . . . . .	267
Implementando o sketch AnalogInOutSerial . . . . .	269
Compreendendo o sketch AnalogInOutSerial . . . . .	272
Detectando Movimento . . . . .	272
Implementando o sketch DigitalReadSerial . . . . .	274
Entendendo o sketch DigitalReadSerial . . . . .	277

Calculando a Distância . . . . .	278
Implementando o sketch MaxSonar . . . . .	279
Entendendo o sketch MaxSonar . . . . .	283
Testando, 1, 2, 3...Tem Alguém Aí? . . . . .	284
Implementando o sketch AnalogInOutSerial . . . . .	285
Compreendendo o sketch AnalogInOutSerial . . . . .	288
<b>CAPÍTULO 12: Torne-se um Especialista em Shields e Bibliotecas</b> . . . . .	289
Analisando os Shields . . . . .	290
Levando em consideração as combinações . . . . .	290
Análise dos shields . . . . .	291
Fique por dentro . . . . .	301
Explorando as Bibliotecas . . . . .	302
Reavaliando as bibliotecas-padrão . . . . .	302
Instalando as bibliotecas adicionais . . . . .	304
Instalando as bibliotecas colaborativas . . . . .	306
<b>PARTE 4: EXPLORANDO MELHOR O SOFTWARE</b> . . . . .	307
<b>CAPÍTULO 13: Familiarizando-se com o Processing</b> . . . . .	309
Analisando como as Coisas Funcionam . . . . .	310
Instalando o Processing . . . . .	312
Conferindo o Processing . . . . .	313
Testando Seu Primeiro Sketchno Processing . . . . .	315
Desenhando formatos . . . . .	318
Alterando a cor e a opacidade . . . . .	322
Brincando com a interação . . . . .	325
<b>CAPÍTULO 14: Processando o Mundo Físico</b> . . . . .	327
Fazendo um Botão Virtual . . . . .	327
Organizando o código do Arduino . . . . .	329
Organizando o código do Processing . . . . .	330
Entendendo o sketch PhysicalPixel do Processing . . . . .	333
Compreendendo o sketch PhysicalPixel do Arduino . . . . .	337
Desenhando um Gráfico . . . . .	339
Organizando o código do Arduino . . . . .	341
Organizando o código do Processing . . . . .	342
Entendendo o sketch Graph do Arduino . . . . .	344
Compreendendo o sketch Graph do Processing . . . . .	345
Enviando Múltiplos Sinais . . . . .	349
Organizando o código do Arduino . . . . .	351
Organizando o código do Processing . . . . .	353

Compreendendo o sketch SerialCallResponse do Arduino . . .	355
Compreendendo o sketch SerialCallResponse do Processing . . . . .	358
<b>PARTE 5: A PARTE DOS DEZ . . . . .</b>	<b>363</b>
<b>CAPÍTULO 15: Dez Lugares para Aprender Mais sobre     Arduino . . . . .</b>	<b>365</b>
Blog do Arduino . . . . .	366
Hack a Day . . . . .	366
SparkFun . . . . .	366
MAKE . . . . .	366
Adafruit . . . . .	367
Instructables . . . . .	367
YouTube . . . . .	367
Hackerspaces . . . . .	367
Fórum . . . . .	368
Amigos, Colegas e Workshops . . . . .	368
<b>CAPÍTULO 16: Dez Lojas Excelentes para Conhecer . . . . .</b>	<b>369</b>
Adafruit . . . . .	369
Loja Arduino . . . . .	370
Seeed Studio . . . . .	370
SparkFun . . . . .	370
Allied Electronics . . . . .	371
Newark Electronics . . . . .	371
Mouser . . . . .	371
Digi-Key . . . . .	371
eBay . . . . .	372
Dumpster Diving . . . . .	372
<b>ÍNDICE . . . . .</b>	<b>373</b>

# Prefácio

---

O momento em que se publica um livro *Para Leigos* é, sem dúvida, um marco na história de um produto.

A programação embarcada (ou embutida) de computadores costumava ser uma coisa difícil, reservada apenas aos engenheiros experientes dispostos a dominar a complicada linguagem Assembly. Nos últimos anos, no entanto, muitas plataformas tentaram fazer com que a programação embarcada ficasse mais simples e mais acessível às pessoas comuns. O Arduino é uma das mais recentes tentativas de tornar a tecnologia menos assustadora e mais criativa.

Com John, autor deste livro, acompanhamos a adoção dessa ferramenta criativa por designers e artistas em Londres, abrindo caminho em muitos projetos memoráveis. Hoje em dia, o Arduino escapuliu do laboratório de Artes & Design e espalhou-se como um vírus, tornando-se a ferramenta favorita de todos os tipos de pessoas que têm grandes ideias e que desejam colocá-las em prática.

Fico contente que John tenha decidido escrever este livro, porque ele é um dos primeiros usuários da plataforma Arduino, desde a época em que ela ainda era bastante experimental. Como ministrou aulas de Arduino por muitos anos, tem competência para apresentar o assunto a todos os públicos.

Os principiantes em Arduino, com as ferramentas e ensinamentos adequados — como os encontrados neste livro — desenvolverão, em pouco tempo, verdadeiras habilidades.

— Massimo Banzi

Cap. de Amostra

# Introdução

---

O Arduino é uma ferramenta, uma comunidade e um modo de pensar que impacta a maneira como usamos e entendemos a tecnologia. Ele reacendeu a chama do interesse e da compreensão pela eletrônica em muitas pessoas (aconteceu comigo) que achavam que a eletrônica era algo que todos haviam deixado para trás, lá na escola técnica.

O Arduino é uma pequena placa de circuito que tem um potencial gigantesco. Pode ser utilizada para fazer piscar um sinal de código Morse usando um único diodo emissor de luz (LED) ou para controlar todas as luzes de um prédio, dependendo da distância em que você está. As capacidades do Arduino estão limitadas somente pela sua imaginação.

O Arduino também está viabilizando uma nova abordagem prática para o ensino técnico, diminuindo o nível de dificuldade para aqueles que querem usar a eletrônica a fim de concluir pequenos projetos, e, assim espero, incentivando as pessoas a ler mais sobre as coisas que ele pode realizar.

Uma comunidade enorme e crescente de arduinistas surgiu — usuários e desenvolvedores que aprendem uns com os outros e contribuem para a filosofia do código aberto compartilhando os detalhes de seus projetos. Essa atitude de código aberto é responsável pela popularidade massiva do Arduino.

O Arduino é mais do que apenas um gadget: é uma ferramenta. Uma tecnologia que facilita a compreensão e o uso tecnológico da atualidade.

Desse modo, se você não achar interessante a perspectiva de entender as possibilidades ilimitadas da tecnologia, por favor, deixe este livro de lado e desista.

Caso contrário, continue lendo!

## Sobre Este Livro

---

Este é um livro técnico, mas não é destinado somente aos tecnólogos de plan-tão. O Arduino foi projetado para ser usado por qualquer pessoa, seja ela técnica, criativa, habilidosa ou apenas curiosa. Tudo o que você precisa é de uma mente aberta ou de um problema a ser solucionado, e logo descobrirá maneiras de utilizar o Arduino que possam beneficiá-lo.

Esta obra se inicia pelo nível mais básico, para você começar a usar e entender o Arduino. Às vezes, ao longo do livro, talvez eu mencione uma série de coisas técnicas que, como qualquer coisa, levam tempo para serem compreendidas.

Vamos percorrer todos os conceitos básicos e, depois, passaremos para as atividades mais avançadas.

Muita coisa neste livro tem como base as minhas experiências de aprendizado e de ensino. Aprendi tudo a respeito do Arduino a partir da estaca zero, mas sempre achei que a melhor maneira de aprender é na prática, fazendo seus próprios projetos. A estratégia é entender os conceitos básicos que abordarei neste livro e, em seguida, aproveitar esse conhecimento pensando em como você pode aplicá-lo para resolver problemas, criar coisas ou simplesmente se entreter.

## Penso que...

Não estou tirando conclusões precipitadas sobre seus conhecimentos técnicos. O Arduino é uma plataforma fácil de usar para aprender sobre eletrônica e programação. É para pessoas de todos os tipos, crenças, profissões e estilos, seja você designer, artista ou hobbista.

Ele pode ser também uma ótima plataforma para pessoas que já são técnicas. Talvez você tenha aprendido um bocado de programação, mas queira concretizar seus projetos no mundo real de alguma forma. Ou talvez você tenha trabalhado com dispositivos eletrônicos e queira ver o que o Arduino pode oferecer.

Seja lá quem for, você descobrirá que o Arduino tem um grande potencial. Cabe a você decidir o que fazer com isso.

## Ícones Usados Neste Livro

O *Arduino Para Leigos* utiliza ícones para enfatizar as coisas importantes. Fique atento ao seguinte:



DICA

Este ícone ressalta algumas informações valiosas. Esta informação pode ser de caráter técnico para ajudá-lo a concluir um projeto com mais facilidade ou para solucionar um problema corriqueiro.



CUIDADO

As placas Arduino não são perigosas por natureza; na verdade, elas são feitas para serem extremamente seguras e fáceis de usar. Porém, caso sejam usadas em um circuito sem o mínimo de planejamento adequado, tampouco atenção ou cuidado, elas podem causar danos ao seu circuito, ao seu computador e a você. Quando vir um ícone de CUIDADO, por favor, preste atenção.



LEMBRE-SE

Às vezes, você precisa levar em consideração determinados pontos antes de prosseguir com uma tarefa. O ícone LEMBRE-SE está aqui justamente para fazê-lo se lembrar de tais pontos.



Algumas informações são mais técnicas do que outras e não são para aqueles que não aguentam o tranco. A graça do Arduino é que você não precisa entender completamente e de cara os detalhes técnicos. Você pode pular qualquer coisa que esteja marcada com este ícone caso ela seja mais complicada do que consegue lidar no momento; você sempre pode voltar a ela quando estiver pronto para isso.

## Além Deste Livro

Além do que você está lendo agora, este produto vem com uma Folha de Cola gratuita, para acesso em qualquer lugar, que fornece informações sobre como usar resistores, obter as ferramentas das quais você precisa e alguns atalhos do sistema. Para obtê-la, acesse [www.altabooks.com.br](http://www.altabooks.com.br) e digite *Arduino Para Leigos* na caixa de pesquisa. Faça o download da Folha de Cola completa, bem como de erratas e possíveis arquivos de apoio. Eu também disponibilizo um capítulo-bônus que ensina tudo a respeito de como usar seu Arduino para hackear outros hardwares, como jogos, controladores e brinquedos.

## De Lá para Cá, Daqui para Lá

Se você tem dúvidas sobre por onde começar, sugiro o início. No final do Capítulo 2, você terá adquirido um entendimento simples do Arduino e saberá onde é possível comprar um kit para continuar aprendendo.

Caso já tenha usado o Arduino antes, talvez queira ir direto ao Capítulo 4, que aborda os princípios básicos novamente, ou ir direto para a área que lhe interesse.

# Cap. de Amostra

1

# Conhecendo Melhor o Arduino

Cap. de Armostra

### **NESTA PARTE . . .**

Saiba tudo sobre a pequena placa de circuito azul.

Descubra tudo o que você precisa para conhecer o Arduino e onde comprá-lo.

Aprenda como usar o incrível poder de um LED, controlando como ele pisca com linhas de código simples.

- » **Descobrimo o que seria um Arduino**
- » **Entendendo de onde ele vem**
- » **Apresentando os princípios básicos**

## Capítulo **1**

# Descobrimo o Arduino

O Arduino pode ser tanto um hardware como um software. A placa Arduino é uma placa de circuito impresso (PCB) projetada para utilizar um chip microcontrolador, bem como para usar outras entradas e saídas. A placa tem muitos outros componentes eletrônicos necessários para o microcontrolador funcionar ou para maximizar as próprias funcionalidades.

Um *microcontrolador* é um pequeno computador dentro de um único circuito integrado ou chip de computador. Os microcontroladores são uma excelente maneira de programar e controlar os aparelhos eletrônicos. As *placas do microcontrolador* têm um chip microcontrolador e outros conectores e componentes valiosos que permitem ao usuário conectar entradas e saídas. Dentre os exemplos de dispositivos com placas de microcontroladores estão a placa de circuito Wiring, o PIC e o Basic Stamp.

Você escreve código no software Arduino para dizer ao microcontrolador o que fazer. Por exemplo, ao escrever uma linha de código, você pode dizer a um LED (diodo emissor de luz) para piscar (o famoso blink) e esmaecer. Se conectar um botão (pushbutton ou chave momentânea) e adicionar outra linha de código, poderá dizer ao LED para piscar somente quando o botão for pressionado. Em

seguida, você talvez queira dizer ao LED para piscar somente quando o botão for segurado. Dessa forma, você pode criar de modo rápido um comportamento para um sistema, coisa que seria difícil de conseguir sem um microcontrolador.

Como um computador convencional, uma placa Arduino pode executar uma infinidade de funções, mas não tem muita utilidade sozinha. Exigem-se entradas e saídas para que ela seja útil. Essas entradas e saídas possibilitam que um computador — e um Arduino — identifiquem e leiam os objetos no mundo externo.

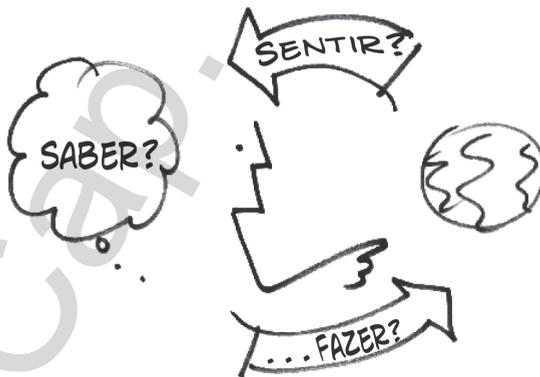
Antes de seguir adiante, talvez algumas coisas o ajudem a compreender a história do Arduino.

## De Onde Veio o Arduino?

O Arduino nasceu lá na Itália, no Interaction Design Institute Ivrea (IDII), em uma pós-graduação de design de interação focada em como as pessoas interagem com produtos, sistemas e ambientes digitais, e como eles, por sua vez, nos influenciam.

O termo *design de interação* foi inventado por Bill Verplank e Bill Moggridge em meados da década de 1980. O esboço na Figura 1-1 de Bill Verplank ilustra a premissa básica do design de interação: se você faz alguma coisa, sente uma mudança e, a partir disso, consegue saber sobre o mundo.

### DESIGN DE INTERAÇÃO



**FIGURA 1-1:** O princípio do design de interação, ilustrado por Bill Verplank.

Cortesia de Bill Verplank

Embora o design de interação seja um princípio geral, ele se refere mais especificamente a como interagimos com computadores convencionais usando periféricos (mouses, teclados e telas sensíveis ao toque) para navegar em um ambiente digital que, por sua vez, é exibido graficamente em uma tela.

Outra via, denominada *computação física*, tem a ver com ampliar o alcance desses programas de computador, software ou sistemas por meio de produtos eletrônicos. Ao usar dispositivos eletrônicos, os computadores podem identificar mais a respeito do mundo e afetá-lo fisicamente.

Ambas as áreas — design de interação e computação física — exigem que os protótipos compreendam e explorem totalmente as interações, o que representa um obstáculo para os estudantes de design que não são técnicos.

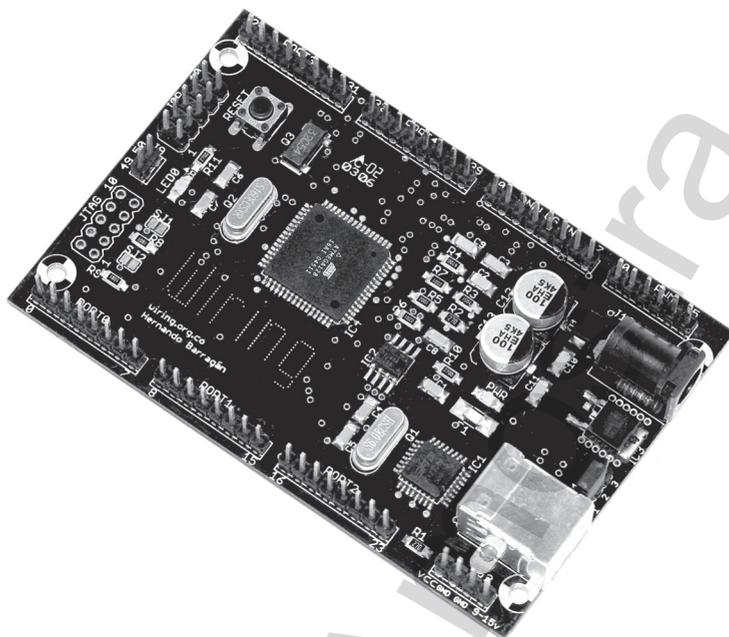
Em 2001, um projeto chamado Processing, iniciado por Casey Reas e Benjamin Fry, tinha como objetivo incluir na programação as pessoas que não eram programadoras, fazendo com que fosse rápido e fácil produzir visualizações e gráficos na tela. O projeto disponibilizou ao usuário um caderno de rascunho digital para testar ideias e dedicar um bocado de tempo experimentando as coisas. Esse projeto, em contrapartida, inspirou um projeto semelhante de experiências no mundo físico.

Em 2003, com base nos mesmos princípios do Processing, Hernando Barragán começou a desenvolver uma placa de microcontrolador chamada Wiring. Essa placa foi a precursora da placa Arduino.

À semelhança do projeto Processing, o projeto Wiring também tinha a finalidade de envolver artistas, designers e outras pessoas que não eram técnicas. No entanto, a Wiring foi projetada para levar as pessoas à eletrônica e também à programação. A placa de circuito Wiring (mostrada na Figura 1-2) era menos cara do que alguns outros microcontroladores, como o PIC e o Basic Stamp, mas, ainda assim, era um investimento considerável para os estudantes.

Em 2005, o projeto Arduino começou em resposta à necessidade de dispositivos acessíveis e fáceis de usar para os estudantes de design de interação utilizarem em seus projetos. Dizem as más-línguas que Massimo Banzi e David Cuartielles deram o nome ao projeto em homenagem a Arduin de Ivrea, um rei italiano; entretanto, ouvi de fontes confiáveis que Arduino também é o nome do barzinho perto da universidade, o que talvez faça mais sentido para o nome do projeto.

O projeto Arduino baseou-se em muitas das experiências do Wiring e do Processing. Por exemplo, uma influência indiscutível do Processing é a *interface gráfica do usuário* (GUI) no software Arduino. A princípio, a GUI foi “empresada” do Processing e, apesar de ainda parecer similar, ela já foi refinada para ser mais específica de acordo com o Arduino. Falo a respeito da interface do Arduino mais detalhadamente no Capítulo 3.

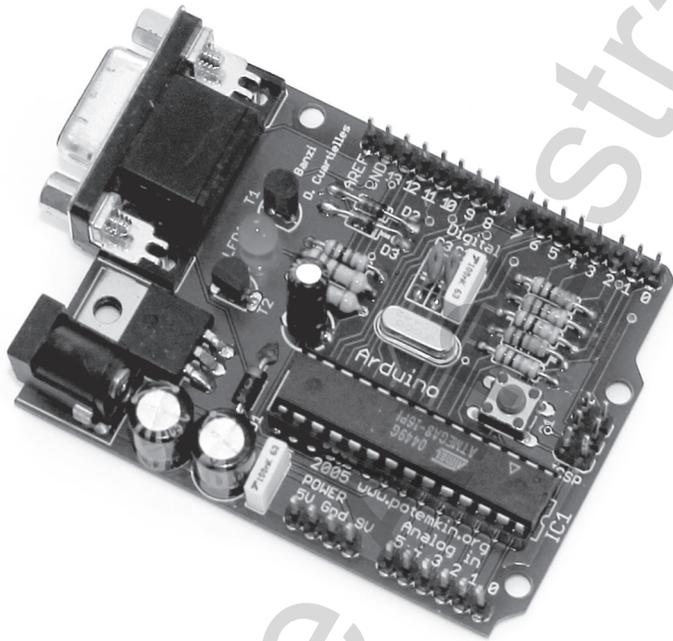


**FIGURA 1-2:**  
Uma placa  
Wiring  
anterior.

O Arduino também manteve a convenção de nomenclatura do Processing, chamando seus programas de *sketches* (um projeto no Arduino é chamado de *sketch*). Da mesma forma que o Processing oferece às pessoas um caderno de rascunho digital para criar e testar programas rapidamente, o Arduino também lhes possibilita uma maneira de esboçar suas ideias de hardware. No decorrer deste livro, mostro muitos sketches que permitem ao seu Arduino executar uma enorme variedade de tarefas. Ao usar e editar os sketches exemplificados neste livro, você pode compreender de imediato como eles funcionam. Você programará sozinho em um piscar de olhos. Cada sketch acompanha uma explicação linha a linha de como ele funciona para garantir que tudo seja esclarecido e nada esquecido.

A placa Arduino, mostrada na Figura 1-3, foi desenvolvida para ser mais resistente e tolerante do que a placa Wiring e outros microcontroladores anteriores. Não raro, os estudantes, sobretudo os de design ou artes, quebravam o próprio microcontrolador poucos minutos depois de começar a usá-lo, simplesmente colocando os fios no lugar errado. Essa fragilidade foi um grande problema, não somente em relação à questão financeira, mas também devido aos comentários das placas fora dos círculos técnicos. Você também pode trocar o chip do microcontrolador em um Arduino; caso o chip seja danificado, em vez de trocar a placa inteira, você pode substituir apenas o chip.

Outra diferença importante entre o Arduino e outras placas de microcontrolador é o custo. Em 2006, outro microcontrolador famoso, o Basic Stamp, custava quase quatro vezes mais (US\$119) que um Arduino (US\$32). Hoje, um Arduino Uno custa apenas US\$22 (no Brasil, o preço varia entre R\$45 e R\$60).



**FIGURA 1-3:**  
A placa  
Arduino se-  
rial original.

Em um dos meus primeiros seminários Arduino, disseram-me que o preço deveria ser acessível para os estudantes. Naquela época, o preço de uma boa refeição e uma taça de vinho era de cerca de US\$42, assim, se você tivesse prazo para entregar o projeto, poderia escolher abrir mão de uma boa refeição naquela semana e fazer o seu projeto.

Hoje, a variedade de placas Arduino no mercado é muito maior do que em 2006. No Capítulo 2, você saberá mais a respeito de apenas algumas das placas Arduino compatíveis com o Arduino e como elas apresentam diferenças em relação à variedade de soluções para seus próprios projetos. Além do mais, no Capítulo 12, você aprende tudo sobre um tipo especial de placa de circuito chamada *shield*, que pode adicionar recursos para lá de úteis ao seu Arduino e, em alguns casos fenomenais, transformá-lo em um receptor GPS (sistema de posicionamento global), um telefone celular ou até mesmo em um contador Geiger, mencionando apenas alguns exemplos.

# Aprendizagem pela Prática

As pessoas têm usado a tecnologia de várias maneiras para alcançar seus próprios objetivos sem se aprofundar nos detalhes da eletrônica. A seguir, veja algumas escolas de pensamento relacionadas que permitiram às pessoas brincarem com a eletrônica.

## Patching

*Patching* é uma técnica de experimentação com sistemas que usam cabos ou fios. Um exemplo de patching são as antigas centrais telefônicas. Para um operador colocá-lo em outra linha, ele ou ela tinha que conectar fisicamente um cabo em um painel.

Essa técnica também era popular para sintetizar música, como no sintetizador Moog. Quando um instrumento eletrônico gera um som, ele está, de fato, gerando uma tensão. As coleções diferentes de componentes no instrumento manipulam essa tensão (medida em volts) antes que ela seja produzida como um som audível. O sintetizador Moog funciona ao alterar o caminho que a tensão percorre, enviando-a através de um número de componentes diferentes para aplicar diferentes efeitos.

Como inúmeras combinações são possíveis, o músico continua a testar as coisas, em grande parte, por tentativa e erro. Porém, a interface simples indica que esse processo é extremamente rápido e exige pouca preparação para se seguir em frente.

## Hacking

Normalmente, *hacking* é um termo que se refere ao uso subversivo da tecnologia. Em termos mais gerais, no entanto, refere-se à exploração de sistemas, ao amplo uso ou à reprogramação deles para atender às suas necessidades.

Nesse sentido, é possível hackear tanto em hardware quanto em software. Um ótimo exemplo de hacking de hardware é um teclado com hack. Digamos que você queira usar um grande botão vermelho para passar de slide em slide em uma apresentação. A maioria dos programas de computadores tem atalhos de teclado e a maioria dos visualizadores de PDF passa para a próxima página em uma apresentação de slides quando o usuário pressiona a barra de espaço. Caso saiba disso, você quer, de preferência, um teclado com apenas uma barra de espaço.

Os teclados de hoje têm uma pequena placa de circuito, um pouco menor que um cartão de crédito (veja a Figura 1-4), incluindo muitos contatos que são conectados quando você pressiona teclas diferentes. Se você conseguir

encontrar a combinação correta, poderá conectar dois contatos usando um botão. Agora, toda vez que pressionar esse botão, você enviará um espaço para o seu computador.

Essa técnica é ótima para se esquivar das complexidades do hardware e conseguir os resultados desejados. No capítulo bônus “Hackeando Outros Hardwares” ([www.altabooks.com.br](http://www.altabooks.com.br)), você aprende mais sobre a satisfação de hackear e como pode construir peças de hardware hackeadas em seu projeto Arduino para controlar com facilidade dispositivos remotos, câmeras e até mesmo computadores.



**FIGURA 1-4:**  
O interior  
de um tecla-  
do, pronto  
para ser  
hackeado.

## Circuit bending

O conceito de *circuit bending* bate de frente com a educação tradicional e tem totalmente a ver com o experimentalismo espontâneo. Os brinquedos infantis são a base de sustentação da galera do circuit bending (os chamados circuit benders), mas, na verdade, qualquer dispositivo eletrônico tem potencial para ser testado.

Ao desmontar um brinquedo ou dispositivo e expor o circuito, você pode alterar o caminho da corrente para afetar o comportamento dela. Embora essa técnica seja semelhante ao patching, ela é mais imprevisível. No entanto, depois de encontrar uma combinação que apresente um bom resultado, você pode

adicionar ou substituir componentes, como resistores ou chaves eletrônicas, para dar ao usuário mais controle sobre o instrumento.

Geralmente, o circuit bending tem a ver com sonoridade, e o instrumento finalizado torna-se um sintetizador rudimentar ou uma bateria eletrônica. Dois dos dispositivos mais populares são o Speak & Spell (veja a Figura 1-5) e o Nintendo GameBoy. Músicos, como os da Modified Toy Orchestra ([modifiedtoyorchestra.com](http://modifiedtoyorchestra.com)) em suas próprias palavras, “exploram o potencial oculto e a mais-valia em potencial dentro da tecnologia obsoleta”. Assim, pense duas vezes antes de colocar seus brinquedos antigos no eBay!



**FIGURA 1-5:** Circuit bending da Modified Toy Orchestra e do Speak & Spell.

*Cortesia da Modified Toy Orchestra*

## Eletrônica

Ainda que haja muitos meios de se trabalhar a tecnologia, em algum momento você precisará de mais de tudo: mais precisão, mais complexidade e mais controle.

Se você aprendeu eletrônica na escola, provavelmente aprendeu como construir circuitos usando componentes específicos. Esses circuitos têm base exclusivamente nas propriedades químicas dos componentes e precisam ser calculados em detalhes para garantir que a quantidade correta de corrente percorra os componentes adequados.

Estes são os tipos de circuitos que você encontra como kits na Radio Shack (ou Maplin, no Reino Unido) que fazem um trabalho específico, como um timer de cozinha ou uma campainha de segurança que dispara quando você abre um pote de biscoitos. No Brasil, você pode encontrar kits no Mercado Livre ou nas seguintes lojas: EletroGate (<https://eletrogate.com>), Arduinolândia (<https://arduinolandia.com.br>), Loja Arduino: (<https://lojaarduino.com.br>). Tais kits são muito bons para funções específicas, mas não fazem mais do que isso.

É neste momento que entram os microcontroladores. Quando utilizados com conjuntos de circuitos analógicos, os microcontroladores podem viabilizar que esse circuito tenha um comportamento mais arrojado. Eles também podem ser reprogramados para executar diferentes funções conforme necessário. A sua placa Arduino é projetada com base nesses microcontroladores e, no Capítulo 2, você analisará de perto um Arduino Uno para ver exatamente como ele é projetado e do que é capaz.

O microcontrolador é o cérebro de um sistema, mas ele precisa de outras entradas e saídas eletrônicas a fim de identificar ou afetar as coisas em seu ambiente.

## Entradas

As *entradas* são os sentidos do Arduino. Elas informam à placa o que está acontecendo no mundo. Basicamente, uma entrada pode ser uma chave eletrônica, como um interruptor de luz em sua casa. Do lado oposto, poderia ser um giroscópio, dizendo ao Arduino a direção exata à qual está se opondo em três dimensões. Você aprenderá tudo a respeito das entradas básicas no Capítulo 6 e mais sobre a variedade de sensores e quando usá-los no Capítulo 11.

## Saídas

As *saídas* permitem que seu Arduino influencie o mundo real de alguma forma. Uma saída pode ser sutil e discreta, como a vibração de um celular, ou pode ser uma enorme exibição visual na lateral de um prédio que pode ser vista a quilômetros de distância. O primeiro sketch do livro lhe apresenta um LED piscando (veja o Capítulo 3). A partir daí, você pode controlar um motor elétrico (Capítulo 7) e até mesmo controlar uma tela de LCD (Capítulo 12).

# Open Source

O software open source (de código aberto), em particular o Processing, influenciou muitíssimo o desenvolvimento do Arduino. No mundo dos programas de computadores, o *open source* é uma filosofia por meio da qual as

peças compartilham os detalhes de um programa e incentivam os outros a usá-lo, recombiná-lo e redistribuí-lo como quiserem.

Assim como o software Processing é open source, o mesmo ocorre com o software e o hardware do Arduino. Isso significa que o software e o hardware do Arduino são liberados gratuitamente para serem adaptados conforme a necessidade. Você também encontra essa mesma filosofia de compartilhamento entre a comunidade nos fóruns sobre Arduino.

Nos fóruns oficiais do Arduino (<http://forum.arduino.cc> — conteúdo em inglês), e também na versão em português (<http://forum.arduino.cc/index.php?board=47.0>), e em muitos outros ao redor do mundo, as pessoas compartilharam o próprio código, projetos e perguntas para uma revisão informal feita pelos colegas. Esse compartilhamento possibilita a todo tipo de pessoa, incluindo engenheiros experientes, desenvolvedores talentosos, designers experientes e artistas inovadores, que empreste seus conhecimentos a novatos em algumas ou em todas essas áreas. Isso também viabiliza um meio de avaliar as áreas de interesse das pessoas, que ocasionalmente aprimoram o lançamento oficial do software Arduino ou o design da placa com refinamentos ou acréscimos. O site do Arduino tem uma área conhecida como Playground (<http://playground.arduino.cc>), no qual as pessoas são livres para fazer upload de seu código para a comunidade usar, compartilhar e editar.

Esse tipo de filosofia tem incentivado essa comunidade relativamente pequena a agrupar conhecimento em fóruns, blogs e sites, criando assim um vasto recurso para os novos arduinistas acessarem.

Apesar da natureza open source do Arduino, existe uma enorme lealdade ao Arduino como marca — tanto que há uma convenção de nomenclatura do Arduino para adicionar *-duino* ou *-ino* ao nome de placas e acessórios (para o desgosto dos italianos da equipe Arduino)!