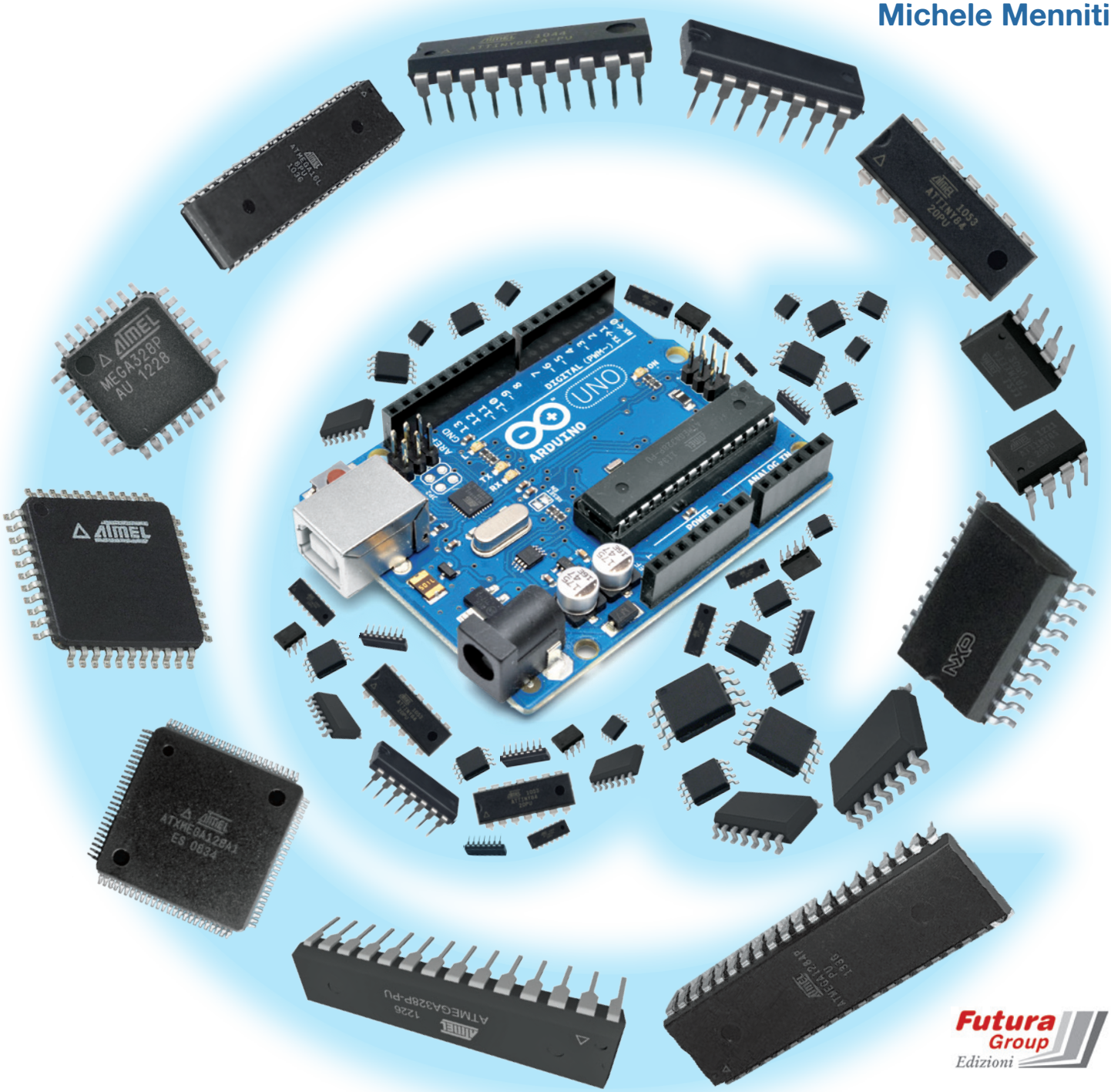
Manuale Tecnico-Pratico

ARDUINO E LE TECNICHE DI PROGRAMMAZIONE DEI MICROCONTROLLORI ATMEL®

Progettazione e realizzazione di circuiti in stand-alone

Michele Menniti



Futura Group Edizioni

Manuale Tecnico-Pratico

ARDUINO E LE TECNICHE DI PROGRAMMAZIONE DEI MICROCONTROLLORI ATMEL®

Progettazione e realizzazione di circuiti in stand-alone

di Michele Menniti

Futura Group Edizioni

Dopo oltre trent'anni trascorsi in varie università al centro-nord ed all'estero, ho deciso di ritornare in Calabria motivato più dalla stanchezza che dall'entusiasmo o dalla percezione di poter vivere il miracolo del risveglio culturale (grazie alla creazione di tre Atenei), scientifico e, perché no, industriale. Così mi sono ritrovato docente di Elettronica, prima, e di Sensori e Sistemi, dopo, all'Università Magna Græcia di Catanzaro. Giorno dopo giorno la stanchezza si allontanava e lasciava spazio all'entusiasmo che, conscio com'ero dei sacrifici che i giovani corsi di laurea in Biotecnologie e Ingegneria Biomedica imponevano ai propri studenti (disagiati, costretti a fare lezione in aule decentrate, per nulla aiutati dai mezzi di trasporto e con la difficoltà di trovare un bar dove poter mangiare anche un semplice toast per "pranzo"), era quotidianamente alimentato dalla voglia di sapere, dalla voglia di fare, dalla voglia di emergere dei nostri ragazzi.

Tanti ne ricordo. Uno di questi è Michele anche lui sempre assetato di sapere, incuriosito come tutti gli altri dal corso di Bioingegneria Elettronica e Informatica, sempre attento all'Elettronica, passione che peraltro aveva già coltivato in gioventù e che ha continuato a coltivare a mia insaputa anche dopo la laurea, cercando di trasmettere agli altri le proprie conoscenze nel modo più accurato possibile, e cercando di rendere semplici anche le cose più complicate. Accuratezza (che definirei piuttosto pignoleria) e semplificazione che emergono anche in questo suo nuovo lavoro. Una volta deciso l'argomento, nella fattispecie i metodi di programmazione dei microcontrollori mediante Arduino, ha studiato, sviscerato, sperimentato, per un anno, tutto quanto ha poi spiegato, disegnato e fotografato. Un lavoro che inizialmente doveva riguardare solo il mondo Arduino, ma che lo ha talmente preso ed entusiasmato da fargli decidere di estendere la sperimentazione a numerosi microcontrollori della ATMEL.

Due cose traspaiono dalla lettura di questo lavoro: la caparbietà e, come ho detto, la pignoleria. La prima la si evince dalle difficoltà incontrate nell'effettuare alcune prove sui componenti (difficoltà peraltro note alla comunità internazionale): invece di rifarsi alla documentazione esistente, per giustificare l'insuccesso, è riuscito a trovare una soluzione estremamente efficace al problema. La seconda, la caparbietà tipicamente calabrese, è definibile come il filo conduttore dell'intero lavoro; ognuna delle decine e decine di test di programmazione è stata sperimentata direttamente e descritta così minuziosamente che è praticamente impossibile commettere errori di esecuzione. Ognuno dei singoli test è stato rappresentato mediante immagini chiarissime, create con software specifici.

È questo un manuale tecnico-pratico che non può mancare sul tavolo del laboratorio di chi pratica (o intenderà farlo) la programmazione dei microcontrollori ATMEL.

Ringrazio Michele per avermi dato l'opportunità, con questo suo lavoro, di fare un augurio sia ad egli stesso che a tutti i suoi colleghi:

" Auguro a tutti di fare altrettanto e vi invito con calore a non permettere a chicchessia di credere, o far credere, che qui ci siano solo dei pocodibuono (per farla breve) anzi, tutt'altro, c'è gente molto capace e degna della massima stima e rispetto. Camminate sempre a testa alta senza vergogna".

Antonino S. Fiorillo, Ph.D

Professor
UNIVERSITY of MAGNA GRAECIA, I
Prof. of Microelectronics,
Prof. of Sensors and Electronic Systems

Director
VENETO RESEARCH CONSORTIUM
Department of Biomedical Sensors and Technologies
Saccolongo-Padova, I

1

Introduzione

Il successo del progetto "Arduino" è scaturito principalmente dall'idea di sposare l'Open-Source e, soprattutto, di realizzare una scheda (board) ed un semplice software di programmazione, che permettessero a chiunque, esperto o no di elettronica e di programmazione, di poter sviluppare progetti, anche piuttosto complessi, semplicemente aggiungendo la componentistica esterna e scrivendo l'opportuno programma da caricare sul microcontrollore, cuore della board. Il progetto ha dato vita a diverse varianti della board, i modelli sicuramente più diffusi e venduti oggi sono Arduino UNO e Arduino MEGA (nelle sue varie versioni). La differenza sostanziale tra le due famiglie di board consiste nel microcontrollore usato (ATmega328P-PU nel caso della UNO e ATmega2560-AU nel caso della MEGA) e, di conseguenza nelle capacità di memoria e nel numero di pin disponibili per gli interfacciamenti, entrambi certamente più elevati nel caso della Arduino MEGA.

Queste schede hanno costi molto accessibili, tuttavia sin da subito si è sentita la necessità di "liberare" la board a progetto finito, per risparmiare denaro e rendere anche il lavoro più professionale. Come ben sappiamo una board Arduino è fondamentalmente costituita da: sezione di alimentazione (presa jack per alimentazione esterna con regolatore a 5V, o porta USB, circuito di riconoscimento della fonte di alimentazione) sezione della conversione USB-Seriale (presa USB e chip di conversione) microcontrollore, interfacciamento I/O (Input/Output) con il mondo esterno (la serie di connettori header presenti ai lati della board) interfaccia di programmazione ICSP (una o due). Alcune considerazioni sono indispensabili per giungere al concetto di stand-alone (letteralmente "stare da solo"): in molti casi la comunicazione col PC, a progetto finito, serve solo per eventuali upgrade del software, e l'alimentazione interna della board è spesso insufficiente per le periferiche esterne; la connessione tra board e componentistica esterna mediante header e cavetti jumper può andar bene in fase prototipale e sperimentale, non certo a lavoro finito, dove bisogna assolutamente escludere la possibilità di falsi contatti. Una possibile soluzione è la realizzazione dei cosiddetti shield, cioè dei circuiti stampati che presentano sui lati delle serie di connettori pin compatibili, per numero e posizione, con gli header della board Arduino; lo shield in tal modo si innesta a pressione sulla board Arduino, garantendo un'ottima qualità nei contatti. Esistono shield che svolgono funzioni predefinite, realizzati dalla stessa società che produce Arduino, o da terze parti; un esempio su tutti è l'Arduino Ethernet, che fornisce appunto la board Arduino di tale importante funzionalità; così come ne esistono altre con funzionalità diverse, ma poi non sempre è possibile dotare una sola board Arduino di più schede shield, a volte semplicemente per ragioni meccaniche. In molti casi è proprio impossibile realizzare uno shield, ed ecco che si arriva alla realizzazione dei circuiti stand-alone.

Resta a questo punto la problematica della programmazione del microcontrollore montato in

stand-alone. Lo scopo fondamentale di questo Manuale tecnico è quello di spiegare come realizzare con estrema semplicità circuiti in stand-alone e come programmare i relativi microcontrollori con tutte le tipologie di tecniche note, anche se non abbiamo trascurato alcune tecniche che consentono di riparare, o semplicemente aggiornare i microcontrollori montati sulle schede Arduino.
Allo scopo abbiamo progettato un PCB opzionale che, a scelta dell'utente, potrà essere usato come shield o come scheda esterna collegabile ad Arduino, che faciliterà enormemente le operazioni di programmazione dei microcontrollori, oltre ad una serie di PCB accessori per estendere la programmazione a quasi tutte le intere famiglie ATMEL ATmega e ATtiny. Lo presenteremo nel capitolo 4 di questo Manuale.
Specifichiamo che il presente Manuale è rivolto indistintamente a: professionisti, esperti, principianti e neofiti; la semplicità delle spiegazioni e le numerose immagini non richiedono alcuna particolare conoscenza per la messa in pratica di quanto si studierà. Inoltre sottolineiamo che tutti i contenuti sono applicabili non solo alle già citate board Arduino UNO (versioni r1, r2, r3) e MEGA (versioni 1280, 2560 e ADK) ma anche alla precedente e diffusissima versione dell'Arduino UNO, denominata Arduino Duemilanove (basata sempre sull'ATmega328P).
Alcune delle tecniche descritte potranno essere applicate anche a board diverse, come le: Mini, Nano, Micro o altre, a condizione che siano dotate di connettore seriale o ISP[1] e che siano alimentabili a 5V.
Ci preme concludere con una importante raccomandazione, rivolta soprattutto a quegli utenti che hanno sempre molta fretta di fare i loro esperimenti e risolvere i loro problemi: per quanto possibile abbiamo sempre riportato interamente i passaggi per svolgere le varie tecniche ma molte volte abbiamo anche dovuto dare per scontate alcune informazioni, in quanto ampiamente trattate all'inizio dello stesso capitolo o all'interno di capitoli precedenti. Saltare direttamente al capitolo o paragrafo di interesse può significare perdere importanti informazioni o passaggi esplicativi, ai fini della comprensione delle tecniche; questo potrebbe comportare la non riuscita di un esperimento.
Laddove necessario abbiamo terminato il capitolo con una serie di note per spiegare come creare ulteriori configurazioni con impostazioni proprie, partendo da quelle illustrate in precedenza, ma bisogna prima farsi un'idea chiara delle tecniche, in caso contrario ci si espone ad errori, magari banali, che però impediscono il buon fine dell'operazione.
I capitoli 2, 3, 4 e 5 contengono informazioni importantissime che vanno lette e, possibilmente, apprese; dal capitolo 6 iniziano le tecniche di programmazione, sempre precedute da note introduttive esplicative, e spesso seguite da note generali riepilogative.
Questo Manuale nasce sull'onda del grande entusiasmo generato dalla mia "Guida alla programmazione ISP e seriale dei micro ATMEL", giunta nel luglio 2012 alla quarta versione, distribuita on-line gratuitamente e che conta ad oggi oltre 15.000 download; l'esperienza maturata in questi anni, anche a motivo delle innumerevoli consulenze fatte ai lettori, mi ha fatto decidere di dare origine ad un nuovo lavoro molto più completo e complesso, che ha richiesto un anno di ricerca e sperimentazione.
Riguardo tutte le tecniche e gli esperimenti presentati in questo Manuale l'Autore garantisce di averli realizzati e testati con successo uno per uno, ma non si assume alcuna responsabilità nel caso si verificassero malfunzionamenti o danni derivanti dalla loro replicazione. In ogni caso l'Autore è disponibile, sulle pagine del blog aperto espressamente per questa pubblicazione, a fornire chiarimenti e/o suggerimenti a riguardo.

Buona lettura e buon divertimento. □

Prof. Michele Menniti

1 Il termine ISP è acronimo di In–System Programming e indica una tecnica di programmazione diretta del micro, con la quale è possibile caricare, nella sua memoria flash, un qualsiasi tipo di firmware.

Sommario