

Explorando o Arduino®

Técnicas e ferramentas para
mágicas de engenharia

Jeremy Blum



ALTA BOOKS
E D I T O R A
Rio de Janeiro, 2016



Sobre o autor

Jeremy Blum recentemente recebeu seu diploma de mestrado em engenharia elétrica e computação pela Universidade de Cornell, onde anteriormente recebeu seu diploma de bacharel no mesmo campo. Na Cornell, ele supervisionou a criação e realização de vários edifícios sustentáveis em todo o mundo e no mercado interno por meio de sua fundação e liderança na Cornell University Sustainable Design, uma organização de design sustentável reconhecida nacionalmente que foi especificamente elogiada pelo CEO dos U. S. e World Green Building Councils. Nessa linha, Jeremy aplicou sua paixão pela engenharia elétrica para projetar sistemas residenciais de monitoramento de energia solar, sistemas revolucionários de iluminação LED por fibra óptica, e painéis solares inteligentes de rastreamento do sol. Ele também é responsável por ajudar a iniciar o primeiro espaço empresarial colaborativo que contribui para o desenvolvimento de dezenas de *start-ups* de estudantes (incluindo alguns de sua própria criação) a cada ano.

Jeremy projetou métodos premiados de controle protético, sistemas de reconhecimento de gesto e sistemas de automação de edifícios, entre muitas outras coisas. Ele projetou os componentes eletrônicos para as impressoras 3D MakerBot Replicator (que são usadas por pessoas em todo o mundo, e por organizações notáveis como a NASA), e o protótipo da eletrônica e do firmware para o Scanner 3D MakerBot Digitizer. Como um pesquisador no renomado Creative Machines Lab, ele tem contribuído para a criação de robôs que podem se montar sozinhos, robôs quadrúpedes autoguiados e impressoras 3D que redefinem o conceito de manufatura doméstica. Ele apresentou essa pesquisa em revistas especializadas e em conferências em lugares tão distantes como a Índia.

Jeremy produz vídeos do YouTube que introduziram milhões de pessoas à engenharia e estão entre os tutoriais sobre Arduino mais populares na internet. Ele é bem conhecido dentro das comunidades internacionais de *open source* e do Movimento Maker por seu desenvolvimento de projetos e tutoriais de hardware de *open source* que foram apresentados no Discovery Channel, e ganharam vários prêmios e *hack-a-thons*. Jeremy

foi selecionado pelo Instituto Americano de Engenheiros Elétricos e Eletrônicos como o New Face of Engineering de 2012.

Ele oferece serviços de consultoria de engenharia por meio de sua empresa, a Blum Idea Labs LLC, e ensina engenharia e sustentabilidade para jovens estudantes na cidade de Nova York. A paixão de Jeremy está melhorando a vida das pessoas e o nosso planeta por meio de soluções criativas de engenharia. Você pode aprender mais sobre Jeremy e seu trabalho em seu site: www.jeremyblum.com (o site possui conteúdo em inglês, assim como todos os outros sites indicados ao longo da obra).

Sobre o editor técnico

Scott Fitzgerald é um artista e educador que tem usado a plataforma Arduino como ferramenta de ensino e em sua prática desde 2006. Ele ensina computação física no Programa de Telecomunicações Interativas (ITP) da Universidade de Nova York desde 2005, introduzindo artistas e designers aos microcontroladores. Scott trabalha para a equipe Arduino, documentando a criação de novos produtos e tutoriais para introduzir a plataforma ao público em geral. Ele foi editor técnico da segunda edição de *Making Things Talk* em 2011, e foi o autor do livro que acompanha o Arduino Starter Kit oficial em 2012.



Agradecimentos

Em primeiro lugar, devo agradecer a meus amigos na Wiley por ajudar a tornar isso possível: Mary James, por incentivar-me a escrever este livro, em primeiro lugar; e Jennifer Lynn, por me manter no caminho enquanto eu escrevia todos os capítulos. Também devo um grande obrigado a Scott Fitzgerald por seu olhar crítico na edição técnica deste livro.

Se não fosse pelas excelentes pessoas na element14, talvez eu nunca tivesse começado a produzir minhas Arduino Tutorial Series, um prelúdio para o livro que você está prestes a ler. Sabrina Deitch e Sagar Jethani, em particular, têm sido parceiros maravilhosos com quem eu tive o privilégio de trabalhar.

Escrevi a maior parte deste livro enquanto simultaneamente, concluía meu mestrado e tocava duas empresas; portanto, sou imensamente grato aos meus professores e colegas que me aturaram enquanto eu tentava equilibrar todas minhas responsabilidades.

Por fim, quero agradecer à minha família, especialmente aos meus pais e ao meu irmão, David, cujo constante encorajamento lembra-me por que eu faço as coisas que faço.



Sumário resumido

Introdução		xix
Parte I	Fundamentos de engenharia do Arduino	1
Capítulo 1	Ligando o Arduino e fazendo-o piscar	3
Capítulo 2	Entradas e saídas digitais, e modulação por largura de pulso	19
Capítulo 3	Lendo sensores analógicos	41
Parte II	Controlando seu ambiente	61
Capítulo 4	Usando transistores e motores de tração	63
Capítulo 5	Produzindo sons	91
Capítulo 6	USB e comunicação serial	107
Capítulo 7	Registradores de deslocamento	145
Parte III	Interfaces de comunicação	161
Capítulo 8	O barramento I ² C	163
Capítulo 9	O barramento SPI	181
Capítulo 10	Interface com telas de cristal líquido	199
Capítulo 11	Comunicação sem fio com rádios XBee	221
Parte IV	Temas e projetos avançados	255
Capítulo 12	Interrupções por hardware e por timer	257
Capítulo 13	Registro de dados em log com cartões SD	277
Capítulo 14	Como conectar o Arduino à internet	313
Apêndice	Decifrando as especificações ATmega e o esquema do Arduino	341
Índice		349

Sumário

Introdução	xix
Parte I Fundamentos de engenharia do Arduino	1
Capítulo 1 Ligando o Arduino e fazendo-o piscar	3
Explorando a plataforma do Arduino	4
Funcionalidade do Arduino	4
Microcontroladores Atmel	6
Interfaces de programação	6
E/S geral e ADCs	7
Suprimentos de energia	7
Placas Arduino	8
Criando seu primeiro programa	13
Baixando e instalando o IDE Arduino	13
Executando o IDE e conectando-se ao Arduino	14
Analisando seu primeiro programa	16
Resumo	18
Capítulo 2 Entradas e saídas digitais, e modulação por largura de pulso	19
Saídas digitais	20
Conectando um LED e usando placas protoboards ou matrizes de contatos	20
Trabalhando com protoboards ou matrizes de contatos	21
Conectando LEDs	22
Programando saídas digitais	24
Usando laços for	25
Modulação por largura de pulso com analogWrite()	27
Lendo entradas digitais	29
Lendo entradas digitais com resistores pull-down	29
Trabalhando com botões de pulso	32
Construindo um LED RGB Nightlight controlável	35
Resumo	39

Capítulo 3	Lendo sensores analógicos	41
	Entendendo sinais analógicos e digitais	42
	Comparando sinais analógicos e digitais	43
	Convertendo um sinal analógico em um digital	44
	Lendo sensores analógicos com o Arduino: analogRead()	45
	Lendo um potenciômetro	45
	Usando sensores analógicos	50
	Trabalhando com sensores analógicos para detectar a temperatura	52
	Usando resistores variáveis para fazer seus próprios sensores analógicos	54
	Usando divisores de tensão resistivos	55
	Usando entradas analógicas para controlar saídas analógicas	56
	Resumo	59
Parte II	Controlando seu ambiente	61
Capítulo 4	Usando transistores e motores de tração	63
	Acionando motores DC	65
	Manipulando cargas indutivas de corrente alta	65
	Usando transistores como comutadores	66
	Usando diodos de proteção	67
	Usando uma fonte de energia secundária	68
	Conectando o motor	68
	Controlando a velocidade do motor com PWM	70
	Usando uma ponte H para controlar a direção do motor DC	72
	Construindo um circuito de ponte H	73
	Operando de um circuito de ponte H	76
	Acionando servomotores	80
	Compreendendo a diferença entre rotação contínua e servos padrão	80
	Entendendo o servo controle	80
	Controlando um servo	85
	Construindo um sensor de distância por varredura	86
	Resumo	90
Capítulo 5	Produzindo sons	91
	Entendendo como funcionam os alto-falantes	92
	As propriedades do som	92
	Como um alto-falante produz som	94
	Usando tone() para fazer sons	95
	Incluindo um arquivo de definição	95
	Conectando um alto-falante	96
	Criando sequências sonoras	99
	Usando arrays	99
	Criando arrays de notas e durações	100
	Completando o programa	101
	Compreendendo as limitações da função tone()	102

Construindo um micropiano	102
Resumo	105
Capítulo 6 USB e comunicação serial	107
Entendendo as capacidades de comunicação serial do Arduino	108
Placas Arduino com um conversor FTDI USB-serial	
interno ou externo	110
Placas Arduino com uma MCU ATmega secundária com	
USB emulando um conversor serial	112
Placas Arduino com MCU de uma única USB	114
Placas Arduino com capacidades de host USB	114
Ouvindo o Arduino	115
Usando instruções print	115
Usando caracteres especiais	117
Mudando as representações de tipo de dados	119
Conversando com o Arduino	119
Lendo informações a partir de um computador ou	
outro dispositivo serial	120
Instruindo o Arduino para ecoar dados de entrada	120
Entendendo as diferenças entre chars e ints	121
Enviando caracteres simples para controlar um LED	122
Enviando listas de valores para controlar um LED RGB	125
Conversando com um aplicativo desktop	127
Conversando com o Processing	127
Instalando o Processing	128
Controlando um sketch de Processing a partir do Arduino	129
Enviando dados do Processing para o Arduino	132
Aprendendo truques especiais com o Arduino Leonardo	
(e outros arduinos baseados no MCU 32U4)	134
Emulando um teclado	135
Digitando dados no computador	135
Comandando seu computador com emulação de teclas	139
Emulando um mouse	140
Resumo	144
Capítulo 7 Registradores de deslocamento	145
Entendendo registradores de deslocamento	146
Enviando dados seriais e paralelos	147
Trabalhando com o registrador de deslocamento 74HC595	148
Entendendo funções de pino do registrador de deslocamento	148
Entendendo como o registrador de deslocamento funciona	149
Deslocando dados seriais do Arduino	151
Convertendo entre formatos binários e decimais	154
Controlando animações de luz com um registrador de	
deslocamento	154
Construindo um “light rider”	154
Respondendo a entradas com um gráfico de barra LED	157
Resumo	160

Parte III	Interfaces de comunicação	161
Capítulo 8	O barramento I²C	163
	História do barramento I ² C	164
	Design de hardware I ² C	165
	Esquema de comunicação e números de identificação	165
	Requisitos de hardware e resistores pull-up	167
	Comunicando com uma sonda de temperatura I ² C	167
	Configurando o hardware	168
	Referenciando a especificação	169
	Escrevendo o software	171
	Combinando registradores de deslocamento, comunicação serial e comunicações I ² C	173
	Construindo o hardware para um sistema de monitoração de temperatura	173
	Modificando o programa integrado	174
	Escrevendo o sketch de Processing	177
	Resumo	180
Capítulo 9	O barramento SPI	181
	Visão geral do barramento SPI	182
	Hardware SPI e design de comunicação	183
	Configuração de hardware	184
	Esquema de comunicação	184
	Comparando SPI com I ² C	185
	Comunicando-se com um potenciômetro digital SPI	185
	Coletando informações a partir da especificação	186
	Configurando o hardware	189
	Escrevendo o software	190
	Criando um monitor audiovisual usando potenciômetros digitais SPI	193
	Configurando o hardware	194
	Modificando o software	195
	Resumo	197
Capítulo 10	Interface com telas de cristal líquido	199
	Configurando o LCD	200
	Usando a biblioteca LiquidCrystal para escrever no LCD	203
	Adicionando texto à tela	204
	Criando caracteres especiais e animações	206
	Construindo um termostato pessoal	209
	Configurando o hardware	210
	Exibindo dados no LCD	211
	Ajustando o <i>set point</i> com um botão	213
	Adicionando um aviso sonoro e uma ventoinha	214
	Juntando tudo: o programa completo	215
	Elevando o nível desse projeto	219
	Resumo	219

Capítulo 11	Comunicação sem fio com rádios XBee	221
	Entendendo as comunicações sem fio XBee	222
	Rádios XBee	223
	O shield de rádio XBee e conexões seriais	224
	Regulador 3.3V	226
	Deslocamento do nível lógico	226
	LED associado e LED RSSI	226
	Jumper ou comutador de seleção UART	226
	Opção de conexão serial UART por hardware <i>versus</i> software	227
	Configurando o XBees	228
	Configurando com um protetor ou um adaptador USB	228
	Opção de programação 1: Usando o Uno como um programador (não recomendado)	229
	Opção de programação 2: Usando o SparkFun USB Explorer (recomendado)	230
	Escolhendo as configurações de XBee e conectando o XBee com o computador host	230
	Configurando o XBee com X-CTU	231
	Configurando o XBee com um terminal serial	235
	Conversando com um computador sem fio	236
	Ligando um Arduino remoto	236
	USB com um computador ou um adaptador de parede 5V	237
	Baterias	237
	Adaptadores de energia de parede	239
	Revisitando os exemplos seriais: Controlando o processamento com um potenciômetro	239
	Revisitando os exemplos seriais: Controlando um LED RGB	243
	Conversando com outro Arduino: Construindo uma campainha sem fio	246
	Projeto de sistema	246
	Hardware transmissor	247
	Hardware receptor	248
	Software transmissor	249
	Software receptor	250
	Resumo	252
Parte IV	Temas e projetos avançados	255
Capítulo 12	Interrupções por hardware e por timer	257
	Usando interrupções de hardware	258
	Conhecendo as vantagens e desvantagens de polling e interrupção	259
	Facilidade de implementação (software)	260
	Facilidade de implementação (hardware)	260
	Multitarefa	260
	Precisão de aquisição	261
	Entendendo as capacidades de interrupção de hardware do Arduino	261

Construindo e testando um circuito de interrupção de botão com debouncing de hardware	262
Criando um botão com debouncing de hardware	262
Montando um circuito de teste completo	267
Escrevendo o software	267
Usando interrupções por timer	270
Entendendo interrupções por timer	270
Obtendo a biblioteca	270
Executando duas tarefas simultaneamente	271
Construindo um aparelho de som baseado em interrupção	272
Hardware do aparelho de som	272
Software do aparelho de som	273
Resumo	275
Capítulo 13 Registro de dados em log com cartões SD	277
Preparando-se para o registro de dados em log	278
Formatando dados com arquivos CSV	279
Preparando um cartão SD para registro de dados em log	279
A interface do Arduino com um cartão SD	284
Shields de cartão SD	284
Interface SPI de cartão SD	288
Gravando em um cartão SD	289
Lendo um cartão SD	293
Usando um relógio de tempo real	297
Entendendo relógios de tempo real	298
Usando o relógio de tempo real DS1307	298
Usando a RTC Arduino Third-Party Library	299
Usando o relógio de tempo real	300
Instalando os módulos de cartão de RTC e SD	300
Atualizando o software	301
Construindo um registro de log de presença	305
Hardware de registro de log	306
Software de registro de log	307
Análise de dados	311
Resumo	312
Capítulo 14 Como conectar o Arduino à internet	313
A web, o Arduino e você	314
Jargão de rede	314
Endereço IP	314
Network Address Translation	315
Endereço MAC	316
HTML	316
HTTP	316
GET/POST	316
DHCP	316
DNS	317

Clientes e servidores	317
Colocando o Arduino em rede	317
Controlando o Arduino via web	318
Configurando o hardware de controle de E/S	318
Projetando uma simples página web	318
Escrevendo um sketch de servidor Arduino	320
Conectando-se à rede e recuperando um IP via DHCP	321
Respondendo a um cliente	321
Juntando as peças: Sketch de servidor web	322
Controlando o Arduino via rede	326
Controlando o Arduino via rede local	326
Usando o encaminhamento de porta para controlar o Arduino de qualquer lugar	327
Enviando dados em tempo real a um serviço de gráficos	329
Construindo uma alimentação de dados ao vivo sobre Xively	330
Criando uma conta Xively	330
Criando um feed de dados	330
Instalando as bibliotecas Xively e HttpClient	331
Colocando o Arduino em rede	332
Configurando o sketch de Xively e executando o código	332
Exibindo dados na web	335
Adicionando componentes de feed	336
Adicionando um sensor de temperatura analógico	336
Adicionando leituras de sensor adicionais ao fluxo de dados	336
Resumo	339
Apêndice	
Decifrando as especificações ATmega e o esquema do Arduino	341
Especificações	341
Analisando uma especificação	341
Entendendo a pinagem dos componentes	344
Entendendo o esquema do Arduino	345
Índice	349



Introdução

Você tem um excelente senso de oportunidade. Como muitas vezes eu gosto de dizer: “Nós estamos vivendo no futuro”. Com as ferramentas disponíveis hoje, muitas das quais aprenderá neste livro, você tem a oportunidade e a capacidade de dobrar o mundo físico ao seu capricho. Até muito recentemente, não era possível para alguém pegar um microcontrolador e fazê-lo controlar seu mundo em poucos minutos. Como você deve ter adivinhado, um *microcontrolador* é uma plataforma programável que lhe dá o poder de definir o funcionamento dos sistemas mecânicos, elétricos e de software complexos usando comandos relativamente simples. As possibilidades são infinitas, e a plataforma microcontroladora Arduino vai se tornar sua nova ferramenta favorita à medida que você explora o mundo da eletrônica, programação, interação homem-computador, arte, sistemas de controle e muito mais. Durante todo o curso deste livro, você vai usar o Arduino para fazer de tudo, desde a detecção de movimento até a criação de sistemas de controle sem fios e se comunicar pela internet.

Quer você seja completamente novo em qualquer tipo de engenharia ou um veterano querendo começar o projeto de sistemas embarcados, o Arduino é um ótimo lugar para começar. Você está procurando uma referência geral para o desenvolvimento em Arduino? Este livro também é perfeito para você. Este livro o orienta por meio de uma série de projetos específicos, mas você também vai achar fácil retornar ao livro para consultar trechos de código, melhores práticas, esquemas do sistema e muito mais. As práticas de engenharia elétrica, sistemas de design e programação que você vai aprender ao ler este livro são amplamente aplicáveis além da plataforma Arduino e irão prepará-lo para assumir uma variedade de projetos de engenharia, quer eles usem o Arduino ou qualquer outra plataforma.

Para quem é este livro

Este livro é para os entusiastas do Arduino de todos os níveis de experiência. Os capítulos evoluem um com base no outro, utilizando conceitos e componentes de projeto de capítulos anteriores para desenvolver ideias mais complexas. Mas não se preocupe. Sempre que se deparar com ideias novas e complexas, uma referência cruzada o lembra de onde você encontrou pela primeira vez quaisquer conceitos básicos relevantes para que possa facilmente refrescar sua memória.

Este livro assume que você tem pouca ou nenhuma experiência anterior de trabalho com a programação ou engenharia elétrica. Para dar conta dos vários níveis de experiência dos leitores, o livro apresenta uma série de seções e quadros opcionais, ou breves excertos, que explicam um determinado conceito em maior detalhe. Embora esses quadros não sejam obrigatórios para você ganhar um bom entendimento de como usar o Arduino, eles fornecem um olhar mais atento a temas técnicos para o leitor mais curioso.

O que você aprenderá neste livro

Este livro não é um livro de receitas. Se você quer seguir instruções passo a passo que informam exatamente como construir um projeto específico sem realmente explicar por que está fazendo o que está fazendo, este livro não é para você. Você pode pensar neste livro como uma introdução à engenharia elétrica, ciência da computação, design de produto e um pensamento de alto nível usando o Arduino como um veículo para ajudá-lo a experimentar esses conceitos de uma maneira prática.

Ao construir os componentes de hardware dos projetos Arduino demonstrados neste livro de hardware, você aprenderá não só a ligar as coisas, mas também a ler esquemas, saber por que determinadas partes são usadas para funções específicas, e a ler especificações (ou *datasheets*) que lhe permitirão escolher as peças adequadas para construir seus próprios projetos. Ao escrever um software, forneço seu código completo, mas primeiro você percorrerá vários processos iterativos para criar o programa final. Isso vai ajudar a reforçar funções de programação específicas e as boas práticas de formatação de código, além de proporcionar uma melhor compreensão de algoritmos.

Este livro ensina conceitos de física, algoritmos, princípios de design digital e conceitos de programação específicos do Arduino. Minha esperança é que trabalhar com os projetos neste livro não só vai tornar você um desenvolvedor bem versado em Arduino, mas também lhe dará as habilidades de que precisa para desenvolver sistemas elétricos mais complexos, e para prosseguir os esforços de engenharia em outros campos, e com diferentes plataformas.

Recursos usados neste livro

Os seguintes recursos e ícones são utilizados neste livro para ajudar a chamar a atenção para algumas das informações mais importantes ou úteis neste livro:

ATENÇÃO Certifique-se de tomar o devido cuidado quando você vir um desses apartes. Quando procedimentos específicos podem causar danos à eletrônica se realizados de forma incorreta, você verá um desses apartes.

DICA Esses apartes contêm dicas rápidas sobre como executar tarefas simples que podem ser úteis para a tarefa à mão.

NOTA Esses apartes contêm informações adicionais que podem ser de importância para você, incluindo links para vídeos e material online que facilitarão acompanhar o desenvolvimento de um projeto particular.

TÍTULO DE EXEMPLO

Esses apartes entram em profundidade adicional sobre o tema atual ou um tópico relacionado.

Obtendo as peças

Para sua sorte, você pode facilmente obter os componentes necessários para executar os projetos neste livro em lojas especializadas na sua cidade ou mesmo na internet!

No início de cada capítulo, você encontrará uma lista detalhada das peças necessárias para concluir cada projeto.

Do que você precisa

Além das peças reais que você vai usar para construir seus projetos Arduino, existem algumas outras ferramentas e materiais de que precisa em suas aventuras Arduino. Mas o importante é que você precisará de um computador que seja compatível com o ambiente de desenvolvimento integrado (IDE) Arduino (Mac OSX 10.4+, Windows XP+ ou uma distribuição Linux). Fornecerei instruções para todos os sistemas operacionais quando necessário.

Você também pode querer algumas ferramentas adicionais que serão usadas ao longo do livro para depurar, montar hardware etc. Estes não são explicitamente necessários para concluir os projetos neste livro. À medida que você desenvolver seu conjunto de habilidades de engenharia elétrica, essas ferramentas serão úteis para outros projetos. Eu recomendo o seguinte:

- Solda e um ferro de solda (Nota: você não vai precisar de solda para completar os projetos neste livro, mas pode desejar montar seus próprios circuitos em uma protoplaca, ou você pode querer comprar shields que exijam a montagem com solda.)
- Um multímetro (Isso vai ser útil para conceitos de depuração dentro deste livro, mas não é explicitamente necessário.)
- Um conjunto de chaves de fenda
- Uma pistola de cola quente

Código-fonte e conteúdo digital

O site principal deste livro é www.exploringarduino.com, que é mantido pelo autor. Você vai encontrar downloads de código para cada capítulo neste site (juntamente com vídeos, links e outros materiais úteis). A Wiley também mantém um repositório de conteúdo digital que acompanha este livro em www.wiley.com. Especificamente para este livro, o download do código está na guia Download Code www.wiley.com/go/exploringarduino. A Editora Alta Books também disponibiliza os códigos em seu site. Entre em www.altabooks.com.br e procure pelo nome do livro.

No início de cada capítulo, você encontra o local dos principais arquivos de código para o uso. Também poderá encontrar referências aos nomes dos arquivos de código conforme necessário no sumário do capítulo e ao longo do texto.

O código disponível em www.exploringarduino.com, www.altabooks.com.br e www.wiley.com é fornecido em arquivos ZIP compactados. Depois de baixar o código, basta descompactá-lo com uma ferramenta apropriada.

NOTA

A Editora Alta Books ressalta que a disponibilidade do conteúdo digital e a manutenção dos sites do autor e da Wiley são de responsabilidade dos mesmos.

Errata

Fizemos, e continuamos fazendo, todos os esforços para garantir que não haja erros no texto ou no código. Mas ninguém é perfeito, e erros ocorrem. Se você encontrar problemas neste livro, como um erro de ortografia ou um trecho de código falho, ficaríamos muito gratos por seu feedback. Contribuindo para essa lista de erratas, você pode poupar outros leitores de horas de frustração e, ao mesmo tempo, pode nos ajudar a fornecer informações de qualidade ainda maior.

Para encontrar a página de erratas deste livro, visite www.altabooks.com.br e procure pelo nome do livro. Nessa página, você poderá verificar se o livro tem erratas e também indicar possíveis correções.

Material e suporte suplementar

Durante suas aventuras com o Arduino, inevitavelmente você terá perguntas e, talvez, problemas. Uma das melhores partes sobre como usar o Arduino é a excelente comunidade de suporte que você pode encontrar na web. Essa base extremamente ativa de usuários do Arduino irá prontamente ajudá-lo ao longo de seu caminho. Eis apenas alguns dos recursos que você vai achar útil em sua jornada:

- Official Arduino Reference
www.arduino.cc/en/Reference/HomePage
- My Arduino Tutorial Series
www.jeremyblum.com/category/arduino-tutorials
- adafruit Industries Arduino Tutorial Series
learn.adafruit.com/category/learn-arduino
- SparkFun's Electronics Tutorials
learn.sparkfun.com/tutorials
- The Official Arduino Forum
www.arduino.cc/forum
- The element14 Arduino Community
www.element14.com/community/groups/arduino

Se você já esgotou todos esses recursos e ainda não conseguiu resolver o problema, entre em contato comigo pelo Twitter (@sciguy14); talvez eu possa ajudar. Ou pode entrar em contato diretamente pela página de contato em meu site (www.jeremyblum.com/contact), mas, em geral, não garanto tempos de resposta rápidos.

NOTA Todos os sites relacionados se encontram em inglês.

O que é um Arduino?

A melhor parte sobre a plataforma de prototipagem Arduino é que ela pode ser qualquer coisa que você quiser que ela seja. O Arduino poderia ser um sistema de controle automático de irrigação de plantas. Pode ser um servidor web. Pode até ser um piloto automático de um quadricóptero.

O Arduino é uma plataforma microcontroladora de desenvolvimento associada com uma linguagem de programação intuitiva que você desenvolve utilizando o ambiente de desenvolvimento integrado (IDE) Arduino. Ao equipar o Arduino com sensores, atuadores, luzes, alto-falantes, módulos adicionais (chamados de *shields*), e outros circuitos integrados, você pode transformar o Arduino em um “cérebro” programável para praticamente qualquer sistema de controle.

É impossível cobrir tudo de que o Arduino é capaz, porque as possibilidades são limitadas apenas por sua imaginação. Assim, este livro serve como um guia para você se familiarizar com as funcionalidades do Arduino, executando uma série de projetos que lhe darão as habilidades de que precisa para desenvolver seus próprios projetos.

Você vai aprender mais sobre o Arduino e as variações disponíveis da placa no Capítulo 1, “Ligando o Arduino e fazendo-o piscar”. Se está ansioso para conhecer todo o funcionamento interno do Arduino, você está com sorte: ele é totalmente *open source*, e todos os esquemas e documentação estão disponíveis gratuitamente no site do Arduino. O Apêndice A, “Decifrando as especificações ATmega e o esquema do Arduino”, aborda algumas das especificações técnicas do Arduino.

Uma plataforma *open source*

Se é novo ao mundo do *open source*, você terá uma boa surpresa. Este livro não entra em detalhes sobre o movimento hardware *open source*, mas vale a pena saber um pouco sobre as ideologias que tornam o trabalho com o Arduino tão maravilhoso. Se você quiser um resumo completo do que um hardware *open source* é, confira as definições oficiais no site Open Source Hardware Association (www.oshwa.org/definition), em inglês.

NOTA Saiba tudo sobre o movimento *open source* a partir de minha palestra TEDx

Talk: www.jeremyblum.com/portfolio/tedx-cornell-university-2011/.

Como o Arduino é um hardware *open source*, todos os arquivos de projeto, diagramas esquemáticos, e código-fonte estão disponíveis gratuitamente para todos. Isso não só significa que você pode facilmente hackear o Arduino para servir a uma função muito particular, mas também que pode até mesmo integrar a plataforma Arduino em seus projetos, fazer e vender clones Arduino, e usar as bibliotecas de software do Arduino em outros projetos. Embora este livro concentre-se principalmente no uso de hardware

Arduino oficial, você também pode usar centenas de placas derivadas do Arduino (muitas vezes com funções específicas adicionadas a elas) para criar os projetos neste livro.

A licença *open source* do Arduino também permite a reutilização comercial de seus projetos (desde que você não utilize a marca Arduino em seus projetos). Então, se usar um Arduino para prototipar um projeto promissor e quiser transformá-lo em um produto comercial, você pode fazer isso. Por exemplo, você vai ler sobre produtos como a impressora 3D MakerBot Replicator, que usa eletrônica baseada na plataforma Arduino Mega (www.thingiverse.com/thing:16058). (Revelação de interesses: fui eu que projetei essa placa-mãe!)

Certifique-se de respeitar as licenças de código-fonte e hardware que você usa ao longo deste livro. Algumas licenças exigem que você forneça os devidos créditos do autor original quando publicar um projeto baseado no trabalho anterior dele. Outros exigem que você sempre compartilhe as melhorias feitas sob uma licença equivalente. Esse compartilhamento ajuda a comunidade crescer, e resulta em toda essa maravilhosa documentação e suporte online que você, sem dúvida, consultará muitas vezes durante suas aventuras com o Arduino. Todos os exemplos de código que eu escrevi para este livro (salvo indicação em contrário) são licenciados sob a GNU General Public License (GPL), permitindo que você os use para qualquer coisa que quiser.

Para além deste livro

Alguns de vocês podem já estar familiarizados com minha popular série no YouTube Arduino e meus tutoriais eletrônicos (https://www.youtube.com/watch?v=fCxzA9_kg6s&list=PLV009FNOX7Tf-XSyghg2vrSYXw1QcCHaX). Faço referências a eles durante todo este livro como uma maneira de você ver orientações mais detalhadas dos tópicos abordados aqui. Se você está curioso sobre algumas das coisas notáveis que pode fazer com inteligentes combinações de eletrônica, microcontroladores, informática e criatividade, confira meu portfólio (www.jeremyblum.com/portfolio) para ver uma amostra de projetos. Assim como o Arduino, mais do que eu faço é liberado por meio de licenças *open source* que permitem que você facilmente duplique meu trabalho para suas próprias necessidades.

Estou ansioso para ouvir sobre o que você fez com as habilidades adquiridas com este livro. Eu o incentivo a compartilhá-los comigo e com o resto do mundo. Boa sorte em suas aventuras com o Arduino!

Fundamentos de engenharia do Arduino

Nesta parte

Capítulo 1: Ligando o Arduino e fazendo-o piscar

Capítulo 2: Entradas e saídas digitais, e modulação por largura de pulso

Capítulo 3: Lendo sensores analógicos

Ligando o Arduino e fazendo-o piscar

Peças que você precisa para este capítulo:

Arduino Uno

Cabo USB

CÓDIGO E CONTEÚDO DIGITAL PARA ESTE CAPÍTULO

Links para baixar código, vídeos e outros conteúdos digitais para este capítulo podem ser encontrados em www.exploringarduino.com/content/ch1.

Além disso, os códigos também podem ser encontrados em www.altabooks.com.br, procurando pelo nome do livro. Outra opção é em www.wiley.com/go/exploringarduino, na guia Download Code. Eles estão no arquivo chapter 01 download e individualmente nomeados de acordo com seus nomes ao longo do capítulo.

Agora que você tem alguma perspectiva sobre a plataforma Arduino e suas capacidades, é hora de explorar suas opções no mundo do Arduino. Neste capítulo, você irá examinar o hardware disponível, aprender sobre o ambiente de programação e linguagem, e instalar e fazer funcionar seu primeiro programa. Depois de adquirir controle sobre as funcionalidades que o Arduino pode fornecer, você vai escrever seu primeiro programa e fazer o Arduino piscar!

NOTA Para acompanhar, juntamente com um vídeo que apresenta a plataforma Arduino, visite www.jeremyblum.com/2011/01/02/arduino-tutorial-series-it-begins/.

Explorando a plataforma do Arduino

Em suas aventuras com o Arduino, você vai depender de três componentes principais para seus projetos:

- A placa Arduino em si
- Hardware externo (incluindo shields e circuitos manufaturados, que você vai explorar ao longo deste livro)
- O ambiente de desenvolvimento integrado do Arduino, ou IDE Arduino

Todos esses componentes do sistema funcionam em conjunto para permitir que você faça qualquer coisa com o Arduino.

Você tem diversas opções quando se trata de placas de desenvolvimento Arduino, mas este livro concentra-se no uso de placas oficiais Arduino. Como as placas são todas projetadas para serem programáveis via Arduino IDE, você geralmente pode usar qualquer uma das placas Arduino modernas para concluir os projetos neste livro com mínimas alterações, ou mesmo nenhuma. Mas, quando necessário, você verá advertências sobre a utilização de placas diferentes para vários projetos. A maioria dos projetos usa o Arduino Uno.

Você começa a explorar a funcionalidade básica fornecida em toda placa Arduino. Depois, você examina as diferenças entre cada placa moderna para poder tomar uma decisão informada ao escolher uma placa para usar em seu próximo projeto.

Funcionalidade do Arduino

Todas as placas Arduino têm algumas características e funções específicas. Dedique um tempo para examinar o Arduino Uno (veja a Figura 1-1); será seu equipamento básico. Esses são alguns componentes-chave com que você vai lidar:

- Microcontrolador Atmel
- Interface(s) de programação/comunicação USB
- Regulador de tensão e conexões de alimentação
- Pinos de interrupção de E/S
- LEDs de depuração, energia e RX/TX
- Botão de reset
- Conector(es) de programação serial no circuito (ICSP)

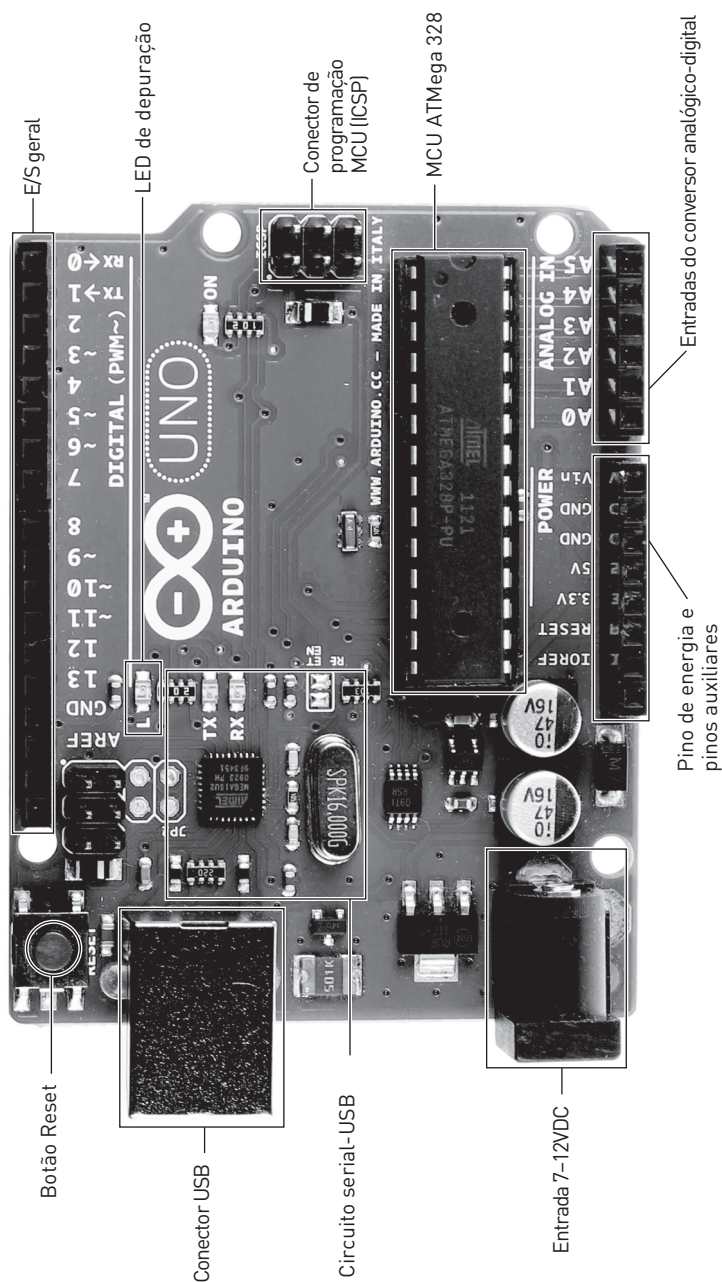


Figura 1-1: Componentes da placa Arduino Uno

Microcontroladores Atmel

No coração de cada Arduino está uma unidade microcontroladora (MCU) Atmel. A maioria das placas Arduino, incluindo o Arduino Uno, usa um microcontrolador AVR ATMega. O Arduino Uno na Figura 1-1 utiliza um 328p ATMega. O Due é uma exceção; ele usa um microcontrolador ARM Cortex. Esse microcontrolador é responsável por manter a totalidade de seu código compilado e executar os comandos que você especifica. A linguagem de programação Arduino dá acesso aos periféricos do microcontrolador, incluindo conversores analógico-digitais (ADCs), pinos de entrada/saída (E/S) de uso geral, barramento de comunicação (incluindo I2C e SPI) e interfaces seriais. Toda essa funcionalidade útil é obtida a partir dos minúsculos pinos no microcontrolador da placa Arduino nos quais você pode conectar fios ou shields usando conectores fêmeas. Um ressonador cerâmico de 16MHz é ligado aos pinos do clock da ATMEGA, que serve como referência por meio da qual todos os comandos de programa são executados. Você pode usar o botão Reset para reiniciar a execução de seu programa. A maioria das placas Arduino vem com um LED de depuração já ligada ao pino 13, que permite que você execute seu primeiro programa (fazer um LED piscar) sem conectar quaisquer circuitos adicionais.

Interfaces de programação

Normalmente, os programas de microcontroladores ATMega são escritos em C ou Assembly e programados por meio da interface ICSP usando um programador dedicado (veja a Figura 1-2). Talvez a característica mais importante de um Arduino é que você pode programá-lo facilmente via USB, sem utilizar um programador separado. Essa funcionalidade é possível graças ao bootloader, ou sistema de inicialização, do Arduino. O bootloader é carregado de fábrica no ATMega (usando o cabeçalho ICSP), que permite que uma USART (Universal Synchronous/Asynchronous Receiver/Transmitter) carregue seu programa no Arduino sem o uso de um programador separado. (Você pode aprender mais sobre como funciona o bootloader no quadro “A configuração do bootloader e do firmware do Arduino”.)

No caso do Arduino Uno e do Mega 2560, um microcontrolador secundário (um ATMega 16U2 ou 8U2 dependendo de sua revisão) serve como uma interface entre um cabo USB e os pinos USART seriais no microcontrolador principal. O Arduino Leonardo, que usa um ATMega 32U4 como o microcontrolador principal, tem uma USB integrada, de modo que um microcontrolador secundário não é necessário. Em placas Arduino mais antigas, um chip FTDI USB-serial era usado como a interface entre a porta serial USART do ATMega e uma conexão USB.