

Sommario

Prefazione.....	XI
A chi è diretto questo libro	XI
Come è organizzato il libro	XII
Che cosa è stato tralasciato	XV
Lo stile del codice	XVI
Note sul rilascio della piattaforma di Arduino	XVII
Note sulla terza edizione	XVIII
Convenzioni adottate nel libro	XVIII
Utilizzo degli esempi di codice	XIX
Come contattarci	XIX
Ringraziamenti per la seconda edizione	XX
Ringraziamenti per la terza edizione (Brian Jepson)	XXI
Gli autori	XXII
1. Per cominciare.....	1
Introduzione	1
1.1 Installare l'IDE (Integrated Development Environment)	6
1.2 Configurare la scheda Arduino	10
1.3 Usare l'IDE (Integrated Development Environment) per preparare uno sketch di Arduino	13
1.4 Caricare ed eseguire lo sketch Blink	16
1.5 Creare e salvare uno sketch	18
1.6 Utilizzare Arduino	22

1.7	Utilizzare Arduino con le schede che non fanno parte della distribuzione standard	26
1.8	Utilizzo di un Arduino a 32 bit (o compatibile)	30
2.	Programmare la scheda Arduino	35
	Introduzione	35
2.1	Un classico sketch di Arduino	36
2.2	Usare tipi primitivi semplici (variabili)	38
2.3	Usare numeri in virgola mobile	41
2.4	Lavorare con gruppi di valori	43
2.5	Usare le funzionalità delle stringhe di Arduino	47
2.6	Usare stringhe di caratteri in C	53
2.7	Separare in gruppi parti di testo divise da virgole	54
2.8	Convertire un numero in una stringa	57
2.9	Convertire una stringa in un numero	59
2.10	Strutturare il codice in blocchi funzionali	62
2.11	Fare in modo che una funzione restituisca più valori	66
2.12	Eseguire azioni in base a determinate condizioni	69
2.13	Ripetere una sequenza di istruzioni	70
2.14	Ripetere una serie di istruzioni adoperando un contatore	73
2.15	Uscire dai cicli	75
2.16	Eseguire varie azioni basandosi su una sola variabile	76
2.17	Confrontare caratteri e valori numerici	79
2.18	Confrontare stringhe	81
2.19	Eseguire confronti logici	83
2.20	Eseguire operazioni bit a bit	84
2.21	Combinare operazioni e assegnazioni	86
3.	Usare gli operatori matematici	89
	Introduzione	89
3.1	Aggiungere, sottrarre, moltiplicare e dividere	89
3.2	Incrementare e decrementare valori	91
3.3	Trovare il resto dopo aver diviso due valori	92
3.4	Determinare il valore assoluto	94
3.5	Limitare un numero all'interno di un intervallo di valori	95
3.6	Trovare il minimo e il massimo di determinati valori	95
3.7	Elevare un numero a una certa potenza	97
3.8	Calcolare la radice quadrata	98
3.9	Arrotondare per eccesso o per difetto i numeri in virgola mobile	98
3.10	Utilizzare funzioni trigonometriche	99
3.11	Generare numeri casuali	100
3.12	Impostare e leggere bit	103

3.13	Spostare i bit	106
3.14	Estrarre i byte alti e bassi in un int o in un long	108
3.15	Formare un int o un long con byte alti e bassi	109
4.	Comunicazioni seriali	113
	Introduzione	113
4.1	Inviare informazioni di debug da Arduino al computer	121
4.2	Inviare testo formattato e dati numerici da Arduino	125
4.3	Ricevere dati seriali in Arduino	129
4.4	Inviare più campi di testo da Arduino in un unico messaggio	134
4.5	Ricevere più campi di testo in un solo messaggio in Arduino	141
4.6	Inviare dati binari da Arduino	144
4.7	Ricevere su un computer dati binari inviati da Arduino	149
4.8	Inviare valori binari da Processing ad Arduino	151
4.9	Inviare il valore di più pin di Arduino	154
4.10	Registrare i dati di Arduino in un file che si trova nel computer	158
4.11	Inviare i dati a più dispositivi seriali	161
4.12	Ricevere dati seriali da più dispositivi	165
4.13	Usare Arduino con il Raspberry Pi	171
5.	Input digitali e analogici semplici.....	177
	Introduzione	177
5.1	Utilizzare un interruttore	181
5.2	Utilizzare un interruttore senza resistenze esterne	186
5.3	Rilevare la chiusura di un interruttore in modo affidabile	188
5.4	Determinare quanto a lungo è mantenuto premuto un interruttore	191
5.5	Leggere un tastierino a matrice	196
5.6	Leggere valori analogici	199
5.7	Cambiare l'intervallo dei valori	201
5.8	Leggere più di sei input analogici	204
5.9	Visualizzare tensioni fino a 5V	207
5.10	Rispondere ai cambiamenti della tensione	210
5.11	Misurare tensioni superiori a 5V (partitori di tensione)	212
6.	Ricavare input dai sensori.....	215
	Introduzione	215
6.1	Un Arduino con molti sensori incorporati	217
6.2	Rilevare il movimento	221
6.3	Rilevare la luce	223
6.4	Rilevare il movimento di esseri viventi	226
6.5	Misurare la distanza	228
6.6	Misurare la distanza con precisione	233

6.7	Rilevare la vibrazione	236
6.8	Rilevare il suono	237
6.9	Misurare la temperatura	242
6.10	Leggere i tag RFID	246
6.11	Seguire il movimento di un quadrante	249
6.12	Seguire il movimento rotatorio in uno sketch che esegue altre operazioni	252
6.13	Utilizzare un mouse	254
6.14	Ottenere la posizione da un GPS	259
6.15	Rilevare la rotazione utilizzando un giroscopio	264
6.16	Rilevare la direzione	267
6.17	Leggere l'accelerazione	270
7.	Output visivo.....	273
	Introduzione	273
7.1	Collegare e utilizzare i LED	277
7.2	Regolare la luminosità di un LED	281
7.3	Gestire LED ad alta potenza	282
7.4	Regolare il colore di un LED	285
7.5	Controllare un sacco di LED colorati	288
7.6	Utilizzare più LED in sequenza: creare un grafico a barre	291
7.7	Utilizzare più LED in sequenza: creare l'effetto Supercar	296
7.8	Controllare una matrice di LED utilizzando il multiplexing	297
7.9	Visualizzare immagini su una matrice di LED	301
7.10	Controllare una matrice di LED: il charlieplexing	304
7.11	Gestire display di LED a sette segmenti	310
7.12	Gestire display di LED a sette segmenti a più cifre: il multiplexing	313
7.13	Gestire display di LED a sette segmenti a più cifre utilizzando il minor numero di pin	315
7.14	Controllare un array di LED utilizzando i registri a scorrimento del MAX72xx	317
7.15	Aumentare il numero di output analogici utilizzando chip d'estensione del segnale PWM	320
7.16	Utilizzare come display un misuratore da pannello analogico	323
8.	L'output fisico	327
	Introduzione	327
8.1	Controllare la posizione di rotazione con un servomotore	331
8.2	Controllare uno o due servomotori con un potenziometro o un sensore	333
8.3	Controllare la velocità dei servomotori a rotazione continua	335
8.4	Controllare i servomotori via software	337

8.5	Gestire un motore senza spazzole (utilizzando un controllore di velocità per modellismo)	339
8.6	Controllare solenoidi e relè	340
8.7	Far vibrare un oggetto	342
8.8	Gestire un motore a spazzole utilizzando un transistor	345
8.9	Controllare la direzione di un motore a spazzole con un ponte ad H	346
8.10	Controllare la direzione e la velocità di un motore a spazzole con un ponte ad H	349
8.11	Usare sensori per controllare la direzione e la velocità dei motori a spazzole	351
8.12	Gestire un motore passo-passo bipolare	358
8.13	Gestire un motore passo-passo bipolare (utilizzando la scheda EasyDriver)	362
8.14	Gestire un motore passo-passo unipolare con il chip driver ULN2003A	365
9.	L'output audio	369
	Introduzione	369
9.1	Riprodurre toni	373
9.2	Riprodurre una semplice melodia	375
9.3	Generare più toni contemporaneamente	377
9.4	Generare toni audio senza interferire con PWM	379
9.5	Controllare MIDI	381
9.6	Creare un sintetizzatore audio	384
9.7	Sintesi audio di alta qualità: Attain	386
10.	Controllare da remoto dispositivi esterni	391
	Introduzione	391
10.1	Rispondere a un telecomando a infrarossi	392
10.2	Decodificare i segnali dei telecomandi a infrarossi	395
10.3	Imitare i segnali dei telecomandi	398
10.4	Controllare una fotocamera digitale	401
10.5	Controllare i dispositivi AC modificando un interruttore controllato in remoto	404
11.	Utilizzare gli schermi	409
	Introduzione	409
11.1	Collegare e utilizzare LCD alfanumerici	410
11.2	Formattare il testo	414
11.3	Accendere e spegnere lo schermo e far lampeggiare il cursore	417
11.4	Far scorrere il testo	418
11.5	Visualizzare simboli speciali	421

11.6 Creare caratteri personalizzati	423
11.7 Visualizzare simboli più lunghi di un solo carattere	426
11.8 Visualizzare pixel più piccoli di un solo carattere	428
11.9 Come scegliere un LCD grafico	431
11.10 Controllare un LCD a colori	433
11.11 Controllare uno schermo OLED monocromatico	436
12. Utilizzare data e ora	443
Introduzione	443
12.1 Utilizzare millis per determinare una durata	443
12.2 Creare pause nello sketch	445
12.3 Misurare con maggiore precisione la durata di un impulso	449
12.4 Utilizzare Arduino come un orologio	450
12.5 Creare un avviso per chiamare periodicamente una funzione	458
12.6 Utilizzare un orologio in tempo reale	461
13. Comunicare utilizzando I2C e SPI	467
Introduzione	467
13.1 Collegamento di più dispositivi I2C	474
13.2 Collegamento di più dispositivi SPI	477
13.3 Lavorare con un circuito integrato I2C	480
13.4 Aumentare l'I/O con un modulo di espansione I2C	484
13.5 Far comunicare tra loro due o più schede Arduino	487
13.6 Utilizzare l'accelerometro del nunchuck della Wii	491
14. La comunicazione wireless	499
Introduzione	499
14.1 Inviare messaggi utilizzando moduli wireless economici	499
14.2 Collegare Arduino a una rete ZigBee o 802.15.4	506
14.3 Inviare un messaggio a uno specifico XBee	514
14.4 Scambiare tra XBee i dati ricavati da alcuni sensori	517
14.5 Attivare un attuatore collegato a un XBee	523
14.6 Comunicare con dispositivi Bluetooth classici	528
14.7 Comunicare con dispositivi Bluetooth a basso consumo	530
15. WiFi ed Ethernet	535
Introduzione	535
15.1 Collegamento a una rete Ethernet	537
15.2 Ottenere automaticamente un indirizzo IP	542
15.3 Inviare e ricevere messaggi semplici (UDP)	543
15.4 Usare una scheda Arduino con WiFi integrato	550
15.5 Connettersi a WiFi con moduli economici	553

15.6 Estrarre dati da una risposta web	558
15.7 Richiedere dati da un server web utilizzando XML	562
15.8 Trasformare Arduino in un server web	565
15.9 Gestire richieste provenienti dal Web	570
15.10 Gestire le richieste di pagine specifiche	573
15.11 Utilizzare HTML per formattare le risposte del server web	578
15.12 Richiedere dati web utilizzando i form (POST)	582
15.13 Gestire pagine web che contengono grandi quantità di dati	586
15.14 Inviare messaggi a Twitter	592
15.15 Scambiare dati per IoT (Internet of Things)	595
15.16 Pubblicare dati su un broker MQTT	596
15.17 Ricevere dati da un broker MQTT	597
15.18 Ricavare l'ora da un Internet time server	599
16. Usare, modificare e creare librerie	605
Introduzione	605
16.1 Utilizzare le librerie integrate	606
16.2 Installare librerie di terze parti	609
16.3 Modificare una libreria	611
16.4 Creare una libreria personalizzata	614
16.5 Creare una libreria che utilizza altre librerie	621
16.6 Aggiornare librerie di terze parti per Arduino 1.0	626
17.Codice avanzato e gestione della memoria.....	629
Introduzione	629
17.1 Il processo di costruzione di Arduino	631
17.2 Determinare la quantità di RAM libera e utilizzata	634
17.3 Salvare e ricavare valori numerici nella memoria del programma (program memory)	637
17.4 Archiviare e recuperare stringhe dalla memoria del programma	640
17.5 Utilizzare #define e const al posto di valori interi	642
17.6 Utilizzare compilazioni condizionali	643
18. Utilizzare l'hardware del controller.....	647
Introduzione	647
18.1 Salvare dati nella memoria EEPROM permanente	652
18.2 Eseguire automaticamente un'azione quando cambia lo stato di un pin	656
18.3 Eseguire azioni periodiche	658
18.4 Impostare la lunghezza e la durata degli impulsi del timer	660
18.5 Creare un generatore di impulsi	662
18.6 Modificare la frequenza del segnale PWM di un timer	665
18.7 Contare gli impulsi	668

18.8 Misurare gli impulsi con maggiore precisione	670
18.9 Misurare rapidamente i valori analogici	673
18.10 Ridurre il consumo della batteria	674
18.11 Impostare rapidamente i pin digitali	677
18.12 Caricare gli sketch attraverso un programmatore	679
18.13 Sostituire il bootloader di Arduino	681
18.14 Spostare il puntatore del mouse su un PC o un Mac	682
Appendice A. Componenti elettronici	685
Appendice B. Utilizzare i diagrammi schematici e le schede tecniche.....	691
Appendice C. Costruire e collegare un circuito.....	697
Appendice D. Consigli per la risoluzione dei problemi software.....	701
Appendice E. Consigli per la risoluzione dei problemi hardware.....	707
Appendice F. Pin digitali e pin analogici	711
Appendice G. I set di caratteri ASCII ed esteso	715
Indice analitico	719